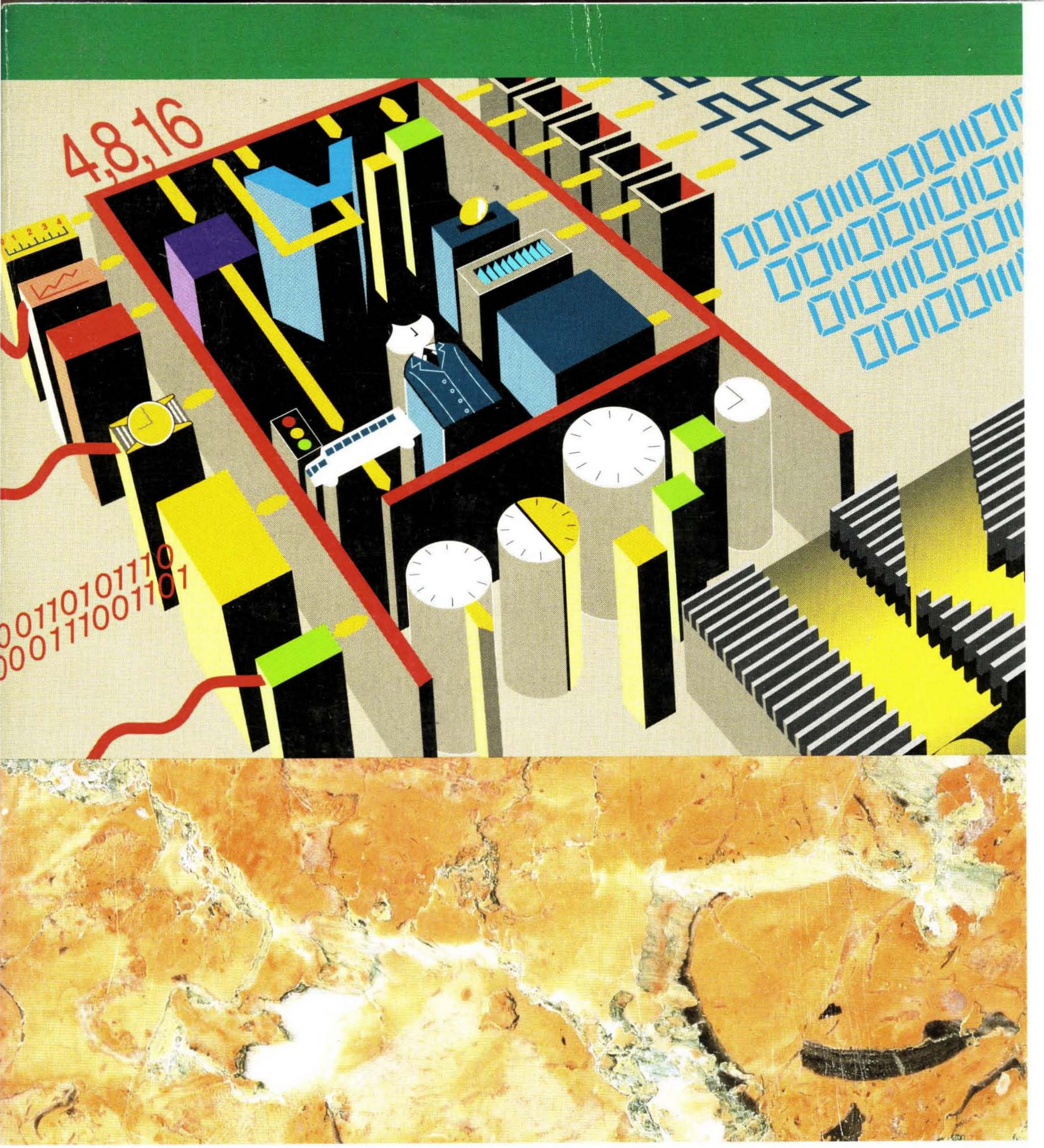


SINGLE-CHIP MICROCONTROLLER DATA BOOK

NEC

NEC Electronics Inc. NEC

1990
Single-Chip Microcontroller
Data Book

May 1990
Document No. 50053

The information in this document is subject to change without notice. Devices sold by NEC Electronics Inc. are covered by the warranty and patent indemnification provisions appearing in NEC Electronics Inc. Terms and Conditions of Sale only. NEC Electronics Inc. makes no warranty, express, statutory, implied, or by description, regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. NEC Electronics Inc. makes no warranty of merchantability or fitness for any purpose. NEC Electronics Inc. assumes no responsibility for any errors that may appear in this document. NEC Electronics Inc. makes no commitment to update or to keep current the information contained in this document.

Contents

Section 1
Selection Guides

Section 2
Reliability and Quality Control

Section 3
μPD7500 Series: 4-Bit, CMOS Microcomputers

Contents

Section 4
μPD75000 Series:
4-Bit, High-Integration Microcomputers

Section 5
μPD7800 Series:
8-Bit, General-Purpose Microcomputers

Section 6
μPD78K2 Series:
8-Bit, Advanced Microcomputers

Section 7
μPD78K3 Series:
16-Bit, Advanced Microcomputers

Section 8
μPD722x Series:
Intelligent LCD Controller/Drivers

Section 9
Development Tools

4-Bit; μPD7500 Series

4-Bit; μPD75000 Series

8-Bit; μPD7800 Series

8-Bit; μPD78K2 Series

8-Bit; μPD78K2 Series (cont)

8/16-Bit; μPD78K3 Series

Section 10
Package Drawings

Numerical Index

Numerical Index (cont)

Section 1
Selection Guides

Part Numbering System

μPD72001L	Typical microdevice part number
μP	NEC monolithic silicon integrated circuit
D	Device type (D = digital MOS)
72001	Device identifier (alphanumeric)
L	Package type (L = PLCC)

A part number may include an alphanumeric suffix that identifies special device characteristics; for example, μPD72001L-11 has an 11-MHz data clock rating.

4-Bit, Single-Chip CMOS Microcomputers

Device, µPD	Features	Clock (MHz)	Supply Voltage (V)	ROM (X8)	RAM (X4)	I/O	# Package	Pins
7502	LCD controller/driver	0.41	2.5 to 6.0	2K	128	23	QFP	64
7503	LCD controller/driver	0.41	2.5 to 6.0	4K	224	23	QFP	64
7507	General-purpose	0.41	2.5 to 6.0	2K	128	32	DIP SDIP QFP	40 40 52
7507H	General-purpose	4.19	2.7 to 6.0	2K	128	32	DIP SDIP QFP	40 40 52
7508	General-purpose	0.41	2.5 to 6.0	4K	224	32	DIP SDIP QFP	40 40 52
7508H	General-purpose	4.19	2.7 to 6.0	4K	224	32	DIP SDIP QFP	40 40 52
75CG08	Piggyback EPROM	0.41	4.5 to 5.5	2K or 4K	224	32	Ceramic DIP	40
75CG08H	Piggyback EPROM	4.19	4.5 to 5.5	2K or 4K	224	32	Ceramic DIP	40
7527A	FIP controller/driver	0.61	2.7 to 6.0	2K	128	35	DIP SDIP	42 42
7528A	FIP controller/driver	0.61	2.7 to 6.0	4K	160	35	DIP SDIP	42 42
75CG28	Piggyback EPROM; FIP controller/driver	0.5	4.5 to 5.5	4K	160	35	Ceramic DIP	42
7533	A/D converter	0.5	2.7 to 6.0	4K	160	30	DIP SDIP QFP	42 42 44
75CG33	Piggyback EPROM; A/D converter	0.5	4.5 to 5.5	4K	160	30	Ceramic DIP	42
7537A	FIP controller/driver	0.61	2.7 to 6.0	2K	128	35	DIP SDIP	42 42
7538A	FIP controller/driver	0.61	2.7 to 6.0	4K	160	35	DIP SDIP	42 42
75CG38	Piggyback EPROM; FIP controller/driver	0.61	4.5 to 5.5	4K	160	35	Ceramic DIP	42
7554	Serial I/O; external clock or RC oscillator	0.71	2.5 to 6.0	1K	64	16	SDIP SOP	20 20
7554A	Serial I/O; external clock or RC oscillator	0.71	2.0 to 6.0	1K	64	16	SDIP SOP	20 20
75P54	Serial I/O; external clock or RC oscillator	0.71	4.5 to 6.0	1K OTPROM	64	16	SDIP SOP	20 20
7564/7564A	Serial I/O; ceramic oscillator	0.71	2.7 to 6.0	1K	64	15	SDIP SOP	20 20
75P64	Serial I/O; ceramic oscillator	0.71	4.5 to 6.0	1K OTPROM	64	15	SDIP SOP	20 20
7556	Comparator; external clock or RC oscillator	0.71	2.5 to 6.0	1K	64	20	SDIP SOP	24 24
7556A	Comparator; external clock or RC oscillator	0.71	2.0 to 6.0	1K	64	20	SDIP SOP	24 24
75P56	Comparator; external clock or RC oscillator	0.71	4.5 to 6.0	1K OTPROM	64	20	SDIP SOP	24 24
7566/7566A	Comparator; ceramic oscillator	0.71	2.7 to 6.0	1K	64	19	SDIP SOP	24 24
75P66	Comparator; ceramic oscillator	0.71	4.5 to 6.0	1K OTPROM	64	19	SDIP SOP	24 24
75004	General-purpose	4.19	2.7 to 6.0	4K	512	34	SDIP QFP	42 44

# Plastic unless ceramic (or cerdip) is specified.

* Under development; consult Microcontroller Marketing for availability.

4-Bit, Single-Chip CMOS Microcomputers (cont)

Device, µPD	Features	Clock (MHz)	Supply Voltage (V)	ROM (X8)	RAM (X4)	I/O	# Package	Pins
75006	General-purpose	4.19	2.7 to 6.0	6K	512	34	SDIP QFP	42 44
75008	General-purpose	4.19	2.7 to 6.0	8K	512	34	SDIP QFP	42 44
75P008	General-purpose	4.19	4.5 to 5.5	8K OTPROM	512	34	SDIP QFP	42 44
75028 *	A/D converter	4.19	2.7 to 6.0	8K	512	48	SDIP QFP	64 64
75P036 *	A/D converter	4.19	2.7 to 6.0	16K	1024	48	SDIP QFP	64 64
75048 *	A/D converter; 1K x 4 EEPROM	4.19	2.7 to 6.0	8K	512	48	SDIP QFP	64 64
75P056 *	A/D converter; 1K x 4 EEPROM	4.19	2.7 to 6.0	16K	512	48	SDIP QFP	64 64
75104	High-end with 8-bit instruction	4.19	2.7 to 6.0	4K	320	58	SDIP QFP	64 64
75104A	High-end with 8-bit instruction	4.19	2.7 to 6.0	4K	320	58	QFP	64
75106	High-end with 8-bit instruction	4.19	2.7 to 6.0	6K	320	58	SDIP QFP	64 64
75108	High-end with 8-bit instruction	4.19	2.7 to 6.0	8K	512	58	SDIP QFP	64 64
75108A	High-end with 8-bit instruction	4.19	2.7 to 6.0	8K	512	58	QFP QFP	64 64
75P108	High-end with 8-bit instruction; on-chip OTPROM or UVEPROM	4.19	4.5 to 5.5	8K	512	58	SDIP QFP Shrink cerdip	64 64 64
75P108B	High-end with 8-bit instruction; on-chip OTPROM	4.19	2.7 to 6.0	8K	512	58	SDIP QFP	64 64
75112	High-end with 8-bit instruction	4.19	2.7 to 6.0	12K	512	58	SDIP QFP	64 64
75116	High-end with 8-bit instruction	4.19	2.7 to 6.0	16K	512	58	SDIP QFP	64 64
75P116	High-end with 8-bit instruction on-chip OTPROM	4.19	4.5 to 5.5	16K OTPROM	512	58	SDIP QFP	64 64
75206	FIP controller/driver	4.19	2.7 to 6.0	6K	369	33	SDIP QFP	64 64
75208	FIP controller/driver	4.19	2.7 to 6.0	8K	497	33	SDIP QFP	64 64
75CG208	FIP controller/driver; piggyback EPROM	4.19	4.5 to 5.5	8K	512	33	Ceramic SDIP Ceramic QFP	64 64
75212A	FIP controller/driver	4.19	2.7 to 6.0	12K	512	33	SDIP QFP	64 64
75216A	FIP controller/driver	4.19	2.7 to 6.0	16K	512	33	SDIP QFP	64 64
75CG216A	FIP controller/driver; piggyback EPROM	4.19	4.5 to 5.5	16K	512	33	Ceramic SDIP Ceramic QFP	64 64
75P216A	FIP controller/driver	4.19	4.5 to 5.5	16K OTPROM	512	33	SDIP	64
75268	FIP controller/driver	4.19	2.7 to 6.0	8K	512	32	SDIP QFP	64 64
75304	LCD controller/driver	4.19	2.7 to 6.0	4K	512	40	QFP	80
75306	LCD controller/driver	4.19	2.7 to 6.0	6K	512	40	QFP	80
75308	LCD controller/driver	4.19	2.7 to 6.0	8K	512	40	QFP	80
75P308	LCD controller/driver; on-chip OTPROM or UVEPROM	4.19	4.75 to 5.25	8K	512	40	QFP Ceramic LCC	80 80

4-Bit, Single-Chip CMOS Microcomputers (cont)

Device, µPD	Features	Clock (MHz)	Supply Voltage (V)	ROM (X8)	RAM (X4)	I/O	# Package	Pins
75312	LCD controller/driver	4.19	2.7 to 6.0	12K	512	40	QFP	80
75316	LCD controller/driver	4.19	2.7 to 6.0	16K	512	40	QFP	80
75P316	LCD controller/driver; on-chip OTPROM	4.19	4.75 to 5.25	16K OTPROM	512	40	QFP	80
75P316A *	LCD controller/driver; on-chip OTPROM or UVEPROM	4.19	2.7 to 6.0	16K OTPROM	512	40	QFP Ceramic LCC	80 80
75328	LCD controller/driver; A/D converter	4.19	2.7 to 6.0	8K	512	44	QFP	80
75P328	LCD controller/driver; A/D converter	4.19	4.5 to 5.5	8K OTPROM	512	44	QFP	80
75402A	Low-end	4.19	2.7 to 6.0	2K	64	22	DIP SDIP QFP	28 28 44
75P402	Low-end	4.19	4.5 to 5.5	2K OTPROM	64	22	DIP SDIP QFP	28 28 44
75512	High-end; A/D converter	4.19	2.7 to 6.0	12K	512	64	QFP	80
75516	High-end; A/D converter	4.19	2.7 to 6.0	16K	512	64	QFP	80
75P516	High-end; A/D converter	4.19	4.75 to 5.5	16K OTPROM	512	64	QFP Ceramic LCC	80 80

8-Bit, Single-Chip CMOS Microcomputers

Device, µPD	Features	Clock (MHz)	Supply Voltage (V)	ROM (X8)	RAM (X8)	I/O	# Package	Pins
78C10/78C10A	CMOS; A/D converter	15	4.5 to 5.5	External	256	32	QUIP SDIP QFP PLCC	64 64 64 68
78C11/78C11A	CMOS; A/D converter	15	4.5 to 5.5	4K	256	44	QUIP SDIP QFP PLCC	64 64 64 68
78C12A	CMOS; A/D converter	15	4.5 to 5.5	8K	256	44	QUIP SDIP QFP PLCC	64 64 64 68
78C14/78C14A	CMOS; A/D converter	15	4.5 to 5.5	16K	256	44	QUIP SDIP QFP PLCC	64 64 64 68
78CP14	CMOS; A/D converter	15	4.75 to 5.25	16K OTPROM	256	44	QUIP SDIP QFP PLCC	64 64 64 68
				16K UVEPROM	256	44	Ceramic QUIP Shrink cerdip	64 64
78CG14	CMOS; A/D converter; piggyback EPROM	15	4.5 to 5.5	4K, 8K or 16K	256	44	Ceramic QUIP	64
78213	CMOS; A/D converter; advanced peripherals	12	4.5 to 5.5	External	512	54	SDIP QUIP QFP PLCC	64 64 74 68
78214	CMOS; A/D converter; advanced peripherals	12	4.5 to 5.5	16K	512	54	SDIP QUIP QFP PLCC	64 64 74 68

8-Bit, Single-Chip NMOS/CMOS Microcomputers (cont)

Device, µPD	Features	Clock (MHz)	Supply Voltage (V)	ROM (X8)	RAM (X8)	I/O	# Package	Pins
78P214	CMOS; A/D converter; advanced peripherals	12	4.5 to 5.5	16K OTPROM	512	54	SDIP	64
							QUIP	64
							QFP	74
							PLCC	68
				16K UVEPROM	512	54	Shrink cerdip	64
78220	CMOS; analog comparator; large I/O	12	4.5 to 5.5	External	640	71	PLCC	84
							QFP	94
78224	CMOS; analog comparator; large I/O	12	4.5 to 5.5	16K	640	71	PLCC	84
							QFP	94
78P224	CMOS; analog comparator; large I/O	12	4.5 to 5.5	16K OTPROM	640	71	PLCC	84
							QFP	94
78233	CMOS; real-time outputs; A/D and D/A converters	12	4.5 to 5.5	External	640	64	QFP	80
							QFP	94
							PLCC	84
78234	CMOS; real-time outputs; A/D and D/A converters	12	4.5 to 5.5	16K	640	64	QFP	80
							QFP	94
							PLCC	84
78P238	CMOS; real-time outputs; A/D and D/A converters	12	4.5 to 5.5	32K OTPROM	640	64	QFP	80
							QFP	94
							PLCC	84
				32K UVEPROM	640	64	Ceramic LCC	94

8/16-Bit, Single-Chip CMOS Microcomputers

Device, µPD	Features	Clock (MHz)	Supply Voltage (V)	ROM (X8)	RAM (X8)	I/O	# Package	Pins
78310A	Real-time motor control	12	4.5 to 5.5	External	256	48	SDIP	64
							QUIP	64
							QFP	64
							PLCC	68
78312A	Real-time motor control	12	4.5 to 5.5	8K	256	48	SDIP	64
							QUIP	64
							QFP	64
							PLCC	68
78P312A	Real-time motor control	12	4.5 to 5.5	8K UVEPROM	256	48	Shrink cerdip	64
							Ceramic QUIP	64
				8K OTPROM	256	48	SDIP	64
							QUIP	64
							QFP	64
							PLCC	68
78320	High-end; advanced analog and digital peripherals	16	4.5 to 5.5	External	640	55	QFP	64
							PLCC	68
78322	High-end; advanced analog and digital peripherals	16	4.5 to 5.5	16K	640	55	QFP	64
							PLCC	68
78P322	High-end; advanced analog and digital peripherals	16	4.5 to 5.5	16K OTPROM	640	55	PLCC	68
							QFP	74
				16K UVEPROM	640	55	Ceramic LCC	68
							Ceramic LCC	74
71P301	Port and memory extender used with 7832X microcomputer family; UVEPROM or OTPROM	–	4.5 to 5.5	16K OTPROM	1K	16	PLCC	44
							QFP	64
							QUIP	64
				16K UVEPROM	1K	16	Ceramic LCC	44
							Ceramic LCC	64
							Ceramic QUIP	64

µPD75XX Series Development Tools Selection Guide

Part Number (Note 1)	Emulator*	Add-on Board*	System Evaluation Board	EPROM/OTP Device	PG-1500 Adapter (Note 2)	Absolute Assembler (Note 3)
µPD7502G-12	EVAKIT-7500B	EV7514	SE-7514A	—	—	ASM75
µPD7503G-12	EVAKIT-7500B	EV7514	SE-7514A	—	—	ASM75
µPD7507C	EVAKIT-7500B	—	—	µPD78CG08E	—	ASM75
µPD7507CU	EVAKIT-7500B	—	—	—	—	ASM75
µPD7507G-00	EVAKIT-7500B	—	—	—	—	ASM75
µPD7507HC	EVAKIT-7500B	EV7508H	—	µPD75CG08HE	—	ASM75
µPD7507HCU	EVAKIT-7500B	EV7508H	—	—	—	ASM75
µPD7507HG-22	EVAKIT-7500B	EV7508H	—	—	—	ASM75
µPD7508C	EVAKIT-7500B	—	—	µPD78CG08E	—	ASM75
µPD7508CU	EVAKIT-7500B	—	—	—	—	ASM75
µPD7508G-00	EVAKIT-7500B	—	—	—	—	ASM75
µPD75CG08E	EVAKIT-7500B	—	—	—	—	ASM75
µPD7508HC	EVAKIT-7500B	EV7508H	—	µPD78CG08HE	—	ASM75
µPD7508HCU	EVAKIT-7500B	EV7508H	—	—	—	ASM75
µPD7508HG-22	EVAKIT-7500B	EV7508H	—	—	—	ASM75
µPD75CG08HE	EVAKIT-7500B	EV7508H	—	—	—	ASM75
µPD7527AC	EVAKIT-7500B	EV7528	—	µPD78CG28E	—	ASM75
µPD7527ACU	EVAKIT-7500B	EV7528	—	—	—	ASM75
µPD7528AC	EVAKIT-7500B	EV7528	—	µPD78CG28E	—	ASM75
µPD7528ACU	EVAKIT-7500B	EV7528	—	—	—	ASM75
µPD75CG28E	EVAKIT-7500B	EV7528	—	—	—	ASM75
µPD7533C	EVAKIT-7500B	EV7533	—	µPD75CG33E	—	ASM75
µPD7533CU	EVAKIT-7500B	EV7533	—	—	—	ASM75
µPD7533G-22	EVAKIT-7500B	EV7533	—	—	—	ASM75
µPD75CG33E	EVAKIT-7500B	EV7533	—	—	—	ASM75
µPD7537AC	EVAKIT-7500B	EV7528	—	µPD75CG38E	—	ASM75
µPD7537ACU	EVAKIT-7500B	EV7528	—	—	—	ASM75
µPD7538AC	EVAKIT-7500B	EV7528	—	µPD75CG38E	—	ASM75
µPD7538ACU	EVAKIT-7500B	EV7528	—	—	—	ASM75
µPD75CG38E	EVAKIT-7500B	EV7528	—	—	—	ASM75
µPD7554CS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P54CS	PA-75P54CS	ASM75
µPD7554G	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P54G	PA-75P54CS	ASM75
µPD7554ACS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P54CS	PA-75P54CS	ASM75
µPD7554AG	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P54G	PA-75P54CS	ASM75
µPD75P54CS	EVAKIT-7500B	EV7554A	—	—	—	ASM75
µPD75P54G	EVAKIT-7500B	EV7554A	—	—	—	ASM75
µPD7556CS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P56CS	PA-75P56CS	ASM75
µPD7556G	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P56G	PA-75P56CS	ASM75
µPD7556ACS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P56CS	PA-75P56CS	ASM75
µPD7556AG	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P56G	PA-75P56CS	ASM75
µPD75P56CS	EVAKIT-7500B	EV7554A	—	—	—	ASM75
µPD75P56G	EVAKIT-7500B	EV7554A	—	—	—	ASM75
µPD7564CS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P64CS	PA-75P54CS	ASM75
µPD7564G	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P64G	PA-75P54CS	ASM75
µPD7564ACS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P64CS	PA-75P54CS	ASM75
µPD7564AG	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P64G	PA-75P54CS	ASM75
µPD75P64CS	EVAKIT-7500B	EV7554A	—	—	—	ASM75
µPD75P64G	EVAKIT-7500B	EV7554A	—	—	—	ASM75
µPD7566CS	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P66CS	PA-75P56CS	ASM75
µPD7566G	EVAKIT-7500B	EV7554A	SE-7554A	µPD75P66G	PA-75P56CS	ASM75

* Required Tools

μPD75XX Series Development Tools Selection Guide (cont)

Part Number (Note 1)	Emulator*	Add-on Board*	System Evaluation Board	EPROM/OTP Device	PG-1500 Adapter (Note 2)	Absolute Assembler (Note 3)
μPD7566ACS	EVAKIT-7500B	EV7554A	SE-7554A	μPD75P66CS	PA-75P56CS	ASM75
μPD7566AG	EVAKIT-7500B	EV7554A	SE-7554A	μPD75P66G	PA-75P56CS	ASM75
μPD75P66CS	EVAKIT-7500B	EV7554A	—	—	—	ASM75
μPD75P66G	EVAKIT-7500B	EV7554A	—	—	—	ASM75

* Required Tools

Notes:

(1) Packages:

Package	Description
C	40-pin plastic DIP (μPD7507/07H/08/08H) 42-pin plastic DIP (μPD7527A/28A/33/37A/38A)
CS	20-pin plastic shrink DIP (μPD7554/54A/P54/64/64A/P64) 24-pin plastic shrink DIP (μPD7556/56A/P56/66/66A/P66)
CU	40-pin plastic shrink DIP (μPD7507/07H/08/08H) 42-pin plastic shrink DIP (μPD7527A/28A/33/37A/38A)
E	40-pin ceramic piggy-back DIP (μPD75CG08/08H) 42-pin ceramic piggy-back DIP (μPD75CG28/33/38)
G	20-pin plastic SO (μPD7554/54A/P54/64/64A/P64) 24-pin plastic SO (μPD7556/56A/P56/66/66A/P66)
G-00	52-pin plastic QFP
G-12	64-pin plastic QFP (μPD7502/03)
G-22	44-pin plastic QFP

(2) By using the specified adapter, the PG-1500 EPROM programmer can be used to program the OTP device.

(3) The ASM75 Absolute Assembler is provided to run under the MS-DOS® operating system. (ASM75-D52).

MS-DOS is a registered trademark of Microsoft Corporation.

μPD75XXX Series Development Tools Selection Guide

Part Number (Note 5)	Emulator*	Emulation Probe*	Optional Socket Adapter (Note 1)	EPROM/OTP Device (Note 2)	Relocatable Assembler (Note 3)	Structured Assembler (Note 4)
μPD75004CU	IE-75000-R	EP-75008CU-R	—	μPD75P008CU	RA75X	ST75X
μPD75006GB-3B4	IE-75000-R	EP-75008GB-R	EV-9200G-44	μPD75P008GB	RA75X	ST75X
μPD75006CU	IE-75000-R	EP-75008CU-R	—	μPD75P008CU	RA75X	ST75X
μPD75006GB-3B4	IE-75000-R	EP-75008GB-R	EV-9200G-44	μPD75P008GB	RA75X	ST75X
μPD75008CU	IE-75000-R	EP-75008CU-R	—	μPD75P008CU	RA75X	ST75X
μPD75008GB-3B4	IE-75000-R	EP-75008GB-R	EV-9200G-44	μPD75P008GB	RA75X	ST75X
μPD75P008CU	IE-75000-R	EP-75008CU-R	—	—	RA75X	ST75X
μPD75P008GB-3B4	IE-75000-R	EP-75008GB-R	EV-9200G-44	—	RA75X	ST75X
μPD75028CW	IE-75000-R	EP-75028CW-R	—	μPD75P036CW	RA75X	ST75X
μPD75028GC-AB8	IE-75000-R	EP-75028GC-R	EV-9200GC-64	μPD75P036GC	RA75X	ST75X
μPD75P036CW	IE-75000-R	EP-75028CW-R	—	—	RA75X	ST75X
μPD75P036GC-AB8	IE-75000-R	EP-75028GC-R	EV-9200GC-64	—	RA75X	ST75X
μPD75048CW	IE-75000-R	EP-75028CW-R	—	μPD75P056CW	RA75X	ST75X
μPD75048GC-AB8	IE-75000-R	EP-75028GC-R	EV-9200GC-64	μPD75P056GC	RA75X	ST75X
μPD75P056CW	IE-75000-R	EP-75028CW-R	—	—	RA75X	ST75X
μPD75P056GC-AB8	IE-75000-R	EP-75028GC-R	EV-9200GC-64	—	RA75X	ST75X
μPD75104CW	IE-75000-R	EP-75108CW-R	—	μPD75P108CW/DW μPD75P116CW	RA75X	ST75X
μPD75104G-1B	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P108G/GF μPD75P116GF	RA75X	ST75X
μPD75104GF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P108G/GF μPD75P116GF	RA75X	ST75X
μPD75104AGC-AB8	IE-75000-R	EP-75108AGC-R	EV-9200GC-64	—	RA75X	ST75X
μPD75106CW	IE-75000-R	EP-75108CW-R	—	μPD75P108CW/DW μPD75P116CW	RA75X	ST75X
μPD75106G-1B	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P108G/GF μPD75P116GF	RA75X	ST75X
μPD75106GF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P108G/GF μPD75P116GF	RA75X	ST75X
μPD75108AG-22	IE-75000-R	EP-75108AGC-R	EV-9200GC-64	—	RA75X	ST75X
μPD75108AGC-AB8	IE-75000-R	EP-75108AGC-R	EV-9200GC-64	—	RA75X	ST75X
μPD75108CW	IE-75000-R	EP-75108CW-R	—	μPD75P108CW/DW μPD75P116CW	RA75X	ST75X
μPD75108G-1B	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P108G/GF μPD75P116GF	RA75X	ST75X
μPD75108GF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P108G/GF μPD75P116GF	RA75X	ST75X
μPD75P108BCW	IE-75000-R	EP-75108CW-R	—	—	RA75X	ST75X
μPD75P108BGF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	—	RA75X	ST75X
μPD75P108CW	IE-75000-R	EP-75108CW-R	—	—	RA75X	ST75X
μPD75P108DW	IE-75000-R	EP-75108CW-R	—	—	RA75X	ST75X
μPD75P108G-1B	IE-75000-R	EP-75108GF-R	EV-9200G-64	—	RA75X	ST75X
μPD75112CW	IE-75000-R	EP-75108CW-R	—	μPD75P116CW	RA75X	ST75X
μPD75112GF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P116GF	RA75X	ST75X
μPD75116CW	IE-75000-R	EP-75108CW-R	—	μPD75P116CW	RA75X	ST75X
μPD75116GF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	μPD75P116GF	RA75X	ST75X
μPD75P116CW	IE-75000-R	EP-75108CW-R	—	—	RA75X	ST75X
μPD75P116GF-3BE	IE-75000-R	EP-75108GF-R	EV-9200G-64	—	RA75X	ST75X
μPD75206CW	IE-75000-R	EP-75216ACW-R	—	μPD75P216ACW	RA75X	ST75X
μPD75206G-1B	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
μPD75206BGF-3BE	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
μPD75208CW	IE-75000-R	EP-75216ACW-R	—	μPD75P216ACW	RA75X	ST75X

* Required Tools

µPD75XXX Series Development Tools Selection Guide (cont)

Part Number (Note 5)	Emulator*	Emulation Probe*	Optional Socket Adapter (Note 1)	EPROM/OTP Device (Note 2)	Relocatable Assembler (Note 3)	Structured Assembler (Note 4)
µPD75208G-1B	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75208GF-3BE	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75CG208E	IE-75000-R	EP-75216ACW-R	—	—	RA75X	ST75X
µPD75CG208EA	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75212ACW	IE-75000-R	EP-75216ACW-R	—	µPD75P216ACW	RA75X	ST75X
µPD75212AGF-3BE	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75216ACW	IE-75000-R	EP-75216ACW-R	—	µPD75P216ACW	RA75X	ST75X
µPD75216AGF	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75CG216AE	IE-75000-R	EP-75216ACW-R	—	—	RA75X	ST75X
µPD75CG216AEA	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75P216ACW	IE-75000-R	EP-75216ACW-R	—	µPD75P216ACW	RA75X	ST75X
µPD75268CW	IE-75000-R	EP-75216ACW-R	—	µPD75P216ACW	RA75X	ST75X
µPD75268GF-3BE	IE-75000-R	EP-75216AGF-R	EV-9200G-64	—	RA75X	ST75X
µPD75304GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	µPD75P308GF/K	RA75X	ST75X
µPD75306GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	µPD75P308GF/K	RA75X	ST75X
µPD75308GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	µPD75P308GF/K	RA75X	ST75X
µPD75P308GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	—	RA75X	ST75X
µPD75P308K	IE-75000-R	EP-75308GF-R	EV-9200G-80	—	RA75X	ST75X
µPD75312GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	µPD75P316GF	RA75X	ST75X
µPD75316GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	µPD75P316GF	RA75X	ST75X
µPD75P316GF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	—	RA75X	ST75X
µPD75P316AGF-3B9	IE-75000-R	EP-75308GF-R	EV-9200G-80	—	RA75X	ST75X
µPD75P316AK	IE-75000-R	EP-75308GF-R	EV-9200G-80	—	RA75X	ST75X
µPD75328GC-3B9	IE-75000-R	EP-75328GC-R	EV-9200GC-80	µPD75P328GC	RA75X	ST75X
µPD75P328GC-3B9	IE-75000-R	EP-75328GC-R	EV-9200GC-80	—	RA75X	ST75X
µPD75402AC	IE-75000-R	EP-75402C-R	—	µPD75P402C	RA75X	ST75X
µPD75402ACT	IE-75000-R	EP-75402C-R	—	µPD75P402CT	RA75X	ST75X
µPD75402AGB-3B4	IE-75000-R	EP-75402GB-R	EV-9200G-44	µPD75P402GB	RA75X	ST75X
µPD75P402C	IE-75000-R	EP-75402C-R	—	—	RA75X	ST75X
µPD75P402CT	IE-75000-R	EP-75402C-R	—	—	RA75X	ST75X
µPD75P402GB-3B4	IE-75000-R	EP-75402GB-R	EV-9200G-44	—	RA75X	ST75X
µPD75512GF-3B9	IE-75000-R	EP-75516GF-R	EV-9200G-80	µPD75P516GF/K	RA75X	ST75X
µPD75516GF-3B9	IE-75000-R	EP-75516GF-R	EV-9200G-80	µPD75P516GF/K	RA75X	ST75X
µPD75P516GF-3B9	IE-75000-R	EP-75516GF-R	EV-9200G-80	—	RA75X	ST75X
µPD75P516K	IE-75000-R	EP-75516GF-R	EV-9200G-80	—	—	—

Notes:

(1) The EV-9200G-XX is an LCC socket with the footprint of the flat package. One unit is supplied with the probe. Additional units are available as replacement parts in sets of five.

(2) All EPROM/OTP devices can be programmed using the NEC PG-1500. Refer to the PG-1500 Programming Socket Adapter Selection Guide for the appropriate socket adapter.

(3) The RA75X relocatable assembler package is provided for the following operating systems:
RA75X-D52 (MS-DOS®)
RA75X-VVT1 (VAX/VMS®)

(4) The ST75X structures assembler preprocessor is provided with RA75X.

(5) Packages:

Package	Description
C	28-pin plastic DIP
CT	28-pin plastic shrink DIP
CU	42-pin plastic shrink DIP
CW	64-pin plastic shrink DIP
DW	64-pin ceramic shrink DIP with window
E	64-pin ceramic piggy-back shrink DIP
EA	64-pin ceramic piggy-back QFP
G-1B	64-pin plastic QFP (2.05 mm thick)
G-22	64-pin plastic QFP (1.55 mm thick)
GB-3B4	44-pin plastic QFP
GC-AB8	64-pin plastic QFP (2.55 mm thick)
GC-3B9	80-pin plastic QFP
GF-3BE	64-pin plastic QFP (2.77 mm thick)
GF-3B9	80-pin plastic QFP
K	80-pin ceramic LCC

MS-DOS is a registered trademark of Microsoft Corporation.

VAX and VMS are registered trademarks of Digital Equipment Corporation.

µPD78XX Series Development Tools Selection Guide**

Part Number (Note 1)	Emulator*	Emulation Probe*	EPROM/OTP Device	PG-1500 Adapter (Note 2)	Relocatable Assembler (Note 9)	C Compiler (Note 9)
µPD78C10CW	IE-78C11-M	EV-9001-64 (Note 3)	—	—	RA87	CC87
µPD78C10G-36	IE-78C11-M	(Note 4)	—	—	RA87	CC87
µPD78C10G-1B	IE-78C11-M	(Note 5)	—	—	RA87	CC87
µPD78C10GF-3BE	IE-78C11-M	(Note 5)	—	—	RA87	CC87
µPD78C10L	IE-78C11-M	(Note 5)	—	—	RA87	CC87
µPD78C10ACW	IE-78C11-M (Note 8)	EV-9001-64 (Note 3)	—	—	RA87	CC87
µPD78C10AGQ-36	IE-78C11-M (Note 8)	(Note 4)	—	—	RA87	CC87
µPD78C10AGF-3BE	IE-78C11-M (Note 8)	(Note 5)	—	—	RA87	CC87
µPD78C10AL	IE-78C11-M (Note 8)	(Note 5)	—	—	RA87	CC87
µPD78C11CW	IE-78C11-M	EV-9001-64 (Note 3)	µPD78CP14CW/DW	PA-78CP14CW	RA87	CC87
µPD78C11G-36	IE-78C11-M	(Note 4)	µPD78CP14G-36/R µPD78CP14E	PA-78CP14GQ	RA87	CC87
µPD78C11G-1B	IE-78C11-M	(Note 5)	µPD78CP14GF-3BE	PA-78CP14GF	RA87	CC87
µPD78C11GF-3BE	IE-78C11-M	(Note 5)	µPD78CP14GF-3BE	PA-78CP14GF	RA87	CC87
µPD78C11L	IE-78C11-M	(Note 5)	µPD78CP14L	PA-78CP14L	RA87	CC87
µPD78C11ACW	IE-78C11-M (Note 7)	EV-9001-64 (Note 3)	µPD78CP14CW/DW (Note 6)	PA-78CP14CW	RA87	CC87
µPD78C11AGQ-36	IE-78C11-M (Note 7)	(Note 4)	µPD78CP14G-36/R (Note 6)	PA-78CP14GQ	RA87	CC87
µPD78C11AGF-3BE	IE-78C11-M (Note 7)	(Note 5)	µPD78CP14GF-3BE (Note 6)	PA-78CP14GF	RA87	CC87
µPD78C11AL	IE-78C11-M (Note 7)	(Note 5)	µPD78CP14L (Note 6)	PA-78CP14L	RA87	CC87
µPD78C12ACW	IE-78C11-M (Note 7)	EV-9001-64 (Note 3)	µPD78CP14CW/DW (Note 6)	PA-78CP14CW	RA87	CC87
µPD78C12AGQ-36	IE-78C11-M (Note 7)	(Note 4)	µPD78CP14G-36/R (Note 6)	PA-78CP14GQ	RA87	CC87
µPD78C12AGF-3BE	IE-78C11-M (Note 7)	(Note 5)	µPD78CP14GF-3BE (Note 6)	PA-78CP14GF	RA87	CC87
µPD78C12AL	IE-78C11-M (Note 7)	(Note 5)	µPD78CP14L (Note 6)	PA-78CP14L	RA87	CC87
µPD78C14CW	IE-78C11-M	EV-9001-64 (Note 3)	µPD78CP14CW/DW	PA-78CP14CW	RA87	CC87
µPD78C14G-36	IE-78C11-M	(Note 4)	µPD78CP14G-36/R µPD78CG14E	PA-78CP14GQ —	RA87	CC87
µPD78C14G-1B	IE-78C11-M	(Note 5)	µPD78CP14GF	PA-78CP14GF	RA87	CC87
µPD78C14GF-3BE	IE-78C11-M	(Note 5)	µPD78CP14GF	PA-78CP14GF	RA87	CC87
µPD78C14L	IE-78C11-M	(Note 5)	µPD78CP14L	PA-78CP14L	RA87	CC87
µPD78C14AG-AB8	IE-78C11-M (Note 7)	(Note 5)	—	—	RA87	CC87
µPD78CG14E (Note 8)	IE-78C11-M	(Note 4)	—	—	RA87	CC87
µPD78CP14CW	IE-78C11-M	EV-9001-64 (Note 3)	—	PA-78CP14CW	RA87	CC87
µPD78CP14DW	IE-78C11-M	EV-9001-64 (Note 3)	—	PA-78CP14CW	RA87	CC87

* Required Tools

**For all µPD78C1X devices, you may use the DDK-78C10 for evaluation purposes.

µPD78XX Series Development Tools Selection Guide** (cont)

Part Number (Note 1)	Emulator*	Emulation Probe*	EPROM/OTP Device	PG-1500 Adapter (Note 2)	Relocatable Assembler (Note 9)	C Compiler (Note 9)
µPD78CP14G-36	IE-78C11-M	(Note 4)	—	PA-78CP14GQ	RA87	CC87
µPD78CP14GF-3BE	IE-78C11-M	(Note 5)	—	PA-78CP14GF	RA87	CC87
µPD78CP14L	IE-78C11-M	(Note 5)	—	PA-78CP14L	RA87	CC87
µPD78CP14R	IE-78C11-M	(Note 4)	—	PA-78CP14GQ	RA87	CC87

* Required Tools

**For all µPD78C1X devices, you may use the DDK-78C10 for evaluation purposes.

Notes:

(1) Packages:

Package	Description
CW	64-pin plastic shrink DIP
DW	64-pin ceramic shrink DIP with window
E	64-pin ceramic piggy-back QUIP
G-1B	64-pin plastic QFP (Resin Thickness: 2.05 mm)
G-36	64-pin plastic QUIP
G-AB8	64-pin plastic QFP (Interpin Pitch: 0.8 mm)
GF-3BE	64-pin plastic QFP (Resin Thickness: 2.7 mm)
GQ-36	64-pin plastic QUIP
L	68-pin PLCC
R	64-pin ceramic QUIP with window

(2) By using the specified adapter, the PG-1500 EPROM programmer can be used to program the EPROM/OTP device.

(3) 64-pin shrink DIP adapter which plugs into the EP-7811HGQ emulation probe supplied with each IE.

(4) The emulation probe for the 64-pin QUIP package (EP-7811HGQ) is supplied with the IE.

(5) No emulation probe available.

(6) The µPD78CP14 EPROM/OTP devices do not have pull-up resistors on ports A, B, and C.

(7) The IE-78C11-M can be used by replacing the µPD78C10G-36 with a µPD78C10AGQ-36. However, it will not be able to emulate the optional pull-up resistors on ports A, B, and C.

(8) The µPD78CG14E is a piggy-back EPROM device in a ceramic QUIP package. It accepts 27C256 and 27C256A EPROMS.

(9) The following relocatable assemblers and C compilers are available:

RA87-D52 RA87-VVT1	(MS-DOS®) (VAX/VMS®)	Relocatable assemblers for 78XX series
CCMSD-I5DD-87 CCVMS-OT16-87 CCUNX-OT16-87	(MS-DOS) (VAX/VMS) (VAX/UNIX®) 4.2 BSD or Ultrix®)	C Compilers for 78XX Series

MS-DOS is a registered trademark of Microsoft Corporation.
VAX, VMS and Ultrix are registered trademarks of Digital Equipment Corporation.
UNIX is a trademark of AT&T Bell Laboratories.

µPD782XX Series Development Tools Selection Guide (Note 1)

Part Number (Note 2)	Evaluation Kit (Note 3	Designer Kit (Note 4)	Emulator Kit (Note 5)	Low-End Emulator	Emulation System	Emulation Probe	Device (Note 6)
µPD78213CW	EK-78K2-21X	DK-78K2-21XCW	IK-78K2-21XCW	EB-78210-PC	IE-78210-R	EP-78210CW-R	—
µPD78213GJ-5BJ	EK-78K2-21X	DK-78K2-21XGJ	IK-78K2-21XGJ	EB-78210-PC	IE-78210-R	EP-78210GJ-R (7)	—
µPD78213GQ-36	EK-78K2-21X	DK-78K2-21XGQ	IK-78K2-21XGQ	EB-78210-PC	IE-78210-R	EP-78210GQ-R	—
µPD78213L	EK-78K2-21X	DK-78K2-21XL	IK-78K2-21XL	EB-78210-PC	IE-78210-R	EP-78210L-R	—
µPD78214CW	EK-78K2-21X	DK-78K2-21XCW	IK-78K2-21XCW	EB-78210-PC	IE-78210-R	EP-78210CW-R	µPD78P214CW/DW
µPD78214GJ-5BJ	EK-78K2-21X	DK-78K2-21XGJ	IK-78K2-21XGJ	EB-78210-PC	IE-78210-R	EP-78210GJ-R (7)	µPD78P214GJ
µPD78214GQ-36	EK-78K2-21X	DK-78K2-21XGQ	IK-78K2-21XGQ	EB-78210-PC	IE-78210-R	EP-78210GQ-R	µPD78P214GQ
µPD78214L	EK-78K2-21X	DK-78K2-21XL	IK-78K2-21XL	EB-78210-PC	IE-78210-R	EP-78210L-R	µPD78P214L
µPD78P214CW	EK-78K2-21X	DK-78K2-21XCW	IK-78K2-21XCW	EB-78210-PC	IE-78210-R	EP-78210CW-R	—
µPD78P214DW	EK-78K2-21X	DK-78K2-21XCW	IK-78K2-21XCW	EB-78210-PC	IE-78210-R	EP-78210CW-R	—
µPD78P214GJ-5BJ	EK-78K2-21X	DK-78K2-21XGJ	IK-78K2-21XGJ	EB-78210-PC	IE-78210-R	EP-78210GJ-R (7)	—
µPD78P214GQ-36	EK-78K2-21X	DK-78K2-21XGQ	IK-78K2-21XGQ	EB-78210-PC	IE-78210-R	EP-78210GQ-R	—
µPD78P214L	EK-78K2-21X	DK-78K2-21XL	IK-78K2-21XL	EB-78210-PC	IE-78210-R	EP-78210L-R	—
µPD78220GJ-5BG	EK-78K2-22X	DK-78K2-22XGJ	IK-78K2-22XGJ	EB-78220-PC	IE-78220-R	EP-78220GJ-R (8)	—
µPD78220L	EK-78K2-22X	DK-78K2-22XL	IK-78K2-22XL	EB-78220-PC	IE-78220-R	EP-78220L-R	—
µPD78224GJ-5BG	EK-78K2-22X	DK-78K2-22XGJ	IK-78K2-22XGJ	EB-78220-PC	IE-78220-R	EP-78220GJ-R (8)	µPD78P224GJ
µPD78224L	EK-78K2-22X	DK-78K2-22XL	IK-78K2-22XL	EB-78220-PC	IE-78220-R	EP-78220L-R	µPD78P224L
µPD78P224GJ-5BG	EK-78K2-22X	DK-78K2-22XGJ	IK-78K2-22XGJ	EB-78220-PC	IE-78220-R	EP-78220GJ-R (8)	—
µPD78P224L	EK-78K2-22X	DK-78K2-22XL	IK-78K2-22XL	EB-78220-PC	IE-78220-R	EP-78220L-R	—
µPD78233GC-3B9	EK-78K2-23X	DK-78K2-23XGC	IK-78K2-23XGC	EB-78230-PC	IE-78230-R	EP-78230GC-R	—
µPD78233GJ-5BG	EK-78K2-23X	DK-78K2-23XGJ	IK-78K2-23XGJ	EB-78230-PC	IE-78230-R	EP-78230GJ-R	—
µPD78233LQ	EK-78K2-23X	DK-78K2-23XL	IK-78K2-23XL	EB-78230-PC	IE-78230-R	EP-78230LQ-R	—
µPD78234GC-3B9	EK-78K2-23X	DK-78K2-23XGC	IK-78K2-23XGC	EB-78230-PC	IE-78230-R	EP-78230GC-R	µPD78P238GC
µPD78234GJ-5BG	EK-78K2-23X	DK-78K2-23XGJ	IK-78K2-23XGJ	EB-78230-PC	IE-78230-R	EP-78230GJ-R	µPD78P238GJ/KF
µPD78234LQ	EK-78K2-23X	DK-78K2-23XL	IK-78K2-23XL	EB-78230-PC	IE-78230-R	EP-78230LQ-R	µPD78P238LQ
µPD78P238GC-3B9	EK-78K2-23X	DK-78K2-23XGC	IK-78K2-23XGC	EB-78230-PC	IE-78230-R	EP-78230GC-R	—
µPD78P238GJ-5BG	EK-78K2-23X	DK-78K2-23XGJ	IK-78K2-23XGJ	EB-78230-PC	IE-78230-R	EP-78230GJ-R	—
µPD78P238KF	EK-78K2-23X	DK-78K2-23XGJ	IK-78K2-23XGJ	EB-78230-PC	IE-78230-R	EP-78230GJ-R	—
µPD78P238LQ	EK-78K2-23X	DK-78K2-23XL	IK-78K2-23XL	EB-78230-PC	IE-78230-R	EP-78230LQ-R	—

Notes:

1. The following software packages are available for the µPD782XX Series:

   RA78K2 relocatable assembler package
   RA78K2-D52 (MS-DOS®)
   RA78K2-VVT1 (VAX®/VMS®)
   ST78K2 Structured assembler preprocessor
   Provided with RA78K2
   CC782XX C Compiler package
   CCMSD-I5DD-782XX (MS-DOS®)

(2) Packages:

Package	Description
CW	64-pin plastic shrink DIP
DW	64-pin ceramic shrink DIP with window
GC-3B9	80-pin plastic QFP
GJ-5BG	94-pin plastic QFP
GJ-5BJ	74-pin plastic QFP
GQ-36	64-pin plastic QUIP
KF	94-pin ceramic LCC with window
L	68-pin PLCC (µPD78213/214/P214L) 84-pin PLCC (µPD78220/224/P224L)
LQ	84-pin PLCC

MS-DOS is a registered trademark of Microsoft Corporation.

VAX and VMS are registered trademarks of Digital Equipment Corporation.

(3) The µPD782XX Evaluation Kit contains the appropriate DDB-78K2-2XX evaluation board for the part selected, the RA78K2 Relocatable Assembler Package, and the ST78K2 Structured Assembler Preprocessor.

(4) The µPD782XX Designer Kit contains the appropriate EB-782XX-PC low-end emulator and emulation probe for the part selected, the RA78K2 Relocatable Assembler Package, and the ST78K2 Structured Assembler Preprocessor.

(5) The µPD782XX Emulator Kit contains the appropriate IE-782XX system and emulation probe for the part selected, the RA78K2 Relocatable Assembler Package, and the ST78K2 Structured Assembler Preprocessor.

(6) All EPROM/OTP devices can be programmed using the NEC PG-1500. Refer to the PG-1500 Programming Socket Adapter Selection Guide for the appropriate programming adapter.

(7) The EP-78210GJ-R emulation probe is shipped with one EV-9200G-74, a 74-pin LCC socket with the footprint of the QFP package. Additional sockets are available as replacement parts in sets of five.

(8) The EP-78220GJ-R emulation probe is shipped with one EV-9200G-94, a 94-pin LCC socket with the footprint of the QFP package. Additional sockets are available as replacement parts in sets of five.

µPD783XX Series Development Tools Selection Guide (Note 10)

Part Number (Note 1)	Emulator*	Emulation Probe*	EPROM/OTP Device	PG-1500 Adapter (Note 2)	Relocatable Assembler (Note11)	Structured Assembler (Note 12)	C Compiler (Note 13)
µPD78310ACW	IE-78310A-R	(Note 3)	—	—	RA78K3	ST78K3	CC7831X (Note 5)
µPD78310AGF-3BE	IE-78310A-R	EP-78310GF (Note 6)	—	—	RA78K3	ST78K3	CC7831X (Note 5)
µPD78310AGQ-36	IE-78310A-R	(Note 4)	—	—	RA78K3	ST78K3	CC7831X (Note 5)
µPD78310AL	IE-78310A-R	EP-78310L	—	—	RA78K3	ST78K3	CC7831X (Note 5)
µPD78312ACW	IE-78310A-R	(Note 3)	µPD78P312ACW/DW	PA-78P312CW	RA78K3	ST78K3	CC7831X (Note 5)
µPD78312AGF-3BE	IE-78310A-R	EP-78310GF (Note 6)	µPD78P312AGF-3BE	PA-78P312GF	RA78K3	ST78K3	CC7831X (Note 5)
µPD78312AGQ-36	IE-78310A-R	(Note 4)	µPD78P312AGQ/RQ	PA-78P312GQ	RA78K3	ST78K3	CC7831X (Note 5)
µPD78312AL	IE-78310A-R	EP-78310L	µPD78P312AL	PA-78P312L	RA78K3	ST78K3	CC7831X (Note 5)
µPD78P312ACW	IE-78310A-R	(Note 3)	—	PA-78P312CW	RA78K3	ST78K3	CC7831X (Note 5)
µPD78P312ADW	IE-78310A-R	(Note 3)	—	PA-78P312CW	RA78K3	ST78K3	CC7831X (Note 5)
µPD78P312AGF-3BE	IE-78310A-R	EP-78310GF (Note 6)	—	PA-78P312GF	RA78K3	ST78K3	CC7831X (Note 5)
µPD78P312AGQ-36	IE-78310A-R	(Note 4)	—	PA-78P312GQ	RA78K3	ST78K3	CC7831X (Note 5)
µPD78P312AL	IE-78310A-R	EP-78310L	—	PA-78P312L	RA78K3	ST78K3	CC7831X (Note 5)
µPD78P312AR	IE-78310A-R	(Note 4)	—	PA-78P312GQ	RA78K3	ST78K3	CC7831X (Note 5)
µPD78320GJ-5BJ	IE-78320-R	EP-78320GJ-R (Note 7)	—	—	RA78K3	ST78K3	CC7832X
µPD78320L	IE-78320-R	EP-78320L-R	—	—	RA78K3	ST78K3	CC7832X
µPD78322GJ-5BJ	IE-78320-R	EP-78320GJ-R (Note 7)	µPD78P322GJ µPD78P322KD	PA-78P322GJ PA-78P322KD	RA78K3	ST78K3	CC7832X
µPD78322L	IE-78320-R	EP-78320L-R	µPD78P322L µPD78P322KC	PA-78P322L PA-78P322KC	RA78K3	ST78K3	CC7832X
µPD78P322GJ	IE-78320-R	EP-78320GJ-R (Note 7)	—	PA-78P322GJ	RA78K3	ST78K3	CC7832X
µPD78P322KC	IE-78320-R	EP-78320L-R	—	PA-78P322KC	RA78K3	ST78K3	CC7832X
µPD783P322KD	IE-78320-R	EP-78320GJ-R (Note 7)	—	PA-78P322KD	RA78K3	ST78K3	CC7832X
µPD78P322L	IE-78320-R	EP-78320L-R	—	PA-78P322L	RA78K3	ST78K3	CC7832X
µPD71P301GF-3BE	IE-78320-R	—	—	PA-71P301GF	—	—	—
µPD71P301GQ-36	IE-78320-R	—	—	PA-71P301GQ	—	—	—
µPD71P301KA (Note 8)	IE-78320-R	—	—	PA-71P301KA	—	—	—
µPD71P301KB (Note 9)	IE-78320-R	—	—	PA-71P301KB	—	—	—
µPD71P301L	IE-78320-R	—	—	PA-71P301L	—	—	—
µPD71P301RQ	IE-78320-R	—	—	PA-71P301GQ	—	—	—

* Required Tools

Notes:

(1) Packages:

Package	Description
CW	64-pin plastic shrink DIP
DW	64-pin ceramic shrink DIP with window
GF-3BE	64-pin plastic QFP (Resin Thickness: 2.7 mm)
GJ-5BJ	74-pin plastic QFP (20 mm × 20 mm)
GQ-36	64-pin plastic QUIP
KA	44-pin ceramic LCC with window
KB	64-pin ceramic LCC with window
KC	68-pin ceramic LCC with window
KD	74-pin ceramic LCC with window
L	44-pin PLCC (μPD71P301L) 68-pin PLCC (μPD78310A/312A/P312AL, μPD78320/322L)
R	64-pin ceramic QUIP with window
RQ	64-pin ceramic QUIP with window

(2) By using the specified adapter, the PG-1500 EPROM programmer can be used to program the EPROM/OTP device.

(3) The emulation probe for the 64-pin shrink DIP package (EP-78310CW) is supplied with the IE.

(4) The emulation probe for the 64-pin QUIP package (EP-78310GQ) is supplied with the IE.

(5) There are two C Compilers for the μPD7831X devices: CC7831X from NEC Electronics and one from Lattice Corporation.

(6) The EP-78310GF emulation probe is shipped with one EV-9200G-64, a 64-pin LCC socket with the footprint of the QFP package. Additional sockets are available as replacement parts in sets of five.

(7) The EP-78320GJ-R emulation probe is shipped with one EV-9200G-74, a 74-pin LCC socket with the footprint of the QFP package. Additional sockets are available as replacement parts in sets of five.

(8) Sockets for the μPD71P301KA (44-pin LCC package) are available from Yamaichi, Inc. (IC61-0444-030).

(9) Sockets for the μPD71P301KB (64-pin LCC package) are available from NEC Electronics (EV-9200G-64) in sets of five.

(10) The following evaluation boards are available for the μPD783XX series:

Part Number	Design/Development Boards	Evaluation Boards
μPD7831XA	—	DDK-78310A
μPD7832X	EB-78320-PC	—

(11) The following relocatable packages are available:

RA-78K3-D52	(MS-DOS®)	Relocatable assembler
RA-78K3-VVT1	(VAX/VMS®)	for 783XX series

(12) The ST78K3 structured assembler preprocessor is provided with RA78K3.

(13) The following C Compiler packages are available:

CCMSD-I5DD-7831X	(MS-DOS®)	For μPD7831X series
CCMSD-I5DD-7832X	(MS-DOS®)	For μPD7832X series

MS-DOS is a registered trademark of Microsoft Corporation.

VAX and VMS are registered trademarks of Digital Equipment Corporation.

PG-1500 Programming Adapters

Socket Adapters and Adapter Modules

Target Chip	Socket Adapter (Note 1)	Adapter Module (Note 2)
Standard 27XXX EPROM Devices		
µPD27256 (12.5 V)	—	027A Board
µPD27256 (21 V)	—	027A Board
µPD27C256	—	027A Board
µPD27C256A	—	027A Board
µPD27C512	—	027A Board
µPD27C1000	—	027A Board
µPD27C1001	—	027A Board
µPD27C10124	—	027A Board
µPD75XX Series Devices		
µPD75P54CS	PA-75P54CS	04A Board
µPD75P54G	PA-75P54CS	04A Board
µPD75P56CS	PA-75P54CS	04A Board
µPD75P56G	PA-75P54CS	04A Board
µPD75P64CS	PA-75P54CS	04A Board
µPD75P64G	PA-75P54CS	04A Board
µPD75P66CS	PA-75P56CS	04A Board
µPD75P66G	PA-75P56CS	04A Board
µPD75XXX Series Devices		
µPD75P008CU	PA-75P008CU	04A Board
µPD75P008GB	PA-75P008CU	04A Board
µPD75P036CW	PA-75P036CW	04A Board
µPD75P036GC	PA-75P036GC	04A Board
µPD75P108BCW	PA-75P108CW	04A Board
µPD75P108CW	PA-75P108CW	04A Board
µPD75P108DW	PA-75P108CW	04A Board
µPD75P108BGF	PA-75P108G PA-75P116GF	04A Board 04A Board
µPD75P108G	PA-75P108G PA-75P116GF	04A Board 04A Board
µPD75P116CW	PA-75P108CW	04A Board
µPD75P116GF	PA-75P108G PA-75P116GF	04A Board 04A Board
µPD75P216GF	PA-75P216ACW	04A Board
µPD75P308GF	PA-75P308GF	04A Board
µPD75P308K	PA-75P308K	04A Board
µPD75P316GF	PA-75P308GF	04A Board
µPD75P316AGF	PA-75P308GF	04A Board
µPD75P316AK	PA-75P308K	04A Board
µPD75P328GC	PA-75P328GC	04A Board
µPD75P402C	(Note 3)	027A Board
µPD75P402CT	PA-75P402CT	027A Board
µPD75P402GB	PA-75P402GB	027A Board
µPD75P516GF	PA-75P516GF	04A Board
µPD75P516K	PA-75P516K	04A Board

Target Chip	Socket Adapter (Note 1)	Adapter Module (Note 2)
µPD78XX Series Devices		
µPD78CP14CW	PA-78CP14CW	027A Board
µPD78CP14DW	PA-78CP14CW	027A Board
µPD78CP14G	PA-78CP14GQ	027A Board
µPD78CP14GF	PA-78CP14GF	027A Board
µPD78CP14L	PA-78CP14L	027A Board
µPD78CP14R	PA-78CP14GQ	027A Board
µPD78XXX Series Devices		
µPD71P301GF	PA-71P301GF	027A Board
µPD71P301GQ	PA-71P301GQ	027A Board
µPD71P301KA	PA-71P301KA	027A Board
µPD71P301KB	PA-71P301KB	027A Board
µPD71P301L	PA-71P301L	027A Board
µPD78P214CW	PA-78P214CW	027A Board
µPD78P214GJ	PA-78P214GJ	027A Board
µPD78P214GQ	PA-78P214GQ	027A Board
µPD78P214L	PA-78P214L	027A Board
µPD78P224GJ	PA-78P224GJ	027A Board
µPD78P224L	PA-78P224L	027A Board
µPD78P238GC	PA-78P238GC	027A Board
µPD78P238GJ	PA-78P238GJ	027A Board
µPD78P238KF	PA-78P238KF	027A Board
µPD78P238LQ	PA-78P238LQ	027A Board
µPD78P312ACW	PA-78P312CW	027A Board
µPD78P312ADW	PA-78P312CW	027A Board
µPD78P312AGF	PA-78P312GF	027A Board
µPD78P312AGQ	PA-78P312GQ	027A Board
µPD78P312AL	PA-78P312L	027A Board
µPD78P312AR	PA-78P312GQ	027A Board
µPD78P322GJ	PA-78P322GJ	027A Board
µPD78P322KC	PA-78P322KC	027A Board
µPD78P322KD	PA-78P322KD	027A Board
µPD78P322L	PA-78P322L	027A Board
V-Series Devices		
µPD70P322K	PA-70P322L	027A Board
Digital Signal Processors		
µPD77P56CR	PA-77P56C	04A Board
µPD77P56G	PA-77P56C	04A Board
µPD77P25C	PA-77P25C	027A Board
µPD77P25D	PA-77P25C	027A Board
µPD77P230R	PA-77P230R	027A Board

Notes:

(1) All socket adapters must be purchased separately.

(2) The 27A and 04A Adapter Modules are shipped with the PG-1500.

(3) The µPD75P402C does not require a programming socket adapter. It can be plugged directly into the 027A board.

CMOS Microprocessors

Device, µPD	Features	Data Bits	Clock (MHz)	# Package	Pins
70008A	*Z80 microprocessor	8	8	DIP	40
				QFP	44
				PLCC	44
70108	8088 compatible; enhanced	8/16	8 or 10	DIP	40
(V20)				Ceramic DIP	40
				QFP	52
				PLCC	44
70116	8086 compatible; enhanced	16	8 or 10	DIP	40
(V30)				Ceramic DIP	40
				QFP	52
				PLCC	44
70208	MS-DOS, V20 compatible CPU with peripherals	8/16	8 or 10	Ceramic PGA	68
(V40)				PLCC	68
				QFP	80
70216	MS-DOS, V30 compatible CPU with peripherals	16/16	8 or 10	PGA	68
(V50)				PLCC	68
				QFP	80
70616	32-bit; high-speed	16/32	16	PGA	68
(V60)					
70632	32-bit; high-speed	32/32	20/25	PGA	132
(V70)					
70832	32-bit; high-speed	32/32	25	Ceramic PGA	208
(V80)					
70136	Hardwired, enhanced V30	16	12 or 16	PGA	68
(V33)				PLCC	68
70236	V33 core-based; high-integration; DMA, serial I/O,	16	–	Ceramic PGA	132
(V53)	interrupt controller, etc.			QFP	120
70320	MS-DOS compatible; high-integration; DMA, serial I/O,	8/16	5 or 8	PLCC	84
(V25)	interrupt controller, etc.			QFP	94
70330	MS-DOS compatible; high-integration; DMA, serial I/O,	16	8	PLCC	84
(V35)	interrupt controller, etc.			QFP	94
70325	MS-DOS compatible; high-integration; high-speed DMA	8/16	8 or 10	PLCC	84
(V25+)				QFP	94
70335	MS-DOS compatible; high-integration; high-speed DMA	16	8 or 10	PLCC	84
(V35+)				QFP	94
70327	MS-DOS compatible; high-integration; software protection	8/16	8	PLCC	84
(V25 Software Guard)				QFP	94
70337	MS-DOS compatible; high-integration; software protection	16	8	PLCC	84
(V35 Software Guard)				QFP	94
79011	MS-DOS compatible; high-integration; real-time operating system	8/16	8	PLCC	84
(V25 RTOS)				QFP	94
79021	MS-DOS compatible; high-integration; real-time operating system	16	8	PLCC	84
(V35 RTOS)				QFP	94
70322	MS-DOS compatible; high-integration; 16K-byte ROM	8/16	8	PLCC	84
(V25 ROM)				QFP	94

# Plastic unless ceramic (or cerdip) is specified.

* For additional information, refer to 1987 Microcomputer Data Book.

CMOS Microprocessors (cont)

Device, µPD	Features	Data Bits	Clock (MHz)	* Package	Pins
70P322	MS-DOS compatible; high-integration; 16K-byte UVEPROM; V25 or V35 mode	8/16	8	Ceramic LCC	84
70332 (V35 ROM)	MS-DOS compatible; high-integration; 16K-byte ROM	16	8	PLCC QFP	84 94

NMOS and HMOS Microprocessors

Device, µPD	Features	Data Bits	Clock (MHz)	* Package	Pins
8085A	*8-bit microprocessor; NMOS or HMOS	8	5	DIP	40
8086	*16-bit microprocessor; HMOS	16	8	Cerdip	40
8088	*8-bit microprocessor; HMOS	8	8	Ceramic DIP	40

CMOS System Support Products

Device, µPD	Name	Data Bits	Clock (MHz)	# Package	Pins
71011	Clock Pulse Generator/Driver	–	20	DIP SOP	18 20
71037	Programmable DMA Controller	8	10	DIP QFP PLCC	40 40 44
71051	Serial Control Unit	8	8/10	DIP QFP PLCC	28 44 28
71054	Programmable Timer/Controller	8	8/10	DIP QFP PLCC	24 44 28
71055	Parallel Interface Unit	8	8/10	DIP QFP PLCC	40 44 44
71059	Interrupt Control Unit	8	8/10	DIP QFP PLCC	28 44 28
71071	DMA Controller	8/16	8/10	DIP Ceramic DIP QFP PLCC	48 48 52 52
71082	Transparent Latch	8	8	DIP SOP	20 20
71083	Transparent Latch	8	8	DIP SOP	20 20
71084	Clock Pulse Generator/Driver	–	25	DIP SOP	18 20
71086	Bus Buffer/Driver	8	8	DIP SOP	18 20
71087	Bus Buffer/Driver	8	8	DIP SOP	20 20

CMOS System Support Products (cont)

Device, μPD	Name	Data Bits	Clock (MHz)	# Package	Pins
71088	System Bus Controller	–	8/10	DIP SOP	20 20
82C43	*Input/Output Expander	–	5	DIP Skinny DIP	24 24

NMOS System Support Products

Device, μPD	Name	Data Bits	Clock (MHz)	# Package	Pins
8155H	*256 x 8 RAM; I/O ports and timer	8	3 or 5	DIP	40
8156H	*256 x 8 RAM; I/O ports and timer	8	3 or 5	DIP	40
8237A	*Programmable DMA Controller	8	5	DIP	40
8243	*Input/Output Expander	–	5	DIP	24
8251A	*Programmable Communications Interface	8	3/5	DIP	28
8253	*Programmable Internal Timer	8	5	DIP	24
8255A	*Programmable Peripheral Interface	8	5	DIP	40
8257	*Programmable DMA Controller	8	5	DIP	40
8259A	*Programmable Interrupt Controller	8	5	DIP	28
8279	*Programmable Keyboard/Display Interface	–	5	DIP	40

Communications Controllers

Device, µPD	Name	Description	Data Rate	# Package	Pins
7201A	Multiprotocol Serial Communications Controller	Dual full-duplex serial channels; four DMA channels; programmable interrupt vectors; asychronous COP and BOP support; NMOS	1 Mb/s	DIP Ceramic DIP	40 40
72001	CMOS, Advanced Multiprotocol Serial Communications Controller	Functional superset of 8530; 8086/V30 interface; two full-duplex serial channels; two digital phase-locked loops; two baud-rate generators per channel; loopback test mode; short frame and mark idle detection	2.5 Mb/s	DIP QFP PLCC	40 52 52
72002	CMOS, Advanced Multiprotocol Serial Communications Controller	Low-cost, single-channel version of 72001; software compatible; direct interface to 8237 DMA. Not included in 1989-1990 IPD Data Book; refer to 72002 data sheet.	2.5 Mb/s	DIP QFP PLCC	40 44 44
72103	CMOS, HDLC Controller	Single full-duplex serial channel; on-chip DMA Controller. Not included in 1989-1990 IPD Data Book; refer to 72103 data sheet	4 Mb/s	DIP PLCC	64 68

Graphics Controllers

Device, µPD	Name	Description	Drawing Rate	# Package	Pins
7220A	High-Performance Graphics Display Controller	General-purpose, high-integration controller; hardwired support for lines, arc/circles, rectangles, and graphics characters; 1024x1024 pixel display with four planes	500 ns/dot	Ceramic DIP	40
72020	Graphics Display Controller	CMOS 7220A with 2M video memory; dual-port RAM control; write-masking on any bit; enhanced external synch	500 ns/dot	DIP QFP	40 52
72120	Advanced Graphics Display Controller	High-speed graphics operations including paint, area fill, slant, arbitrary angle rotate, up to 16x enlargement and reduction; dual-port RAM control; CMOS	500 ns/dot	PLCC QFP	84 94
72123	Advanced Graphics Display Controller II	Enhanced 72120; expanded command set; improved painting performance; laser printer interface controls; CMOS	400 ns/dot	PLCC QFP	84 94

Advanced Compression/Expansion Engine

Device, µPD	Name	Description	# Package	Pins
72185	Advanced Compression/ Expansion Engine	High-speed CCITT Group 3/4 bit-map image compression/expansion (A4 test chart, 400 PPI x 400 LPI in under 1 second); 32K-pixel line length; 32-megabyte image memory; on-chip DMA and refresh timing generator; CMOS	SDIP PLCC	64 68

# Plastic unless ceramic (or cerdip) is specified.

Floppy-Disk Controllers

Device, µPD	Name	Description	Transfer Rate	# Package	Pins
765A/B	Floppy-Disk Controller	Industry-standard controller supporting IBM 3740 and IBM System 34 double-density format; enhanced 765B supports multitasking applications	500 kb/s	DIP	40
71065/66	Floppy-Disk Interface	Compatible with 765-family controllers and others; supports multiple data rates from 125 to 500 kb/s	500 kb/s	SOP SDIP	28 30
72064*	Floppy-Disk Controller	CMOS; All features of 72068 with complete AT register set. Pin compatible with WD 37C65/A/B but with higher performance DPLL and reliable multitasking operation	500 kb/s	PLCC QFP	44 52
72065/65B	CMOS Floppy-Disk Controller	100% 765A/B microcode compatible; compatible with 808x microprocessor families	500 kb/s	DIP PLCC QFP	40 44 52
72067	Floppy-Disk Controller	CMOS; 765A/B microcode compatible; clock generation/ switching circuitry; selectable write precompensation; digital phase-locked loop	500 kb/s	DIP QFP PLCC	48 52 52
72068	Floppy-Disk Controller	All features of the 72067 plus IBM-PC, PC/XT, PC/AT, or PS/2 style registers; 24-ma high-current drivers	500 kb/s	QFP PLCC	80 84
72069	Floppy-Disk Controller	All features of the 72067/68 with substitution of high-performance analog phase-locked loop for digital PLL	1 Mb/s	PLCC QFP	84 100

Hard-Disk Controllers

Device, µPD	Name	Description	Read/Write Clock	# Package	Pins
7261A/B	Hard-Disk Controller	Supports eight drives in SMD mode, four drives in ST506 mode; error correction and detection	23 MHz	Ceramic DIP	40
7262	Enhanced Small-Disk Interface (ESDI) Controller	Serial-mode ESDI compatible; controls up to seven drives; supports up to 80 heads; hard and soft-sector interfacing	18 MHz	Ceramic DIP	40
72061	CMOS Hard-Disk Controller	Supports SMD/SMD-E and ST506/412 type drives	24 MHz	DIP QFP PLCC	40 52 52
72111	Small Computer System Interface (SCSI) Controller	Selectable 8/16 data bus width; 16 high-level commands for reduced CPU load; single-command automatic execution; 5-Mb sync/async; CMOS	16 MHz	SDIP QFP PLCC	64 74 68

* Not included in 1989-90 IPD Data Book; refer to 72064 Data Sheet.

DSP and Speech Products

Digital Signal Processors

Device, µPD	Description	Instruction Cycle (ns)	Instruction ROM (Bits)	Data ROM (Bits)	Data RAM (Bits)	# Package	Pins
7720A	16-bit, fixed-point DSP; NMOS	244	512 x 23	510 x 13	128 x 16	DIP	28
						PLCC	44
77C20A	16-bit, fixed-point DSP; CMOS	244	512 x 23	510 x 13	128 x 16	DIP	28
						PLCC	28
						PLCC	44
77P20	16-bit, fixed-point DSP; CMOS	244	512 x 23	510 x 13	128 x 16	Cerdip	28
			UVEPROM	UVEPROM			
77C25	16-bit, fixed-point DSP; CMOS	122	2048 x 24	1024 x 16	256 x 16	DIP	28
						PLCC	44
						SOP	32
77P25	16-bit, fixed-point DSP; CMOS	122	2048 x 24	1024 x 16	256 x 16	DIP	28
			OTPROM	OTPROM		PLCC	44
			2048 x 24	1024 x 16	256 x 16	Cerdip	28
			UVEPROM	UVEPROM			
77220	24-bit, fixed-point DSP; CMOS	122	2048 x 32	1024 x 24	512 x 24	Ceramic PGA	68
						PLCC	68
77P220R	24-bit, fixed-point DSP; CMOS	122	2048 x 32	1024 x 24	512 x 24	Ceramic PGA	68
			UVEPROM	UVEPROM			
77230AR	32-bit, floating-point DSP; CMOS	150	2048 x 32	1024 x 32	1024 x 32	Ceramic PGA	68
77230AR-003	32-bit, floating-point DSP; CMOS; standard library software	150	n/a	n/a	n/a	Ceramic PGA	68
77P230R	32-bit, floating-point DSP; CMOS	150	2048 x 32	1024 x 32	1024 x 32	Ceramic PGA	68
			UVEPROM	UVEPROM			
77810	16-bit fixed-point modem DSP; CMOS	181	2048 x 24	1024 x 16	256 x 16	Ceramic PGA	68
						PLCC	68
7281	Image pipelined processor; NMOS	5-MHz clock	n/a	n/a	512 x 18	Ceramic DIP	40
72181	Image pipelined processor; CMOS	10-MHz clock	n/a	n/a	512 x 18	DIP	40
						QFP	68
9305	Support device for µPD7281 processors; CMOS	10-MHz clock	n/a	n/a	n/a	Ceramic PGA	132

Speech Processors

Device, µPD	Name	Technology	Clock (MHz)	Data ROM (Bits)	# Package	Pins
7730	ADPCM Speech Encoder/Decoder	NMOS	8	–	DIP	28
77C30	ADPCM Speech Encoder/Decoder	NMOS	8	–	DIP	28
					PLCC	44
7755	ADPCM Speech Synthesizer	CMOS	0.7	96K	DIP	18
					SOP	24
7756	ADPCM Speech Synthesizer	CMOS	0.7	256K	DIP	18
					SOP	24
77P56	ADPCM Speech Synthesizer	CMOS	0.7	256K	DIP	20
				OTPROM	SOP	24
7757	ADPCM Speech Synthesizer	CMOS	0.7	512K	DIP	18
					SOP	24
7759	ADPCM Speech Synthesizer	CMOS	0.7	1024K	DIP	40
				external	QFP	52

Plastic unless ceramic (or cerdip) is specified.

V-Series Development Tools

Part Number (Note 1)	Full Emulator	Full Emulator Probe	Mini-IE Emulator	Mini-IE Probe	Evaluation Boards	EPROM/OTP Device	Relocatable Assembler (Note 13)	C Compiler (Note 14)
μPD70136GJ-12	IE-70136-A016	EP-70136L-A (Note 2)	IE-70136-PC	EP-70136L-PC (Note 2)	DDK-70136	–	RA70136	CC70136
μPD70136GJ-16	IE-70136-A016	EP-70136L-A (Note 2)	IE-70136-PC	EP-70136L-PC (Note 2)	DDK-70136	–	RA70136	CC70136
μPD70136L-16	IE-70136-A016	EP-70136L-A	IE-70136-PC	EP-70136L-PC	DDK-70136	–	RA70136	CC70136
μPD70136L-12	IE-70136-A016	EP-70136L-A	IE-70136-PC	EP-70136L-PC	DDK-70136	–	RA70136	CC70136
μPD70136R-12	IE-70136-A016	EP-70136L-A (Note 3)	IE-70136-PC	EP-70136L-PC (Note 3)	DDK-70136	–	RA70136	CC70136
μPD70136R-16	IE-70136-A016	EP-70136L-A (Note 3)	IE-70136-PC	EP-70136L-PC (Note 3)	DDK-70136	–	RA70136	CC70136
μPD70208GF-8	IE-70208-A010	(Note 12)	EB-V40MINI-IE	–	EB-70208	–	RA70116	CC70116
μPD70208GF-10	IE-70208-A010	(Note 12)	EB-V40MINI-IE	–	EB-70208	–	RA70116	CC70116
μPD70208L-8	IE-70208-A010	IE-70000-2958	EB-V40MINI-IE	ADAPT68PGA 68PLCC (Note 4)	EB-70208	–	RA70116	CC70116
μPD70208L-10	IE-70208-A010	IE-70000-2958	EB-V40MINI-IE	ADAPT68PGA 68PLCC (Note 4)	EB-70208	–	RA70116	CC70116
μPD70208R-8	IE-70208-A010	IE-70000-2959	EB-V40MINI-IE	(Note 4)	EB-70208	–	RA70116	CC70116
μPD70208R-10	IE-70208-A010	IE-70000-2959	EB-V40MINI-IE	(Note 4)	EB-70208	–	RA70116	CC70116
μPD70216GF-8	IE-70216-A010	(Note 12)	EB-V50MINI-IE	–	EB70216	–	RA70116	CC70116
μPD70216GF-10	IE-70216-A010	(Note 12)	EB-V50MINI-IE	–	EB70216	–	RA70116	CC70116
μPD70216L-8	IE-70216-A010	IE-70000-2958	EB-V50MINI-IE	ADAPT68PGA 68PLCC (Note 4)	EB70216	–	RA70116	CC70116
μPD70216L-10	IE-70216-A010	IE-70000-2958	EB-V50MINI-IE	ADAPT68PGA 68PLCC (Note 4)	EB70216	–	RA70116	CC70116
μPD70216R-8	IE-70216-A010	IE-70000-2959	EB-V50MINI-IE	(Note 4)	EB70216	–	RA70116	CC70116
μPD70216R-10	IE-70216-A010	IE-70000-2959	EB-V50MINI-IE	(Note 4)	EB70216	–	RA70116	CC70116
μPD70320GJ	IE-70320-A008	EP-70320GJ (Note 5)	EB-V25MINI-IE-P	EP-70320GJ (Note 6)	DDK-70320	–	RA70320	CC70116
μPD70320GJ-8	IE-70320-A008	EP-70320GJ (Note 5)	EB-V25MINI-IE-P	EP-70320GJ (Note 6)	DDK-70320	–	RA70320	CC70116
μPD70320L	IE-70320-A008	EP-70320L	EB-V25MINI-IE-P	(Note 7)	DDK-70320	–	RA70320	CC70116
μPD70320L-8	IE-70320-A008	EP-70320L	EB-V25MINI-IE-P	(Note 7)	DDK-70320	–	RA70320	CC70116
μPD70322GJ	IE-70320-A008	EP-70320GJ (Note 5)	EB-V25MINI-IE-P	EP-70320GJ (Note 6)	DDK-70320	–	RA70320	CC70116
μPD70322GJ-8	IE-70320-A008	EP-70320GJ (Note 5)	EB-V25MINI-IE-P	EP-70320GJ (Note 6)	DDK-70320	–	RA70320	CC70116
μPD70322L	IE-70320-A008	EP-70320L	EB-V25MINI-IE-P	(Note 7)	DDK-70320	70P322K (Note 10)	RA70320	CC70116
μPD70322L-8	IE-70320-A008	EP-70320L	EB-V25MINI-IE-P	(Note 7)	DDK-70320	70P322K (Note 10)	RA70320	CC70116
μPD70325GJ-8	IE-70325-A008	EP-70320GJ (Note 5)	(Note 12)	(Note 12)	DDK-70325	–	RA70320	CC70116
μPD70325GJ-10	IE-70325-A008 (Note 8)	EP-70320GJ (Note 5)	(Note 12)	(Note 12)	DDK-70325	–	RA70320	CC70116
μPD70325L-8	IE-70325-A008	EP-70320L	(Note 12)	(Note 12)	DDK-70325	–	RA70320	CC70116
μPD70325L-10	IE-70325-A008 (Note 8)	EP-70320L	(Note 12)	(Note 12)	DDK-70325	–	RA70320	CC70116

V-Series Development Tools (cont)

Part Number (Note 1)	Full Emulator	Full Emulator Probe	Mini-IE Emulator	Mini-IE Probe	Evaluation Boards	EPROM/OTP Device	Relocatable Assembler (Note 10)	C Compiler (Note 11)
μPD70327GJ-8 (Note 9)	IE-70320-A008	EP-70320GJ (Note 5)	EB-V25MINI-IE-P	EP-70320GJ (Note 6)	–	–	RA70320	CC70116
μPD70327L-8 (Note 9)	IE-70230-A008	EP-70320L	EB-V25MINI-IE-P	(Note 7)	–	–	RA70320	CC70116
μPD70330GJ-8	IE-70330-A008	EP-70320GJ (Note 5)	EB-V35MINI-IE-P	EP-70320GJ (Note 6)	DDK-70330	–	RA70320	CC70116
μPD70330L-8	IE-70330-A008	EP-70320L	EB-V35MINI-IE-P	(Note 7)	DDK-70330	–	RA70320	CC70116
μPD70332GJ-8	IE-70330-A008	EP-70320GJ (Note 5)	EB-V35MINI-IE-P	EP-70320GJ (Note 6)	DDK-70330	–	RA70320	CC70116
μPD70332L-8	IE-70330-A008	EP-70320L	EB-V35MINI-IE-P	(Note 7)	DDK-70330	70P322K (Note 10)	RA70320	CC70116
μPD70335GJ-8	IE-70335-A008	EP-70320GJ (Note 5)	(Note 12)	(Note 12)	DDK-70330	–	RA70320	CC70116
μPD70335GJ-10	IE-70335-A008 (Note 8)	EP-70320GJ (Note 5)	(Note 12)	(Note 12)	DDK-70330	–	RA70320	CC70116
μPD70335L-8	IE-70335-A008	EP-70320L	(Note 12)	(Note 12)	DDK-70330	–	RA70320	CC70116
μPD70335L-10	IE-70335-A008 (Note 8)	EP-70320L	(Note 12)	(Note 12)	DDK-70330	–	RA70320	CC70116
μPD70337GJ-8 (Note 9)	IE-70330-A008	EP-70320GJ (Note 5)	EB-V35MINI-IE-P	EP-70320GJ (Note 6)	–	–	RA70320	CC70116
μPD70337L-8 (Note 9)	IE-70330-A008	EP-70320L	EB-V35MINI-IE-P	(Note 7)	–	–	RA70320	CC70116
μPD79011GJ-8 (Note 11)	IE-70320-A008	EP-70320GJ (Note 5)	(Note 12)	(Note 12)	–	–	RA70320	CC70116
μPD79011L-8 (Note 11)	+IE-70320-RTOS	EP-70320L	(Note 12)	(Note 12)	–	–	RA70320	CC70116
μPD79021L-8 (Note 11)	IE-70330-A008 +IE-70330-RTOS	EP-70320L	(Note 12)	(Note 12)	–	–	RA70320	CC70116

Notes:

(1) Packages:

Package	Description
GF	80-pin plastic QFP
GJ	74-pin or 94-pin plastic QFP
K	84-pin ceramic LCC with window
L	68-pin or 84-pin plastic LCC
R	68-pin PGA

(2) The EP-70136GL-A and EP-70136L-PC contain both a 68-pin PLCC probe and an adapter which converts the 68-pin PLCC probes to a 74-pin QFP footprint.

(3) 68-pin PGA parts are supported by using the EP-70136L-A PLCC probe or EP-70136L-PC PLCC probe, plus a PLCC socket with a PGA-pinout. A PLCC socket of this type is supplied with the EP-70136L-A.

(4) The EB-V40 MINI-IE and EB-V50 MINI-IE support PGA packages directly; the ADAPT68PGA68PLCC adapter converts the PGA-pinout on the MINI-IE to a PLCC footprint. This adapter is supplied with the MINI-IE.

(5) The EP-70320GJ is an adapter to the EP-70320L, which converts 84-pin PLCC probes to a 94-pin QFP footprint. For GJ parts, both the PLCC probe and the adapter are needed.

(6) The EP-70320GJ adapter can be used to convert the supplied 84-pin PLCC cable of the EB-V25 MINI-IE-P or EB-V35 MINI-IE-P to a 94-pin QFP.

(7) The EB-V25 MINI-IE-P and EB-V35 MINI-IE-P are supplied with an 84-pin PLCC cable.

(8) At the current time, the emulators for the μPD70325 and μPD70335 are specified to 8 MHz. Contact your local NEC Sales Office for the latest information on 10 MHz emulation.

(9) Development for the μPD70327 or μPD70337 can be done using the appropriate μPD70320 or μPD70330 tools; however, debugging the programs in the Software Guard mode is not supported at this time.

(10) The μPD70P322K EPROM device can be used for both μPD70322 and μPD70332 emulation. The μPD70P322K EPROM device can be programmed by using the PA-70P322L Programming Adapter and the PG-1500 EPROM Programmer.

V-Series Development Tools (cont)

Notes (continued):

(11) For emulation of μPD79011 or μPD79021, the base emulator (IE-70320 or IE-70330) plus Real-Time Operating System software IE-70320-RTOS or IE-70330-RTOS) is required.

(12) This emulation option is not currently supported, but may be available in the future. Contact your local NEC Sales Office for further information.

(13) The following relocatable assemblers are available:

RA70116-D52 RA70116-VVT1 RA70116-VXT1	For V20®/V30®/ V40™/V50™	(MS-DOS®) (VAX/VMS™) (VAX/UNIX™ 4.2 BSD or Ultrix™)
RA70136-D52 RA70136-VVT1 RA70136-VXT1	For V33™	(MS-DOS) (VAX/VMS) (VAX/UNIX 4.2 BSD or Ultrix)
RA70320-D52 RA70320-VVT1 RA70320-VXT1	For V25™ and V35™	(MS-DOS) (VAX/VMS) (VAX/UNIX 4.2 BSD or Ultrix)

(14) The following C compilers are available:

CC70116-D52 CC70116-VVT1 CC70116-VXT1	For V20/V30/ V40/V50 and V25/V35	(MS-DOS) (VAX/VMS) (VAX/UNIX 4.2 BSD or Ultrix)
CC70136-D52 CC70136-VVT1 CC70136-VXT1	For V33	(MS-DOS) (VAX/VMS) (VAX/UNIX 4.2 BSD or Ultrix)

V20 and V30 are registered trademarks of NEC Corporation.
V25, V33, V35, V40 and V50 are trademarks of NEC Corporation.
MS-DOS is a registered trademark of Microsoft Corporation.
VAX, VMS and Ultrix are trademarks of Digital Equipment Corporation.
UNIX is a trademark of AT&T Bell Laboratories.

DSP and Speech Development Tools

Part Number (Note 7)	Emulator	Evaluation Board	Assembler (Note 1)	Simulator (Note 2)	EPROM/OTP Device	PG-1500 Adapter (Note 3)
µPD7720AC	EVAKIT-7720B	–	ASM77	SIM77	µPD77P20D	(Note 5)
µPD7720AL	EVAKIT-7720B (Note 4)	–	ASM77	SIM77	–	–
µPD77P20D	EVAKIT-7720B	–	ASM77	SIM77	–	–
µPD77C20AC	EVAKIT-7720B	–	ASM77	SIM77	µPD77P20D	(Note 5)
µPD77C20AL	EVAKIT-7720B (Note 4)	–	ASM77	SIM77	–	–
µPD77C20ALK	EVAKIT-7720B (Note 4)	–	ASM77	SIM77	–	–
µPD77220L	EVAKIT-77220	–	RA77230	SM77230	–	–
µPD77220R	EVAKIT-77220	–	RA77230	SM77230	µPD77P220R	PA-77P230R
µPD77P220R	EVAKIT-77220	–	RA77230	SM77230	–	PA-77P230R
µPD77230AR	EVAKIT-77230	DDK-77230	RA77230	SM77230	µPD77P230R	PA-77P230R
µPD77P230R	EVAKIT-77230	DDK-77230	RA77230	SM77230	–	PA-77P230R
µPD77C25C	EVAKIT-77C25	–	RA77C25	–	µPD77P25C/D	PA-77P25C
µPD77C25L	EVAKIT-77C25 (Note 4)	–	RA77C25	–	µPD77P25L	–
µPD77P25C	EVAKIT-77C25	–	RA77C25	–	–	PA-77P25C
µPD77P25D	EVAKIT-77C25	–	RA77C25	–	–	PA-77P25C
µPD77P25L	EVAKIT-77C25 (Note 4)	–	RA77C25	–	–	–
µPD7755C	NV-300 System	EB-7759	–	–	µPD77P56CR	PA-77P56C
µPD7755G	NV-300 System	EB-7759 (Note 6)	–	–	µPD77P56G	PA-77P56C
µPD7756C	NV-300 System	EB-7759	–	–	µPD77P56CR	PA-77P56C
µPD7756G	NV-300 System	EB-7759 (Note 6)	–	–	µPD77P56G	PA-77P56C
µPD77P56CR	NV-300 System	EB-7759	–	–	–	PA-77P56C
µPD77P56G	NV-300 System	EB-7759 (Note 6)	–	–	–	PA-77P56C
µPD7757C	NV-300 System	EB-7759	–	–	–	–
µPD7757G	NV-300 System	EB-7750 (Note 6)	–	–	–	–
µPD7759C	NV-300 System	EB-7759	–	–	–	–
µPD7759GC	NV-300 System	EB-7759	–	–	–	–
µPD77810L	IE-77810	–	RA77810	–	–	–
µPD77810R	IE-77810	–	RA77810	–	–	–

Notes:

(1) The following assemblers are available:

Part Number	Description
ASM77-D52	Assembler for 7720 (MS-DOS®)
RA77C25-D52	Assembler for 77C25 (MS-DOS)
RA77C25-VVT1	Assembler for 77C25 (VAX/VMS™)
RA77230-D52	Assembler for 77230 (MS-DOS)
RA77230-VVT1	Assembler for 77230 (VAX/VMS)
RA77230-VXT1	Assembler for 77230 (VAX/UNIX™ 4.2 BSD or Ultrix™)

(2) The following simulators are available:

Part Number	Description
SIM77-D52	Simulator for 7720 (MS-DOS)
SM77230-VVT1	Simulator for 77230 (VAX/UNIX)
SM77230-VXT1	Simulator for 77230 (VAX/UNIX™ 4.2 BSD or Ultrix)

(3) By using the specified adapter, the NEC PG-1500 EPROM programmer can be used to program the EPROM/OTP device.

(4) Please check with your NEC Sales Representative on the availability of a PLCC emulation probe.

(5) The µPD77P20D can be programmed using the EVAKIT-7720B.

(6) The EB-7759 comes with an emulation probe for only the 18-pin DIP.

(7) Packages:

Package	Description
C	18, 28, or 40-pin plastic DIP
D	28-pin ceramic DIP
G	24-pin plastic SOP
GC	52-pin plastic QFP
L	44-or 68-pin PLCC
LK	28-pin PLCC
R	68-pin ceramic PGA

MS-DOS is a registered trademark of Microsoft Corporation.
VAX, VMS, and Ultrix are trademarks of Digital Equipment Corporation.
UNIX is a trademark of AT&T Bell Laboratories.

Section 2
Reliability and Quality Control

Reliability and Quality Control

Introduction

As large-scale integration reaches a higher level of density, the reliability of individual devices imposes a more profound impact on system reliability. Great emphasis has thus been placed on assuring device reliability.

Conventionally, performing reliability tests and attaining feedback from the field are the only methods by which reliability has been monitored and measured. At these higher levels of LSI density, however, it is increasingly difficult to activate all of the internal circuit elements in a device, moreover, to detect the degradation of those elements by measuring characteristics across external terminals. As a result, testing alone may not provide enough information to insure today's demanding reliability requirements. A different philosophy and methodology is needed for reliability assurance.

In order to guarantee and improve a high level of reliability for large-scale integrated circuits, it is essential to build quality and reliability into the product. Then, conventional testing can be performed to confirm that the product demonstrates acceptable reliability.

Built-In Quality and Reliability

NEC has introduced the concept of total quality control (TQC) across its entire semiconductor product line for implementing this philosophy. Rather than performing only a few simple quality inspections, quality control is distributed into each process step and then summed to form a consolidated system. TQC involves workers, engineers, quality control staffs, and all levels of management in company-wide activities. Please see Figure 1 for the quality control system flowchart. Through TQC, NEC builds quality into the product and thus can assure high reliability. Additionally, NEC has introduced a pre-screening method into the production line for eliminating potentially defective units. This combination of building quality in and screening projected early failures out has resulted in superior quality and excellent reliability.

Technology Description

Most large-scale integrated circuits utilize high density MOS technology. State-of-the-art high performance has been achieved by improving fine-line generation techniques. By reducing physical parameters, circuit density and performance increase while active circuit power dissipation decreases. The data presented here shows that this advanced technology, combined with the practice of TQC, yields products as reliable as those from previous technologies.

Approaches to Total Quality Control

TQC activities are geared towards total satisfaction of the customer. The success of these activities is dependent upon the total commitment of management to enhancing employee development, maintaining a customer-first attitude, and fulfilling community responsibilities.

First, the quality control function is embedded into each process. This method enables early detection of possible causes of failure and immediate feedback.

Second, the reliability and quality assurance policy reflects the beliefs and practices of the entire organization. This enables companywide quality control activities: at NEC, everyone is involved with the concept and methodology of total quality control.

Third, there is an ongoing research and development effort to set even higher standards of device quality and reliability.

Fourth, extensive failure analysis is performed periodically and appropriate corrective actions are taken as preventative measures. Process control is based on statistical data gathered from this analysis.

The new standard is continuously upgraded, and the iterative process continues. The goal is to maintain the superior product quality and reliability that has become synonymous with the NEC name.

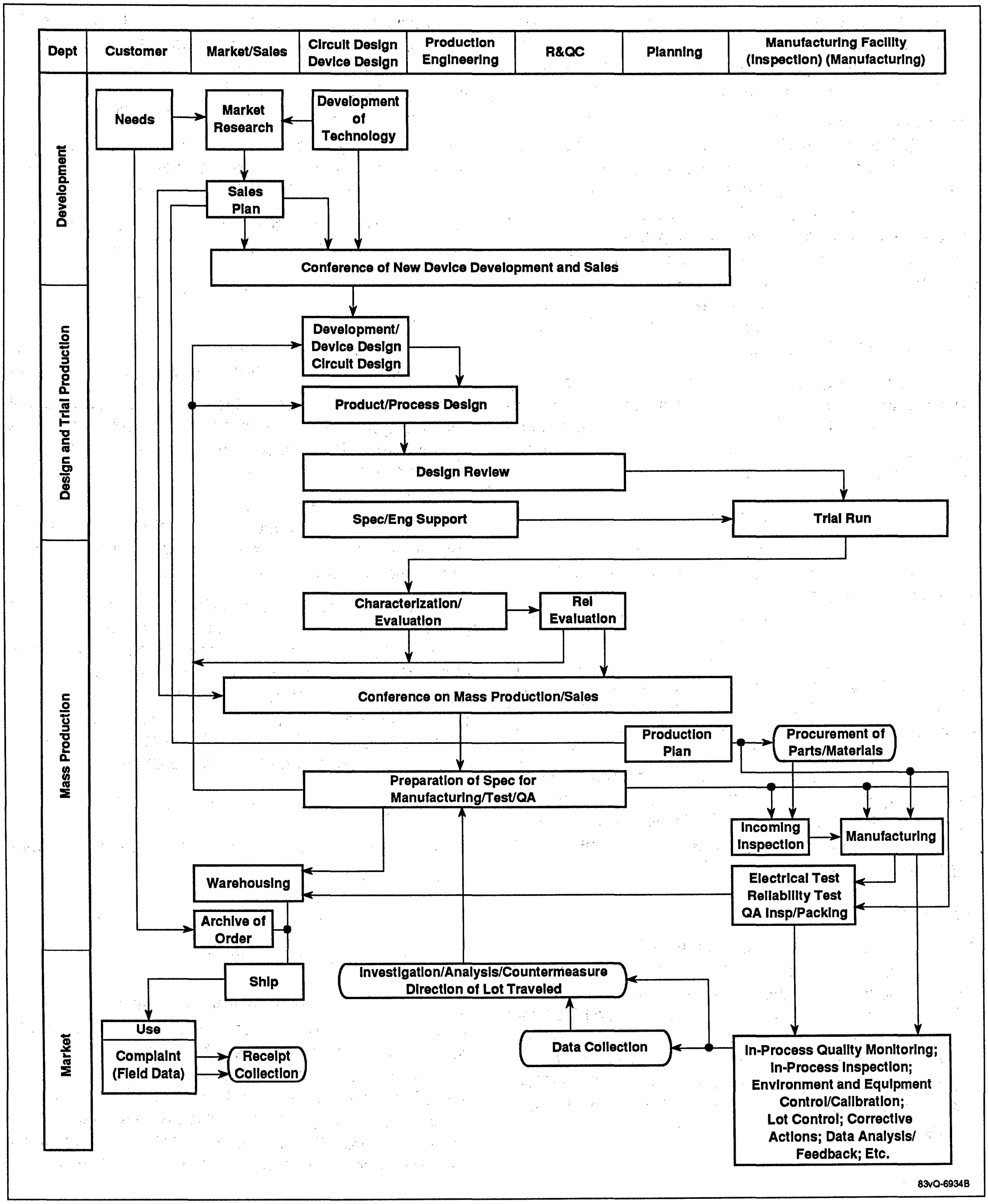
Zero Defect Activities. One of the activities that involves every level of the NEC staff in quality control is the Zero Defects (ZD) Program. As the name implies, the purpose of the ZD program is to minimize, if not eradicate, defects due to controllable causes. Such activities must involve each and every worker and can be most effective when pursued by groups of workers. The groups of workers are organized by consideration of the following:

- A group must have a target to pursue
- Several groups can be organized to pursue the common target
- Each group must have a responsible person
- Each group is well supported

The item of the group target is to be selected among items relating to specifications, inspections, operation standards, and so forth. When data made in the past is available, it is used to make a Pareto diagram which is reviewed for selection of the item most conducive to quality improvement. Records are analyzed and compared with the target, in order to compute the numerical equivalents of the defects. Action is then taken to control these defects as required.

2

Figure 1. Quality Control System Flowchart


Dept
Customer
Market/Sales
Circuit Design Device Design
Production Engineering
R&QC
Planning
Manufacturing Facility (Inspection) (Manufacturing)
Development
Design and Trial Production
Mass Production
Market
Needs
Market Research
Development of Technology
Sales Plan
Conference of New Device Development and Sales
Development/ Device Design Circuit Design
Product/Process Design
Design Review
Spec/Eng Support
Trial Run
Characterization/ Evaluation
Rel Evaluation
Conference on Mass Production/Sales
Production Plan
Procurement of Parts/Materials
Preparation of Spec for Manufacturing/Test/QA
Incoming Inspection
Manufacturing
Warehousing
Electrical Test Reliability Test QA Insp/Packing
Archive of Order
Ship
Investigation/Analysis/Countermeasure Direction of Lot Traveled
Use
Complaint (Field Data)
Receipt Collection
Data Collection
In-Process Quality Monitoring; In-Process Inspection; Environment and Equipment Control/Calibration; Lot Control; Corrective Actions; Data Analysis/ Feedback; Etc.
83VQ-6934B

Statistical Approach. Another approach to quality control is the use of statistical analysis. NEC has been utilizing statistical analysis at each stage of LSI production development, trial runs, and mass production in order to build and maintain product quality. Some of the methods for implementing this statistical approach are:

- Design of experiments
- Control charts
- Data analysis: Variance, correlation, regression, multivariance, etc.
- Cp, Cpk study: Variables and attributes data (Normally, study is done on a monthly basis)

Process control sheets and other QC tools are used to monitor various important parameters such as Cp, Cpk, X, $\overline{X}$, $\overline{X}$-R, electrical parameters, pattern dimensions, bond strength, test percentage defects, etc.

The results of these studies are watched by the production staff, QC Engineers, and other responsible engineers. If any out-of-control or out-of-specification limit is observed, quick action is taken in accordance with corrective action procedures.

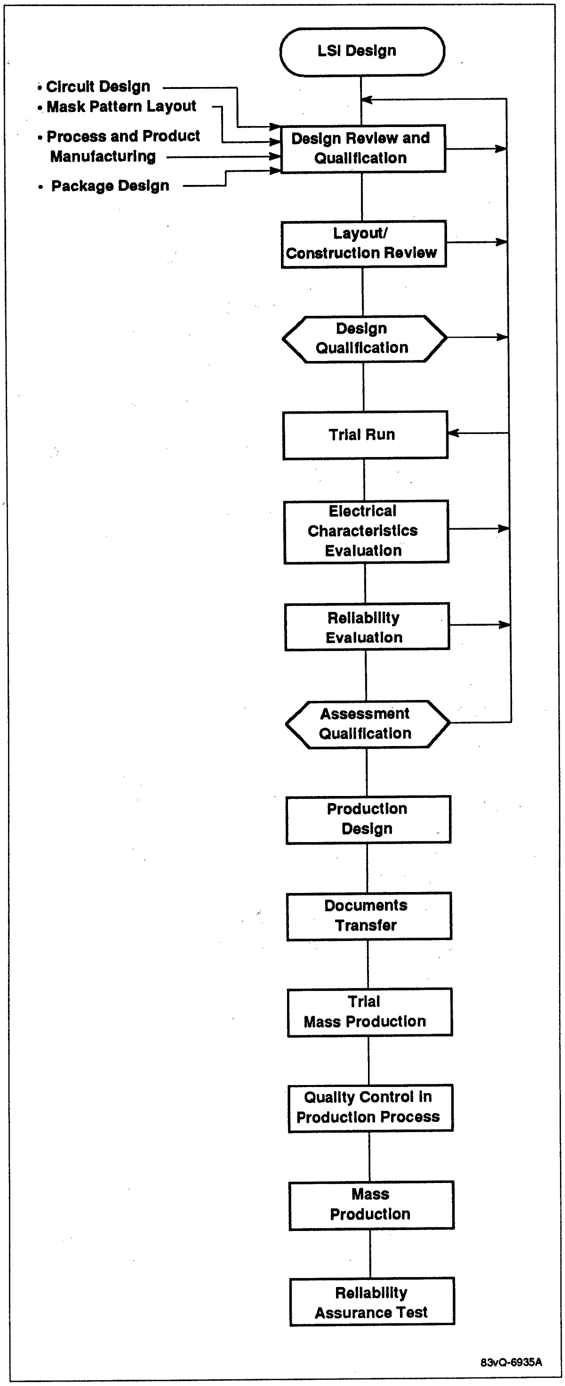
Implementation of Quality Control

Building quality into a product requires early detection of possible causes of failure at each process step, then immediate feedback to remove these causes. A fixed station quality inspection is often lacking in immediate feedback; it is therefore necessary to distribute quality control functions to each process step—including the conceptual stage. Following is a breakdown of the significant steps at which NEC has implemented these functions:

- Product development
- Incoming material inspection
- Wafer processing
- Chip mounting and packaging
- Electrical testing and thermal aging
- Outgoing material inspection
- Reliability testing
- Process/product changes

New Product Development Phase. The product development phase includes conception of a product, review of the device proposal, physical element design and organization, engineering evaluation, and finally, transfer of the product to manufacturing. Quality and reliability are considered at every step. The new product development flow is shown in Figure 2.

Figure 2. New Product Development Flow



Design. Design plays an extremely important role in determining the product quality and reliability. NEC believes that the foundation of device quality is determined at the design stage. The four steps involved in the design of LSI devices are circuit design, mask pattern layout, process and product manufacturing, and package design. Design standards and the standardization of design steps have been established to maximize quality and reliability.

Design Review. After completion of the design, a design review is performed. In this review, the design is compared with design standards and other factors which influence the reliability and quality. If necessary, modification or redesign is then performed. NEC believes that the design review is very essential for not only newly designed products but also for product modifications.

Trial Production/Evaluation/Mass Production. When the design passes the design review successfully, a trial run is carried out. The trial run is evaluated for the products' characteristics and quality/reliability.

Thorough evaluation is carried out by generating samples in which process conditions—ones that cause characteristic factors to change in mass production—are varied deliberately. In addition, reliability tests are conducted for durability, stress resistance, etc., to insure sufficient quality and reliability.

If no problems are found at this stage, the product is approved, after which mass production is possible.

Prior to the transfer, the production Design Department prepares a production schedule, including the reliability and quality control steps relating to the production. Even after the mass-production has started, the standards for those production and control steps are always reexamined for improvements.

Incoming Material Inspection. NEC has various programs to control incoming materials. Some are:

- Vendor/material qualification system
- Purchasing specifications for materials
- Incoming materials inspection
- Inspection data feedback
- Quality meetings with vendor
- Vendor audits

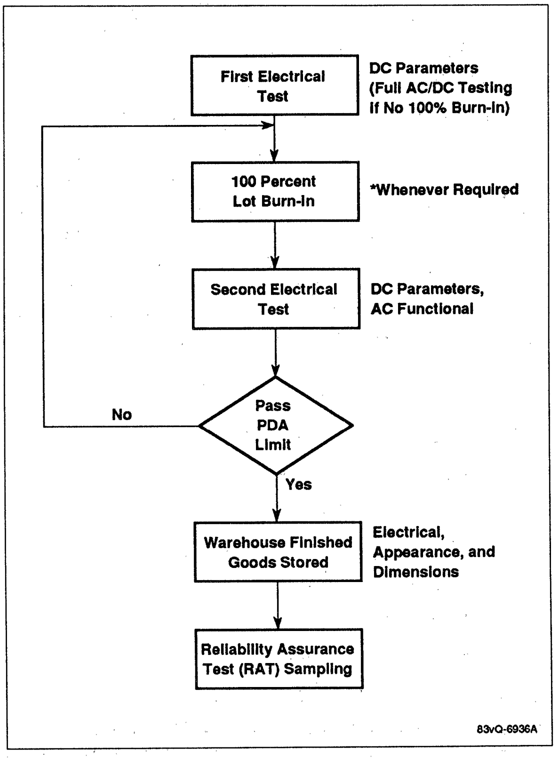
If any parts or materials are rejected at incoming inspection, they are returned to the vendor with a rejection notification form which specifies the failure items and modes. The results of these inspections are used to rate the vendors for future purchasing.

In-process Quality Inspections. Typical in-process quality inspections done at the wafer fabrication, chip mounting and packaging, and device testing stage are listed in Appendix 1.

Electrical Testing and Screening. A flowchart of the typical infant mortality screening (when required) and electrical testing is depicted in Figure 3.

At the first electrical test, DC parameters are tested according to the electrical specifications on 100% of each lot. This is a prescreening prior to any infant mortality test. At the second electrical test, AC functional tests as well as DC parameter tests are performed on 100% of each lot. If the percentage of defective units exceeds the limit, the lot is subjected to rescreen. During this time, the defective units undergo failure analysis, the results of which are fed back into the process through corrective actions.

Figure 3. Electrical Testing and Screening


First Electrical Test
DC Parameters (Full AC/DC Testing If No 100% Burn-In)
100 Percent Lot Burn-In
*Whenever Required
Second Electrical Test
DC Parameters, AC Functional
No
Pass PDA Limit
Yes
Warehouse Finished Goods Stored
Electrical, Appearance, and Dimensions
Reliability Assurance Test (RAT) Sampling
83VQ-6936A

Outgoing Inspection. Prior to warehouse storage, lots are subjected to an outgoing inspection according to the following sampling plan.

• Electrical test:	DC parameters LTPD	3%
	Functional test LTPD	3%
• Appearance:	Major LTPD	3%
	Minor LTPD	7%

Reliability Assurance Tests. Samples are continually taken prior to shipment and subjected to monitoring reliability tests. They are taken from similar process groups, so it may be assumed that the samples' reliability is representative of the reliability of the group.

Reliability Testing

Reliability is defined as the characteristics of an item expressed by the probability that it will perform a required function under stated conditions for a stated period of time. This involves the concepts of probability, the definition of required function(s), and the critical time used in defining the reliability.

Definition of a required function, by implication, treats the definition of a failure. Failure is defined as the termination of the ability of a device to perform its required function. A device is said to have failed if it shows the inability to perform within guaranteed parameters as given in an electrical specification.

Discussion of reliability and failure can be approached in two ways: with respect to systems or to individual devices. Important considerations are the constant failure period, the early failure (infant mortality) period, and overall reliability level.

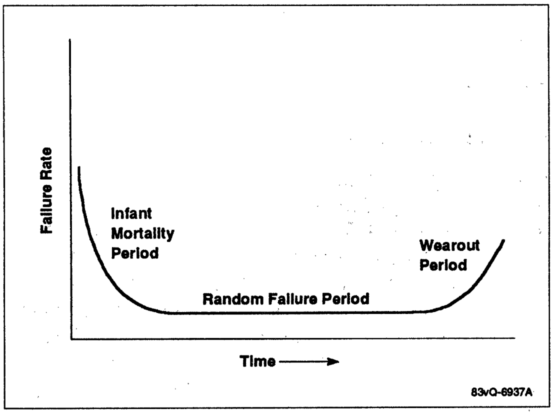
With regard to individual devices, areas of prime interest include specific failure mechanisms, failures in accelerated tests, and failures in screening tests.

The accumulation of normal device failure rates constitutes the expected failure rate of the system hardware: the probability that no device failures will occur in a system is the product of each device's probability that it will not fail. The failure rate of system hardware is then the sum of the failure rates of the components used to construct the system.

Life Distribution

The fundamental principles of reliability engineering predict that the failure rate of a group of devices will follow the well-known bathtub curve in Figure 4. The curve is divided into three regions: infant mortality, random failures, and wearout failures.

Figure 4. Reliability Life (Bathtub) Curve



Infant mortality, as the name implies, represents the early-life failures of devices. These failures are usually associated with one or more manufacturing defects.

After some period of time, the failure rate reaches a low value. This is the random failure portion of the curve, representing the useful portion of the life of a device. During this random failure period, there is a decline in the failure rate due to the depletion of potential random failures from the general population.

The wearout failures occur at the end of the device's useful life. They are characterized by a rapidly rising failure rate over time as devices wear out both physically and electrically.

Thus, for a device that has a very long life expectancy compared to the system which contains it, the areas of concern will be the infant mortality and the random failure portions of the bathtub curve.

Failure Distribution at NEC

In an effort to eliminate infant mortality failures, NEC subjects its products to production burn-in whenever necessary. This burn-in is performed at an elevated temperature for 100 percent of the lots involved and is designed to remove the potentially defective units.

To study the random failure population, integrated circuits returned to NEC from the field undergo extensive failure analysis at respective NEC Manufacturing Divisions. Failure mechanisms are identified and data fed back to cognizant Production and Engineering groups.

This data coupled with in-line data is then used to introduce corrective actions and quality improvement measures.

After elimination of early device failures, a system will be left to the random failure rate of its components. Thus, in order to make proper projections of the failure rate of the system in the operating environment, failure rates must be predicted for the system's components.

Infant Mortality Failure Screening

Establishing infant mortality screening requires knowledge of the likely failure mechanisms and their associated activation energies. The most likely problems associated with infant mortality failures are generally manufacturing defects and process anomalies. These defects and anomalies generally consist of contamination, cracked chips, wire bond shorts, or bad wire bonds. Since these describe a number of possible mechanisms, any one of which might predominate at a given time, the activation energy for infant mortality varies considerably.

Correspondingly, the effectiveness of a screening condition—preferably at some stress level in order to shorten the screening time—varies greatly with the failure mechanism. For example, failures due to ionic contamination have an activation energy of approximately 1.0 eV. Therefore, a 15-hour stress at 125°C junction temperature would be the equivalent of approximately 314 days of operation at a junction temperature of 55°C. On the other hand, failures due to oxide defects have an activation energy of approximately 0.3 eV, and a 15-hour stress at 125°C junction temperature would be the equivalent of approximately four day's operation at 55°C junction temperature. As indicated by this situation, the conditions and duration of infant mortality screening must be strongly dependent on the allowable component, hence system, failures in the field, as well as the economic factors involved.

Empirical data gathered at NEC indicates that early failures (if any) occur after less than 4 hours of stress at 125°C ambient temperature. This fact is supported by the bathtub curve created from the life test results of the same lots, where the failure rate shows a random distribution as opposed to a decreasing failure rate that runs into the random failure region.

Whenever necessary, NEC has adopted this initial infant mortality burn-in at 125°C as a standard production screening procedure. As a result, the field reliability of NEC devices is an order of magnitude higher than the goal set for NEC's integrated circuit products.

NEC believes it is imperative that failure modes associated with infant mortality screens be understood and fixed at the manufacturing level. If such failures can be minimized or eliminated, and countermeasures appropriately monitored, then such screens can be eliminated.

Long-Term Failure Rate

NEC's long-term failure rate goal, based on the mask and process design, is confirmed by life testing using the following conditions:

- A minimum of 1.2 million device hours (= sample size x test period) at 125°C should be accumulated to obtain the accuracy necessary for predicting a failure rate of 0.02% per 1000 hours at 55°C with a 60% confidence level.
- A minimum of 3 million device hours at 125°C should be accumulated to obtain the accuracy necessary for predicting a failure rate of 0.01% per 1000 hours at 55°C with a 60% confidence level.

Accelerated Reliability Testing

NEC performs extensive reliability testing both at preproduction and post-production levels to insure that its products meet the minimum expectations set by NEC. Accelerated reliability testing results are then used to quantitatively monitor the reliability.

As an example, assume that an electronic system contains 1000 integrated circuits and can tolerate 1 percent system failures per month. The failure rate per component is:

$$\frac{\text{1\% Failures}}{\text{720 Hours x 1000 Pcs.}} = .0014 \ \frac{\text{\% Failures}}{\text{1000 Hrs}}$$

or 14 FITs

To demonstrate this failure rate, note that 14 FITs correspond to one failure in about 85 devices during an operating test of 10,000 hours. It is quickly apparent that a test condition is required to accelerate the time-to-failure in a predictable and understandable way. The implicit requirement for the accelerated stress test is that the relationship between the accelerated stress testing condition and the condition of actual use be known.

A most common time-to-failure relationship involves the effect of temperature, which accelerates many physiochemical reactions which may lead to device failure. Other environmental conditions are voltage, current, humidity, vibration, or some combination of these. Appendix 2 lists typical reliability assurance tests performed at NEC for molded integrated circuits. Figure 5 shows the results of some of these tests for various process types.

High-Temperature Operating Life Test. This test is used to accelerate failure mechanisms by operating devices at an elevated temperature of 125°C. The data obtained is translated to a lower temperature by using the Arrhenius relationship.

Figure 5. Typical Reliability Test Results

		HTB	T/H	PCT	T/C
Micro:[1]					
	NMOS	7/19113 (15 FIT)	15/9315	0/11752	—
	CMOS	3/11892 (5.4 FIT)	2/7293	8/9476	—
Memory:		[HTOL]			
	DRAM[2]	10/10052 (19 FIT)	0/9958	0/5880	1/2995
	SRAM[3]	1/10421	2/8142	0/8768	—
	1 MEG DRAM[4]	38/14300 (115 FIT)	0/3634	1/3060	1/1780
Asic:[5]					
	CMOS	2/3506 (33 FIT)	1/1111	1/4764	4/2680
	ECL	0/1080 (8.4 FIT)	—	—	0/141
	BiCMOS	1/895 (18 FIT)	0/1073	0/935	0/1781

Information has been extracted from NEC Report Numbers:
[1] TRQ-89-05-0030 [2] TRQ-89-01-0021
[3] TRQ-88-09-0008 [4] TRQ-89-01-0020
[5] TRQ-89-04-0025

High-Temperature and High-Humidity Test. Semiconductor integrated circuits are highly sensitive to the effect of humidity causing electrolytic corrosion between biased lines. The high-temperature and high-humidity test is performed to detect failure mechanisms that are accelerated by these conditions, such as leakage-related problems and drifts in device parameters due to process instability.

High-Temperature Storage Test. Another common test is the high-temperature storage test, in which devices are subjected to elevated temperatures with no applied bias. This test is used to detect process instability and stress migration problems.

Environmental Tests. Other environmental tests are performed to detect problems related to the package, material, susceptibility to extremes in environment, and problems related to usage of the devices.

Failure Rate Calculation/Prediction

When predicting the failure rate at a certain temperature from accelerated life test data, the activation energies of the failure mechanisms involved should be considered. This is done whenever the exact cause of failures is known through failure analyses results.

In some cases, an average activation energy is assumed in order to accomplish a quick first order approximation. NEC assumes an average activation energy of 0.7 eV for such approximations. This average value has been assessed from extensive reliability test results and yields a conservative failure rate.

Since most semiconductor failures are temperature dependent, the Arrhenius relationship is used to normalize failure rate predictions at a system operation temperature of 55°C. It assumes that temperature dependence is an exponential function that defines the probability of occurrence, and that the degradation of a performance parameter is linear with time. The Arrhenius model includes the effects of temperature and activation energies of the failure mechanisms in the following Arrhenius equation:

$$A = \exp \frac{-E_A (T_{J1} - T_{J2})}{k(T_{J1})(T_{J2})}$$

Where:

A = Acceleration factor
E_A = Activation energy
T_{J1} = Junction temperature (in K) at $T_{A1} = 55°C$
T_{J2} = Junction temperature (in K) at $T_{A2} = 125°C$
k = Boltzmann's constant = 8.62×10^{-5} eV/K.

Because the thermal resistance and power dissipation of a particular device type cannot be ignored, junction temperatures (T_{J1} and T_{J2}) are used instead of ambient temperatures (T_{A1} and T_{A2}). We calculate junction temperatures using the following formula:

$T_J = T_A$ + (Thermal Resistance) (Power Diss. at T_A)

In order to estimate long term failure rate, the acceleration factor must be used to determine the simulated test time. From the high temperature operating life test results, failure rates can then be predicted at a 60% confidence level using the following equation:

$$L = \frac{X^2 10^5}{2T}$$

Where:

L = Failure rate in %/1000 hours

*X^2 = The tabular value of chi-squared distribution at a given confidence level and calculated degrees of freedom (2f + 2, where f = number of failures)

T = # of equivalent device hours
= (# of devices) x (# of test hours) x (acceleration factor)

*Since the failures of concern here are the random, not the infant mortality failures (that is, the end of the downward slope and the middle-constant-section of the bathtub curve in Figure 4), X^2 is determined assuming a one-sided, fixed time test.

Another method of expressing failures is in FITs (failures in time). One FIT is equal to one failure in 10^9 hours. Since L is already expressed as %/1000 hours (10^{-5} failure/hr), an easy conversion from %/1000 hours to FIT would be to multiply the value of L by 10^4.

EXAMPLE: A sample of 960 pieces was subjected to 1000 hours 125°C burn-in. One reject was observed. Given that the acceleration factor was calculated to be 34.6 using the Arrhenius equation, what is the failure rate normalized to 55°C using a confidence level of 60%? Express the failure rate in FIT:

Solution:

For $n = 2f + 2 = 2(1) + 2 = 4$, $X^2 = 4.046$.

$$\text{Then } L = \frac{X^2\ 10^5}{2T} \text{ (\%/1000 hour)}$$

$$= \frac{X^2\ 10^5 \quad \text{(\%/1000 hr)}}{2\ (\#\text{ of dev.})\ (\#\text{ of test hrs.})\ (\text{accl. factor})}$$

$$= \frac{(4.046) \quad 10^5}{2(960)\ (1000)\ (34.6)} = 0.0061 \text{ (\%/1000 hr)}$$

Therefore, FIT = $0.0061 \bullet (10^4) = 61$

Product/Process Changes

As mentioned previously, a design review is performed for product modifications or changes. Once the design is approved, and processes altered (if necessary) for maximum quality, the device goes through qualification testing to check the reliability. If the test results are acceptable, the product is released for mass production.

Testing is also performed when only a process modification or change is made.

The typical qualification/process change tests are listed in Appendix 3.

Failure Analysis

At NEC, failure analysis is performed not only on field failures, but also routinely on products which exhibit defects during the production process. This data is closely checked for correlation with the production process quality information, inspection results, and reliability test data. Information derived from these failure analyses is used to improve product quality.

As there are a lot of failure mechanisms of LSI devices, highly advanced analytical technologies are required to investigate such failures in detail. The standard failure analysis flowchart relating to the returned products from customers is shown in Appendix 4.

NEC's Goals on Failure Rates

The reject rate at customer's incoming inspection, the infant mortality rate, and the long term reliability, are all monitored and checked against NEC's quality and reliability targets (listed in Figure 6).

Figure 6. NEC Quality and Reliability Targets

	Reject Rate at Customer's Incoming Electrical Inspection (PPM)						Long Term Reliability (FIT)						Infant Mortality (RT)					
	Memory		µCOM	Gate Arrays			Memory		µCOM	Gate Arrays			Memory		µCOM	Gate Arrays		
Year	ECL RAM	MOS		BiCMOS	ECL	CMOS	ECL RAM	MOS		BiCMOS	ECL	CMOS	ECL RAM	MOS		BiCMOS	ECL	CMOS
1988	150	50	100	1000	300	300	100	50	100	1000	300	150	100	100	150	1000	300	400
1990	100	50	100	500	200	150	80	50	80	500	250	100	80	100	150	500	250	300

Summary and Conclusion

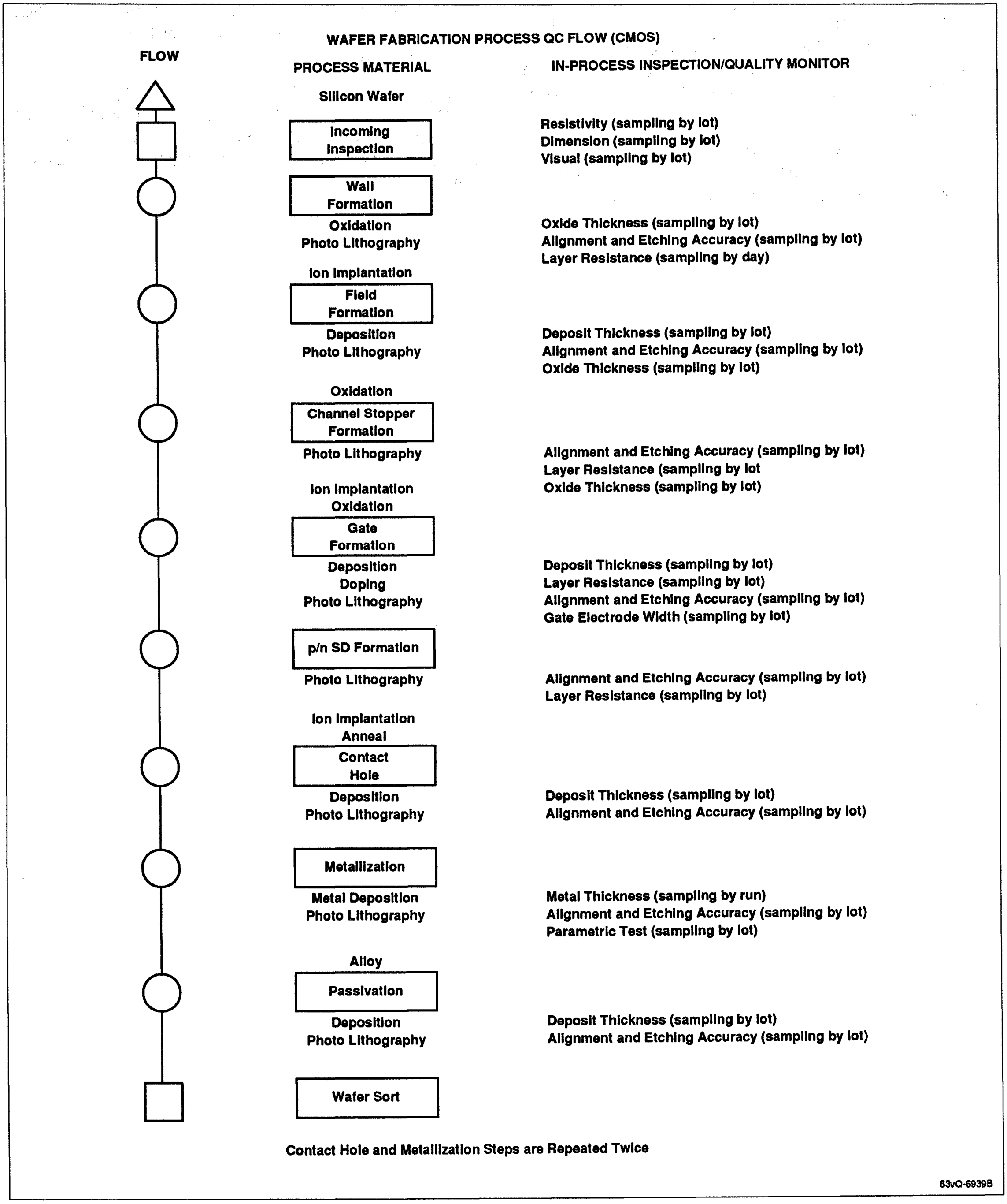
As has been discussed, building quality and reliability into products is the most efficient way to ensure product success. NEC's approach of distributing quality control functions to process steps, then forming a total quality control system, has produced superior quality and excellent reliability.

Prescreening, whenever necessary, has been a major factor in improving reliability. In addition, monthly reliability assurance tests have ensured high outgoing quality levels.

The combination of building quality into products, effective prescreening of potential failures, and monitoring of reliability through extensive testing, has established a singularly high standard of quality and reliability for NEC's large-scale integrated circuits.

Through a companywide quality control program, continuous research and development activities, extensive failure analysis, and process improvements, this higher standard of quality and reliability will continuously be set and maintained.

Appendix 1
Typical QC Flow for CMOS Fabrication



WAFER FABRICATION PROCESS QC FLOW (CMOS)

FLOW	PROCESS MATERIAL	IN-PROCESS INSPECTION/QUALITY MONITOR
	Silicon Wafer	
	Incoming Inspection	Resistivity (sampling by lot) Dimension (sampling by lot) Visual (sampling by lot)
	Wall Formation Oxidation Photo Lithography Ion Implantation	Oxide Thickness (sampling by lot) Alignment and Etching Accuracy (sampling by lot) Layer Resistance (sampling by day)
	Field Formation Deposition Photo Lithography Oxidation	Deposit Thickness (sampling by lot) Alignment and Etching Accuracy (sampling by lot) Oxide Thickness (sampling by lot)
	Channel Stopper Formation Photo Lithography Ion Implantation Oxidation	Alignment and Etching Accuracy (sampling by lot) Layer Resistance (sampling by lot Oxide Thickness (sampling by lot)
	Gate Formation Deposition Doping Photo Lithography	Deposit Thickness (sampling by lot) Layer Resistance (sampling by lot) Alignment and Etching Accuracy (sampling by lot) Gate Electrode Width (sampling by lot)
	p/n SD Formation Photo Lithography Ion Implantation Anneal	Alignment and Etching Accuracy (sampling by lot) Layer Resistance (sampling by lot)
	Contact Hole Deposition Photo Lithography	Deposit Thickness (sampling by lot) Alignment and Etching Accuracy (sampling by lot)
	Metallization Metal Deposition Photo Lithography Alloy	Metal Thickness (sampling by run) Alignment and Etching Accuracy (sampling by lot) Parametric Test (sampling by lot)
	Passivation Deposition Photo Lithography	Deposit Thickness (sampling by lot) Alignment and Etching Accuracy (sampling by lot)
	Wafer Sort	

Contact Hole and Metallization Steps are Repeated Twice

83vQ-6939B

Appendix 1
Typical QC Flow for PLCC Assembly/Test

Process/Materials	The Check of Manufacturing Conditions				The Check of Manufacturing Qualities			
	Check Items	Frequency	Instrument	Checked By	Check Item	Frequency	Instrument	Checked By
1 Sorted Wafers								
2 Wafer Visual					Wafer Visual	100%	(Naked Eye)	Operator
3 Dicing	Table Speed DI Water Blade Height	Every Shift	Indicators Gauges	P.C.	Sawing Dimensions	Before Running	Microscope with Filter Eyepiece	Operator
4 Break and Expand	Wafer Break Conditions Wafer Expand Conditions	Every Shift	Indicators Gauges	P.C.	Wafer Visual	100%	(Naked Eye)	Operator
5 Die Visual Inspection					Die Visual	Every Lot Sampling (Or 100%)	Microscope	Operator
6 Lead Frames	Die Attached Conditions	Every Shift	Indicators Thermocouple, Potentiometer	P.C.	Die Visual Epoxy Coverage	Every Magazine	(Naked Eye)	Operator
7 Die Attached	Temperature					Every Shift	Microscope	
8 Epoxy Cure (Not Done for Gold Die Attached product)	Heat Temperature N_2 Flow	Every Shift	Indicators Gauges	P.C.	Shear Strength	Every Shift	Dynamometer	Operator
9 Fine Wire	Bonding Conditions	Every Shift	Indicators	P.C.	Visual	Every Magazine	Microscope	Operator
10 Wire Bonding	Temperature	Every Week	Thermocouple and Potentiometer	P.C.	Wire Pull Test	Every Shift	Tension Gauge	Operator
11 Pre-Seal Visual Inspection					Die Visual	Every Lot Sampling (or 100%)	Microscope	Inspector
12 Molding Compound	Temperature of Pellet, Expiration Date	Every Shift	Thermocouple	P.C.				
13 Molding	Temperature Profile of Die Set Preheat Temperatue Pressure Cure Time	Every Shift	Thermocouple, Potentiometer	P.C.	Visual	100%	(Naked Eye)	Operator
14 Mold Aging	Temperature	Every Shift	Indicator	P.C.				
15 Deflashing	Deflashing Conditions	Every Shift	Indicators	P.C.	Visual	Every Lot	(Naked Eye)	Operator
	Concentration	Every Week	Titration	Tech.				
	Density	Every Week	Density Meter	Tech.				
	Water Jet Pressure	Every Day	Gauge	Tech.				
16 Plating	Plating Conditions	Every Day	Indicators	P.C.				
	Concentration	Every Week	Titration	Tech.				

83VQ-69140B

Appendix 1
Typical QC Flow for PLCC Assembly/Test (Cont.)

Process/Materials	The Check of Manufacturing Conditions				The Check of Manufacturing Qualities			
	Check Items	Frequency	Instrument	Checked By	Check Item	Frequency	Instrument	Checked By
17 Plating Inspection					Visual Plating Thickness Composition Solderability	Every Lot Every Lot Every Lot Once/Day	(Naked Eye) X-ray X-ray (Naked Eye)	Technician Technician Technician Technician
18 Marking Ink 19 Marking	Marking Conditions	Every Shift	Indicators	P.C.	Visual	Every Lot	(Naked Eye)	Operator
20 Mark Cure	Temperature	Every Shift	Thermocouple	P.C.	Marking Permanency	Twice/Shift	Automatic Tester	Operator
21 Lead Forming	Dimensions	Every Shift (Before Running)	Test Jig. Caliper	Operator	Visual	Every Lot	(Naked Eye)	Operator
22 Final Assembly Inspection					Visual	Every Lot	Magnifying Lamp	Operator
23 1st Electrical Sorting	P.M. Check Sample Check	Every Day Before Testing	P.M. Jig. Test Samples	Operator Operator	Electrical Characteristics	100%	IC Tester	Operator
24 Burn-In (Whenever Necessary)	Burn-In Conditions	Every Batch	Indicator	P.C.				
25 1st Electrical Sorting		Every Day Before Testing	P.M. Jig. Test Samples	Operator Operator	Electrical Characteristics	100%	IC Tester	Operator
26 Reliability Assurance Test		Every Month						
27 In-Warehouse Inspection		Every Day Before Testing	P.M. Jig. Test Samples		Electrical Characteristics Visual (Major) Visual (Minor)	Every Lot Every Lot Every Lot	IC Tester (Naked Eye) and Microscope (Naked Eye)	Inspector Inspector Inspector
28 Warehousing								

83vQ-6941B

Appendix 2
Typical Reliability Assurance Tests

The life tests performed by NEC consist of high temperature bias life (HTB), high temperature storage life (HTSL), high temperature/high humidity (T/H), and high humidity storage life (HHSL) tests. Additionally, various environmental and mechanical tests are performed. The table below shows the conditions of the various life tests, environmental tests, and mechanical tests.

Test Item	Symbol	MIL-STD-883C Method	Condition	Remarks
High Temperature Bias Life	HTB	1005	T_A = 125°C, V_{DD} specified per device type.	(Note 1)
High Temperature Storage Life	HTSL	1008	T_A = 150°C.	(Note 1)
High Temperature/ High Humidity	T/H		T_A = 85°C, RH = 85%, V_{DD} = 5.5 V.	(Note 1)
High Humidity Storage Life	HHSL		T_A = 85°C, RH = 85%.	(Note 1)
Pressure Cooker	PCT		T_A = 125°C, P = 2.3 atm.	(Note 1)
Temperature Cycling	T/C	1010	– 65°C to 150°C, 1 hr/cycle.	(Note 1)
Lead Fatigue	C3	2004	90° bends. 3 bends without breaking.	(Note 2)
Solderability	C4	2003	230°C, 5 sec, Rosin Base Flux.	(Note 3)
Soldering Heat/ Temperature Cycle/ Thermal Shock	C6	(Note 4) 1010 1011	260°C, 10 sec, Rosin Base Flux/ 10-1 hr cycles, –65°C to 150°C/ 15-10 min cycles, 0°C to 100°C	(Note 1)

Notes:

(1) Electrical test per data sheet is performed. Devices that exceed the data sheet limits are considered to be rejects.

(2) Broken lead is considered to be a reject.

(3) Less than 95% coverage is considered to be a reject.

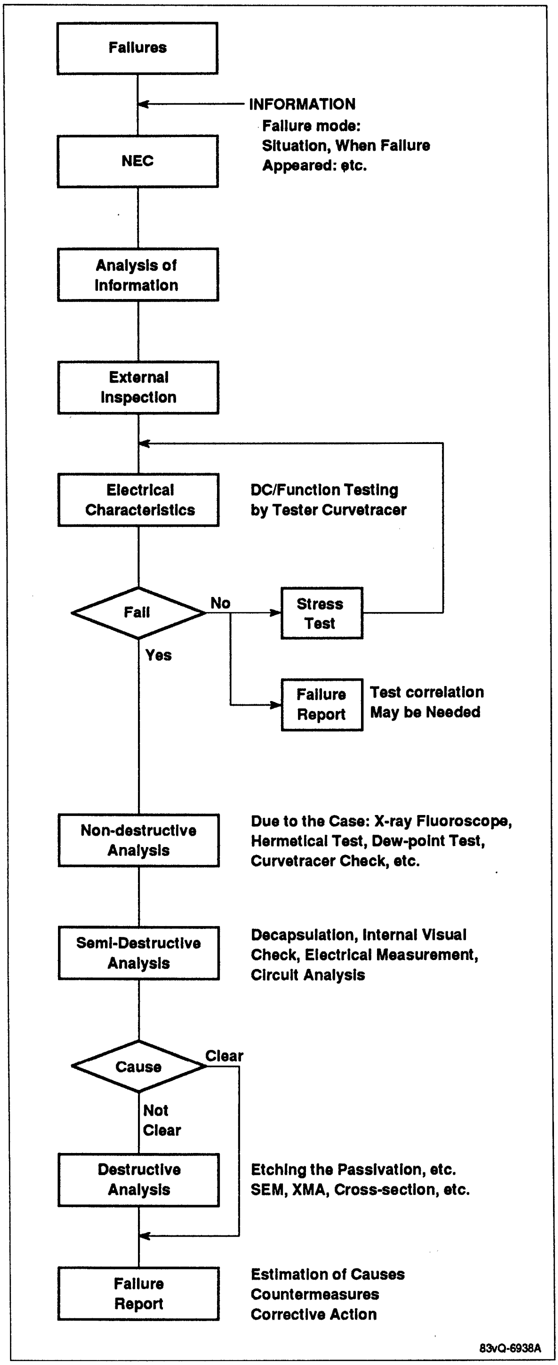
(4) MIL-STD-750A, method 2031.

Appendix 3
New Product / Process Change Tests

Test Item	Test Conditions	Sample Size	Newly Developed Product	Shrink Die	New Package	Wafer	Assembly
High Temp. Operating Life	See Appendix 2, 1000H	20 to 50 pcs X 1 to 3 lots	0	0	0	0	0
High Temp. Storage Life	T = 150°C (Plastic), 175°C (Ceramic), 1000H	10 to 20 pcs X 1 to 3 lots	0	0	0	0	0
High Temp. and Humidity Bias Life (Plastic Device)	See Appendix 2, 1000H	20 to 50 pcs X 1 to 3 lots	0	0	0	0	0
Pressure cooker (Plastic Device)	See Appendix 2, 288H	10 to 20 pcs X 1 to 3 lots	0	0	0	0	0
Thermal Environmental	See Appendix 2	10 to 20 pcs X 1 to 3 lots	0	X	0	X	0
Mechanical Environmental (Ceramic Device)	20G, 10 to 2000 Hz 1500G, 0.5 ms 20000G, 1 min	10 to 20 pcs X 1 to 3 lots	0	X	0	X	0
Lead Fatigue	See Appendix 2	5 pcs X 1 to 3 lots	X	—	X	—	X
Solderability	See Appendix 2	5 pcs X 1 to 3 lots	X	—	X	—	X
ESD	(1) C = 200 pF, R = 0Ω (2) C = 100 pF, R = 1.5 KΩ	20 pcs X 1 to 3 lots	0	0	X	0	X
Long Term T/C	See Appendix 2, 1000 cy	10 to 50 pcs X 1 to 3 lots	0	0	0	0	0

0 – Performed X – Perform if Necessary — – Not Performed

Appendix 4
Failure Analysis Flowchart



Section 3
μPD7500 Series:
4-Bit, CMOS Microcomputers

NEC
NEC Electronics Inc.

µPD7502/03 4-Bit, Single-Chip CMOS Microcomputers With LCD Controller/Driver

Description

The µPD7502 and µPD7503 4-bit, single-chip CMOS microcomputers have advanced fourth-generation architecture with the functional blocks necessary for a single-chip controller, including an 8-bit timer/event counter, an 8-bit serial I/O, and an LCD controller/driver.

The instruction set includes the following types of instructions: addressing, table look-up, bit manipulation, vectored jump, auto increment or decrement data pointer, and conditional skip. These instructions maximize use of fixed program memory space.

Both devices are manufactured with the CMOS process and have a maximum power consumption of 900 µA at 5 V and 300 µA at 3 V. Halt and stop modes further reduce power consumption.

These devices are ideal for a wide range of solar- and battery-powered applications.

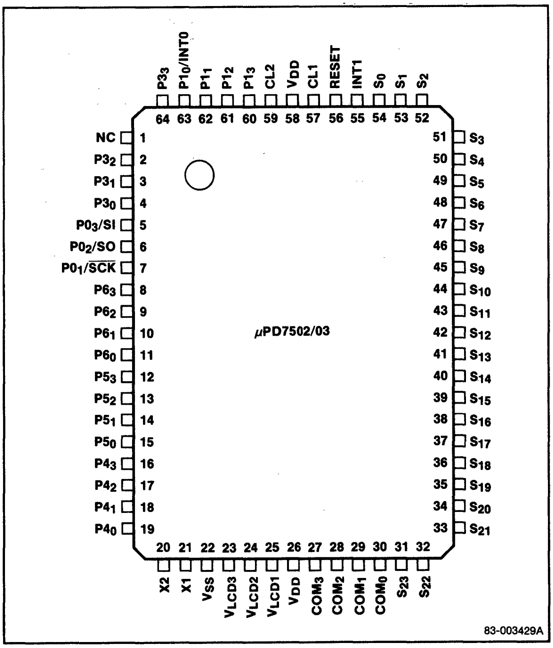
Features

□ 92 powerful instructions
□ Program ROM
— µPD7502: 2048 x 8-bit
— µPD7503: 4096 x 8-bit
□ Data RAM
— µPD7502: 128 x 4-bit
— µPD5703: 224 x 4-bit
□ Interrupts
— External: INT0, INT1
— Internal: INTT (timer/event counter)
INTS (serial interface)
□ 8-bit timer/event counter
— Based on crystal oscillation
— External event counter (prescale option by 64)
□ Serial interface
□ LCD controller/driver
— Programmable multiplexing mode: triplex, quadruplex, or pseudo-static
— 4 common lines (COM_0-COM_3)
— 24 segment lines (S_0-S_{23})
□ Standby modes: stop, halt
□ Data retention mode
□ I/O ports
— 3-bit input port
— 4-bit input port
— 4-bit output port
— Two 4-bit I/O ports with 8-bit capability
— 4-bit I/O port with each bit configurable as an input or output
□ RC oscillation clock
□ Crystal oscillation clock
□ 2.5 to 6.0 V operating voltage
□ CMOS technology

Ordering Information

Part No.	Package Type	Max Frequency of Operation
µPD7502GF-12	64-pin plastic QFP	410 kHz
µPD7503GF-12	64-pin plastic QFP	410 kHz

Pin Configuration



Pin Identification

No.	Symbol	Function
1	NC	No connection
2-4, 64	$P3_3$-$P3_0$	4-bit output port 3
5-7	$P0_3$/SI $P0_2$/$\overline{SO}$ $P0_1$/$\overline{SCK}$	3-bit input port 0, or serial I/O interface
8-11	$P6_3$-$P6_0$	4-bit I/O port 6
12-15	$P5_3$-$P5_0$	4-bit I/O port 5
16-19	$P4_3$-$P4_0$	4-bit I/O port 4
20, 21	X2, X1	Crystal clock/external event input port X
22	V_{SS}	Ground
23-25	V_{LCD3}-V_{LCD1}	LCD bias supply inputs
26, 58	V_{DD}	Positive power supply
27-30	COM_3-COM_0	LCD backplane driver outputs
31-54	S_{23}-S_0	LCD segment driver outputs
55	INT1	External interrupt
56	RESET	RESET input
57, 59	CL1, CL2	System clock input
60-63	$P1_3$-$P1_1$, $P1_0$/INT0	4-bit input port 1, or external interrupt INT0

Status of Unused Pins

Name	Pin Connection
CL2	Open
X1	V_{SS}
X2	Open
$P0_1$/$\overline{SCK}$ $P0_2$/SO $P0_3$/SI	V_{SS} or V_{DD}
$P1_0$/INT0	V_{SS}
$P1_1$-$P1_3$	V_{SS} or V_{DD}
$P3_0$-$P3_3$	Open
$P4_0$-$P4_3$ $P5_0$-$P5_3$ $P6_0$-$P6_3$	Input mode: V_{SS} or V_{DD} Output mode: Open
INT1	V_{SS}
S_0-S_{23} COM_0-COM_3 V_{LCD1}-V_{LCD3}	Open

Pin Functions

$P0_3$/SI, $P0_2$/SO, $P0_1$/$\overline{SCK}$ [Port 0 or Serial Interface]

This port can be configured as a 4-bit parallel input port 0 or as the 8-bit serial I/O interface under control of the serial mode select register. The serial interface consists of the serial input (SI), the serial output (SO), and the serial clock ($\overline{SCK}$), which synchronizes data transfer.

$P1_3$-$P1_1$, $P1_0$/INT0 [Port 1 or Interrupt]

4-bit input port 1. Line $P1_0$ is shared with external interrupt INT0, which is a rising edge-triggered interrupt.

$P3_3$-$P3_0$ [Port 3]

4-bit, latched three-state output port 3.

$P4_3$-$P4_0$ [Port 4]

4-bit input or latched three-state output port 4. Can perform 8-bit I/O in conjunction with port 5.

$P5_3$-$P5_0$ [Port 5]

4-bit input or latched three-state output port 5. Can perform 8-bit I/O in conjunction with port 4.

$P6_3$-$P6_0$ [Port 6]

4-bit input or latched three-state output port 6. The port 6 mode select register configures individual lines as inputs or outputs.

COM_3-COM_0 [LCD Backplane Driver Outputs]

LCD backplane driver outputs.

S_{23}-S_0 [LCD Segment Driver Outputs]

LCD segment driver outputs.

INT1 [Interrupt]

This external interrupt is a rising edge-triggered interrupt latched by CL.

RESET

A high-level input to this pin initializes the μPD7502/7503.

X2, X1 [Crystal Clock/External Event Input Port X]

For crystal clock operation, connect a crystal oscillator circuit to input X1 and output X2. For external event counting, input external event pulses to X1 and leave X2 open.

CL1, CL2 [System Clock Input]

Connect an 82-kΩ resistor across CL1 and CL2, and a 33-pF capacitor from CL1 to V_{SS}. Or, connect an external clock source to CL1 and leave CL2 open.

V_{LCD3}-V_{LCD1} [LCD Bias Voltage Inputs]

LCD bias voltage supply inputs to the LCD voltage controller. Apply appropriate voltages from a voltage ladder connected across V_{DD}.

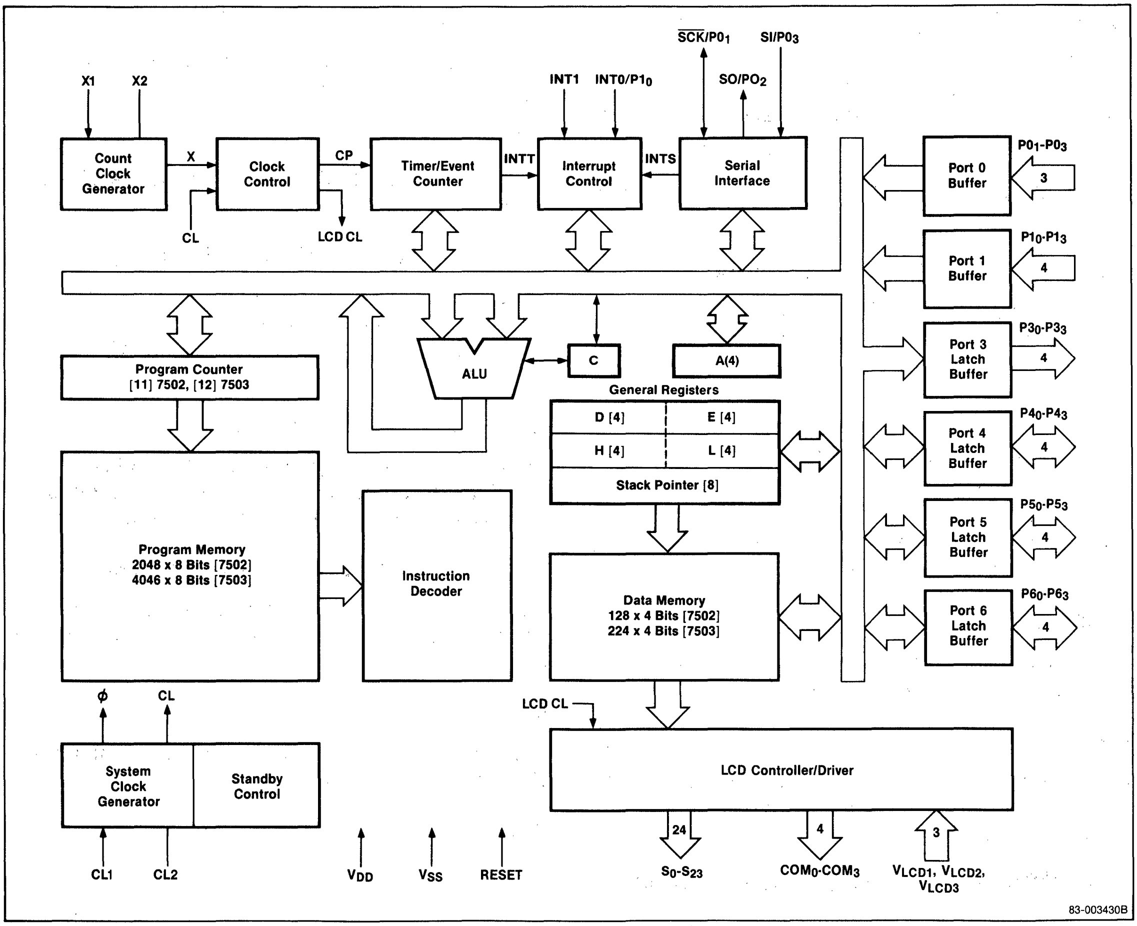
V_{DD}

Positive power supply. For proper operation, apply a single voltage from 2.5 to 6.0 V.

V_{SS}

Ground.

Block Diagram



See figures 1 through 8 for additional block diagram details.

Figure	Title
1	Data Memory Map
2	Program Memory Map
3	Interface at Input/Output Ports
4	Clock Control
5	Timer/Event Counter
6	Interrupt Control
7	Serial Interface
8	LCD Controller/Driver

Figure 1. Data Memory Map

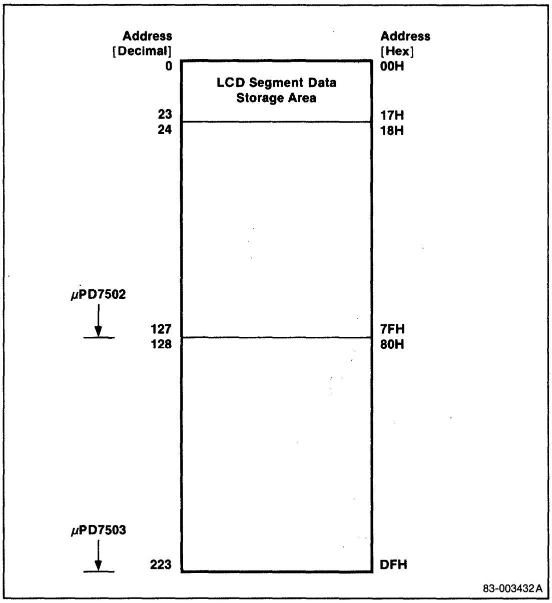


Figure 2. Program Memory Map

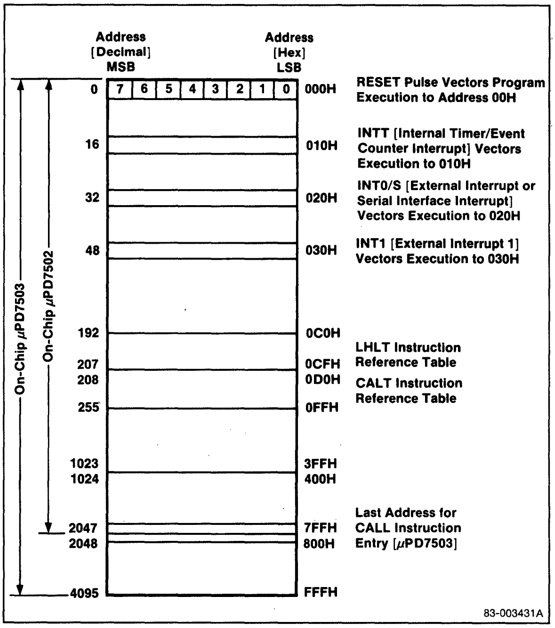
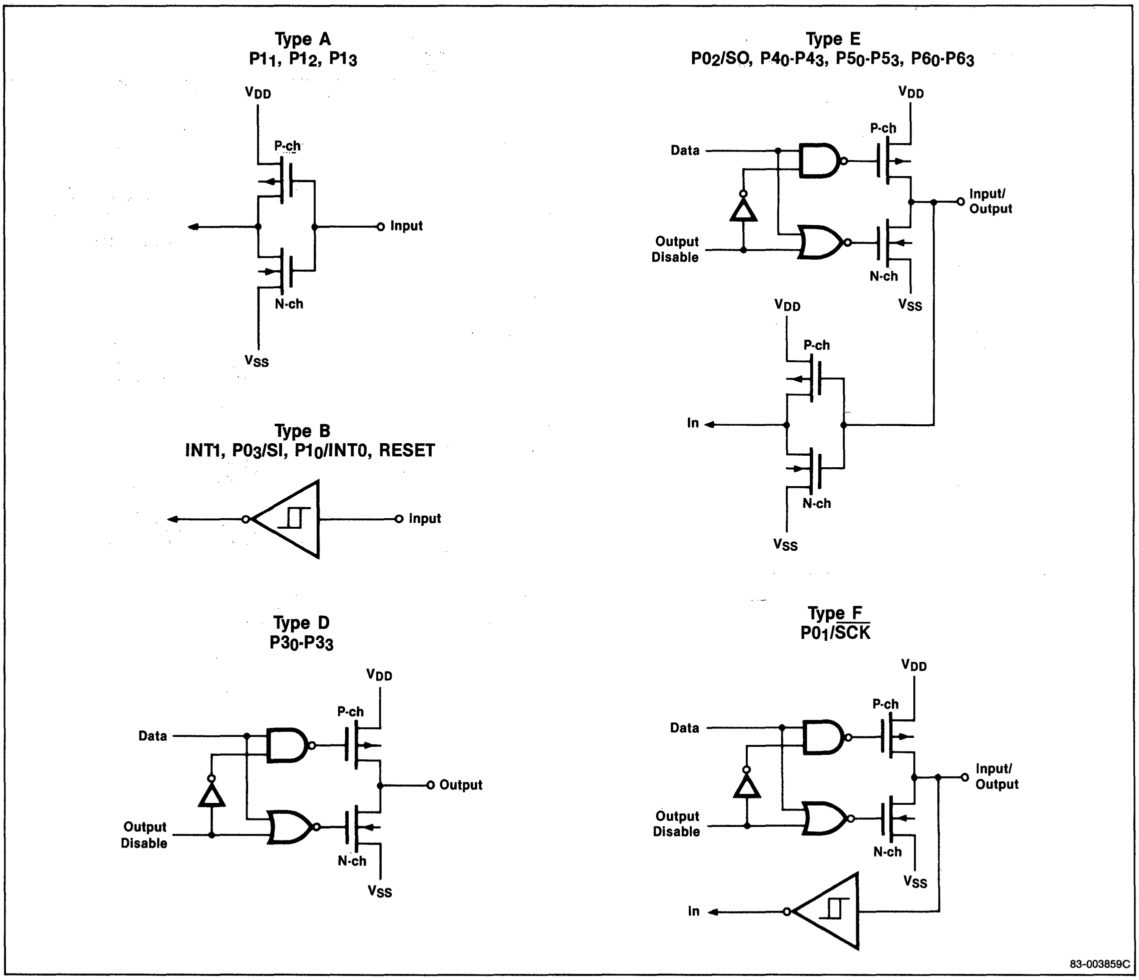

Address
[Decimal]
MSB
Address
[Hex]
LSB
7 6 5 4 3 2 1 0
0
000H
RESET Pulse Vectors Program Execution to Address 00H
16
010H
INTT [Internal Timer/Event Counter Interrupt] Vectors Execution to 010H
32
020H
INT0/S [External Interrupt or Serial Interface Interrupt] Vectors Execution to 020H
48
030H
INT1 [External Interrupt 1] Vectors Execution to 030H
192
0C0H
LHLT Instruction Reference Table
207
0CFH
208
0D0H
CALT Instruction Reference Table
255
0FFH
1023
3FFH
1024
400H
Last Address for CALL Instruction Entry [μPD7503]
2047
7FFH
2048
800H
4095
FFFH
On-Chip μPD7503
On-Chip μPD7502
83-003431A

Figure 3. Interface at Input/Output Ports



Type A
$P1_1$, $P1_2$, $P1_3$

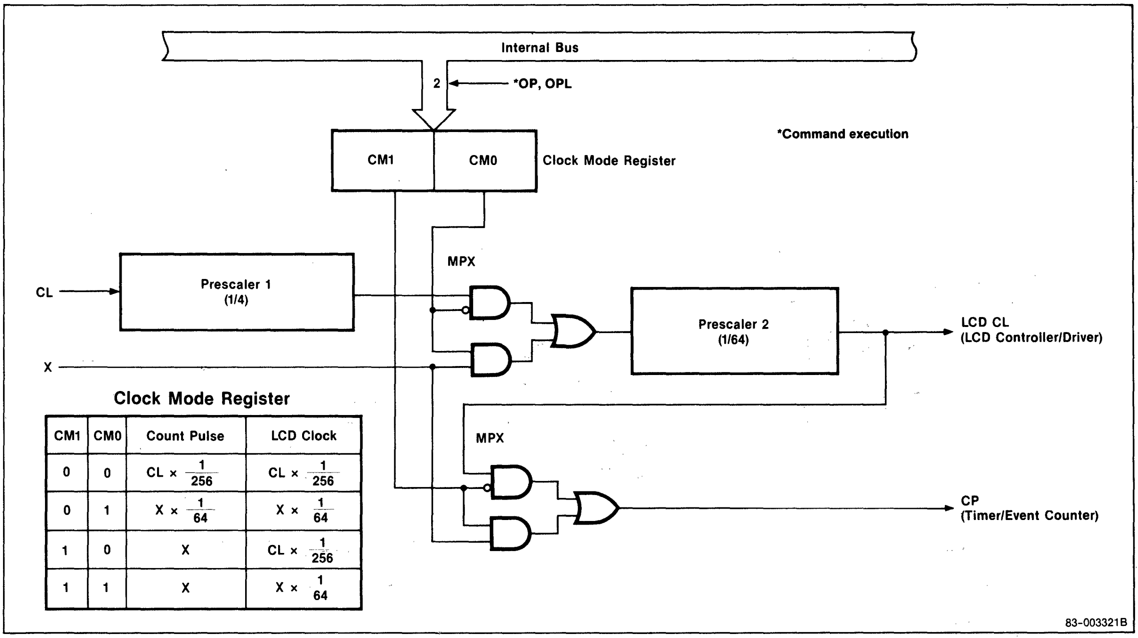
Type B
INT1, $P0_3$/SI, $P1_0$/INT0, RESET

Type D
$P3_0$-$P3_3$

Type E
$P0_2$/SO, $P4_0$-$P4_3$, $P5_0$-$P5_3$, $P6_0$-$P6_3$

Type F
$P0_1$/$\overline{SCK}$

Figure 4. Clock Control



Clock Mode Register

CM1	CM0	Count Pulse	LCD Clock
0	0	CL × $\frac{1}{256}$	CL × $\frac{1}{256}$
0	1	X × $\frac{1}{64}$	X × $\frac{1}{64}$
1	0	X	CL × $\frac{1}{256}$
1	1	X	X × $\frac{1}{64}$

Figure 5. Timer/Event Counter

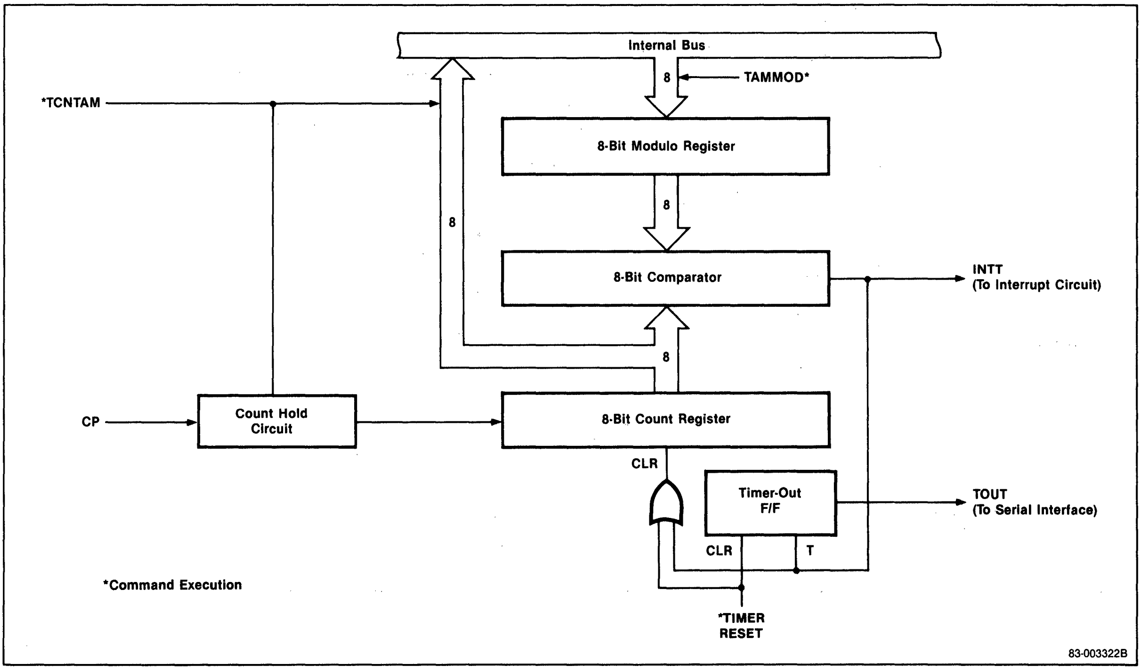


Figure 6. Interrupt Control

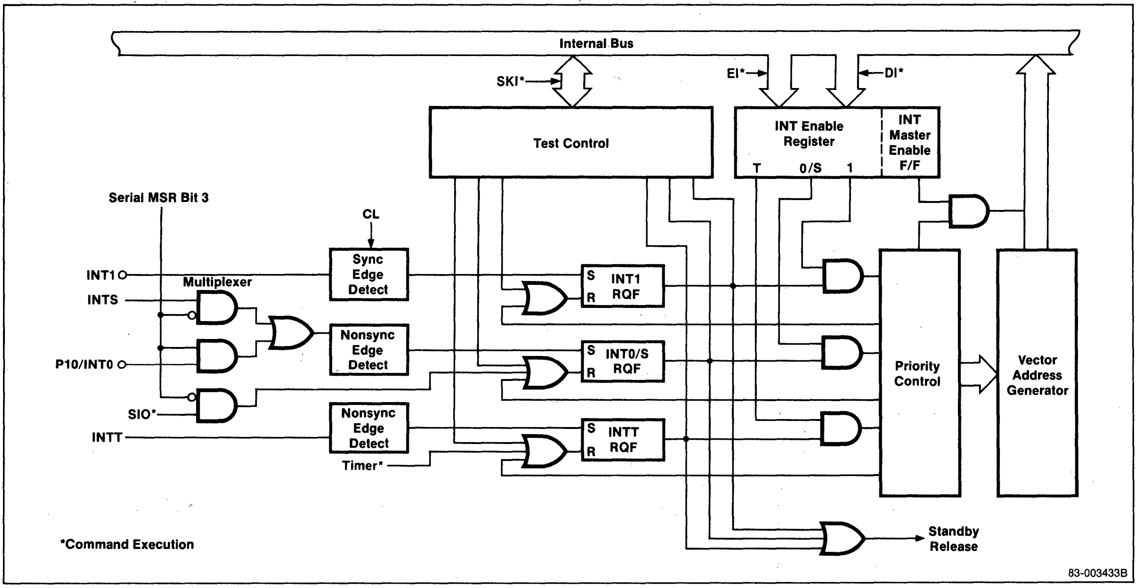


Figure 7. Serial Interface

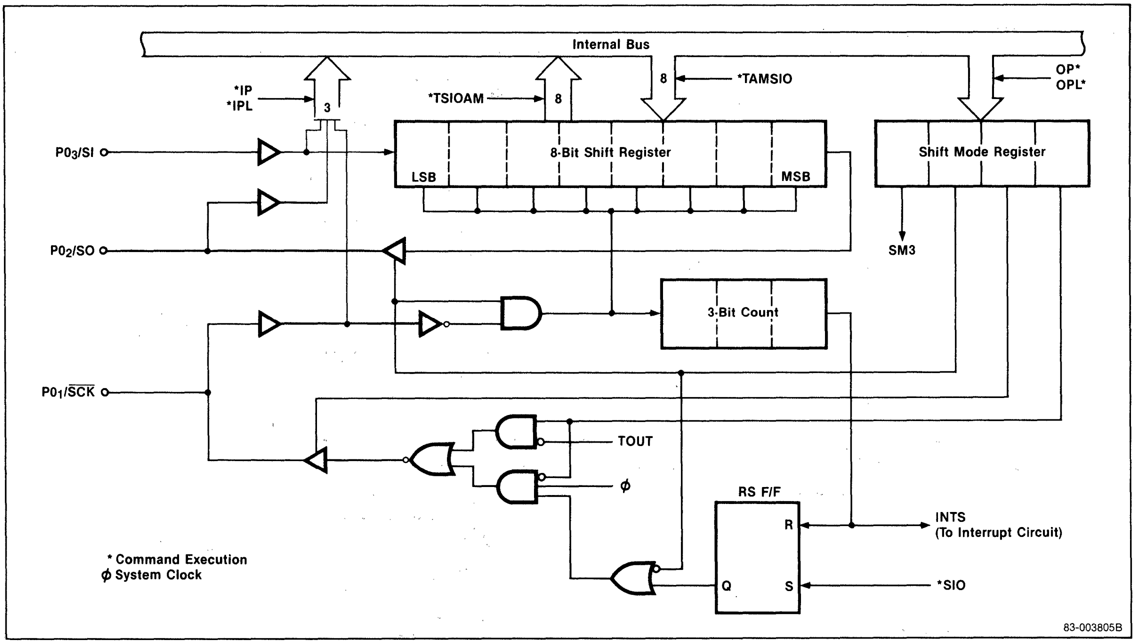
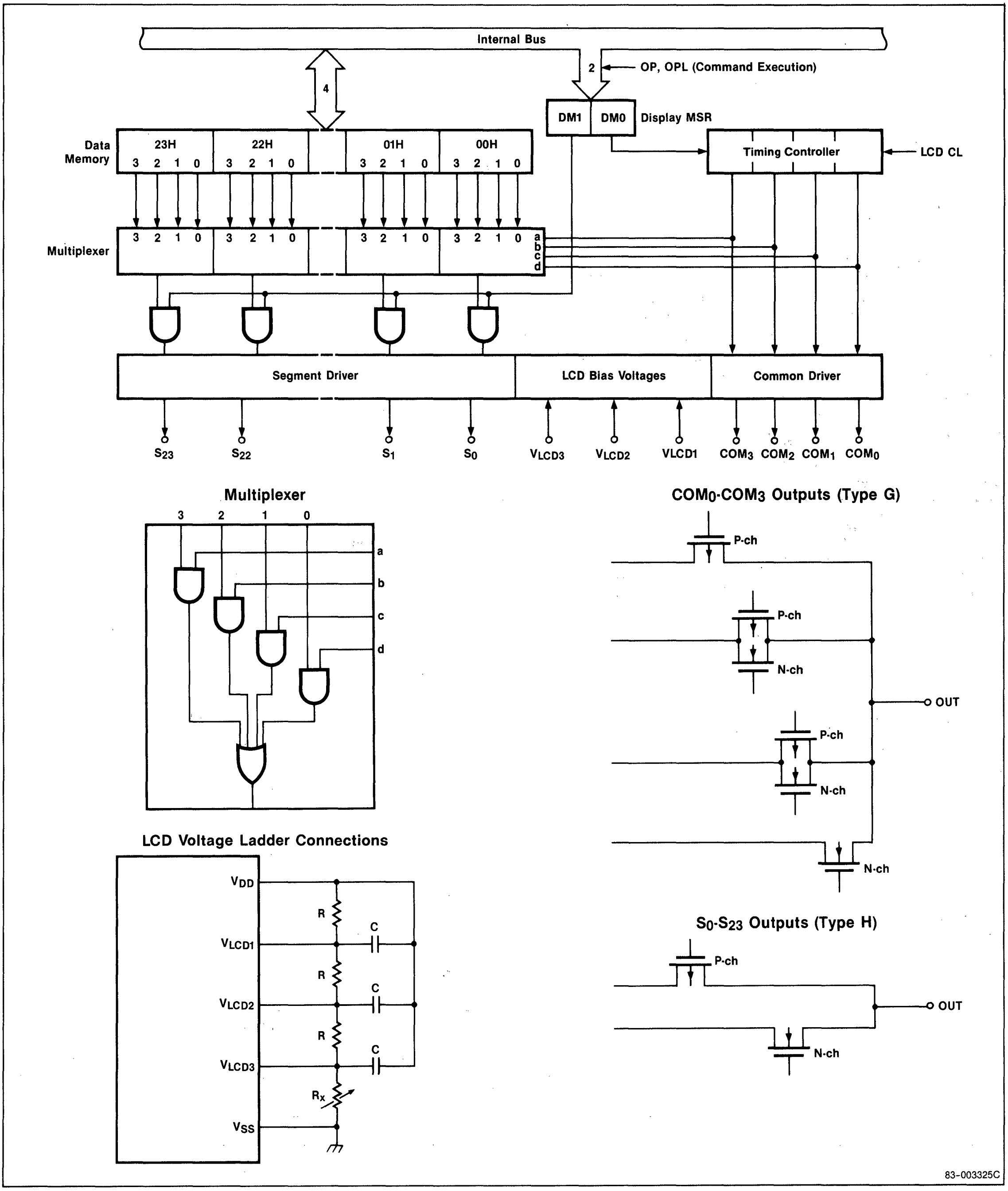


Figure 8. LCD Controller/Driver



Absolute Maximum Ratings

$T_A = 25°C$

Power supply voltage, V_{DD}	−0.3 to +7.0 V
All input and output voltages	−0.3 V to V_{DD} + 0.3 V
Output current high, I_{OH}	
Per pin	−17 mA
Total, output ports	−20 mA
Output current low, I_{OL}	
Per pin	17 mA
Total, output ports	55 mA
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

$T_A = 25°C$; $V_{DD} = 0$ V

		Limits				Test
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I			15	pF	$f_C = 1$ MHz Unmeasured pins returned to V_{SS}
Output capacitance	C_O			15	pF	
I/O capacitance	C_{IO}			15	pF	

DC Characteristics 1

For V_{DD} = 2.5 to 3.3 Volts

$T_A = -10$ to +70°C

		Limits				Test
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, high	V_{IH1}	0.8 V_{DD}		V_{DD}	V	Except CL1, X1
	V_{IH2}	V_{DD} − 0.3		V_{DD}	V	CL1, X1
	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	RESET, data retention mode
Input voltage, low	V_{IL1}	0		0.2 V_{DD}	V	Except CL1, X1
	V_{IL2}	0		0.3	V	CL1, X1
Output voltage, high	V_{OH}	V_{DD} − 0.5			V	$I_{OH} = -80$ μA
Output voltage, low	V_{OL}			0.5	V	$I_{OL} = 350$ μA
Input leakage current, high	I_{LIH1}			3	μA	Except CL1, X1; $V_{IN} = V_{DD}$
	I_{LIH2}			10	μA	CL1, X1; $V_{IN} = V_{DD}$
Input leakage current, low	I_{LIL1}			−3	μA	Except CL1, X1; $V_{IN} = 0$ V
	I_{LIL2}			−10	μA	CL1, X1; $V_{IN} = 0$ V
Output leakage current, high	I_{LOH}			3	μA	$V_O = V_{DD}$
Output leakage current, low	I_{LOL}			−3	μA	$V_O = 0$ V
Supply voltage	V_{DDDR}	2.0			V	Data retention mode
Supply current	I_{DD1}		50	250	μA	Normal operation, $V_{DD} = 3$ V ± 10%; R = 240 kΩ ±2%, C = 33 pF ±5%
			35	230	μA	Normal operation, $V_{DD} = 2.5$ V; R = 240 kΩ ±2%, C = 33 pF ±5%
	I_{DD2}		0.3	10	μA	Stop mode, X1 = 0 V; $V_{DD} = 3$ V ±10%
			0.2	10	μA	Stop mode, X1 = 0 V; $V_{DD} = 2.5$ V
	I_{DDDR}		0.2	10	μA	Data retention mode, $V_{DDDR} = 2.0$ V

DC Characteristics 2

For V_{DD} = 2.7 to 6.0 Volts

T_A = −10 to +70°C

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Except CL1, X1
	V_{IH2}	$V_{DD} - 0.5$		V_{DD}	V	CL1, X1
	V_{IHDR}	0.9 V_{DDDR}		$V_{DDDR} + 0.2$		RESET, data retention mode
Input voltage, low	V_{IL1}	0		0.3 V_{DD}	V	Except CL1, X1
	V_{IL2}	0		0.5	V	CL1, X1
Output voltage, high	V_{OH}	$V_{DD} - 1.0$			V	$I_{OH} = -1.0$ mA, V_{DD} = 4.5 to 6.0 V
		$V_{DD} - 0.5$			V	$I_{OL} = -100$ μA
Output voltage, low	V_{OL}			0.4	V	I_{OL} = 1.6 mA, V_{DD} = 4.5 to 6.0 V
				0.5	V	I_{OL} = 400 μA
Input leakage current, high	I_{LIH1}			3	μA	Except CL1, X1; $V_I = V_{DD}$
	I_{LIH2}			10	μA	CL1, X1
Input leakage current, low	I_{LIL1}			−3	μA	Except CL1, X1; V_I = 0 V
	I_{LIL2}			−10	μA	CL1, X1
Output leakage current, high	I_{LOH}			3	μA	$V_0 = V_{DD}$
Output leakage current, low	I_{LOL}			−3	μA	V_0 = 0 V
Output impedance (1)	R_{COM}		3	5	kΩ	COM_0-COM_3; V_{DD} = 4.5 to 6.0 V
			5	15	kΩ	COM_0-COM_3
	R_S		15	20	kΩ	S_0-S_{23}; V_{DD} = 4.5 to 6.0 V
			20	60	kΩ	S_0-S_{23}
Supply voltage	V_{DDDR}	2.0		6.0	V	Data retention mode
Supply current	I_{DD1}		300	900	μA	Normal operation, V_{DD} = 5 V ± 10%; R = 82 kΩ ± 2%, C = 33 pF ± 5%
			70	300	μA	Normal operation, V_{DD} = 3 V ± 10%; R = 160 kΩ ± 2%, C = 33 pF ± 5%
	I_{DD2}		1.0	20	μA	Stop mode, X1 = 0 V; V_{DD} = 5 V ± 10%
			0.3	10	μA	Stop mode, X1 = 0 V; V_{DD} = 3 V ± 10%
	I_{DDDR}		0.2	10	μA	Data retention mode, V_{DDDR} = 2.0 V

Note:

(1) V_{LCD} = 2.7 V to V_{DD}
$V_{LCD1} = V_{DD} - (1/3)\ V_{LCD}$
$V_{LCD2} = V_{DD} - (2/3)\ V_{LCD}$
$V_{LCD3} = V_{DD} - V_{LCD}$

AC Characteristics 1

For $V_{DD} = 2.7$ to 6.0 Volts

$T_A = -10$ to $+70\,°C$

			Limits			Test
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
System clock frequency	f_{CC}	150	200	240	kHz	$V_{DD} = 5$ V ±10%; R = 82 kΩ ±2% (Note 1)
		75	100	120	kHz	$V_{DD} = 3$ V ±10%; R = 160 kΩ ±2% (Note 1)
		75		135	kHz	R = 160 kΩ ±2% (Note 1)
	f_C	10		410	kHz	CL1, external clock, 50% duty; $V_{DD} = 4.5$ to 6.0 V
		10		125	kHz	CL1, external clock, 50% duty; $V_{DD} = 2.7$ V
System clock rise and fall time	t_{CR}, t_{CF}			0.2	µs	CL1, external clock
System clock pulse width	t_{CH}, t_{CL}	1.2		50	µs	CL1, external clock; $V_{DD} = 4.5$ to 6.0 V
		4.0		50	µs	CL1, external clock; $V_{DD} = 2.7$ V
Counter clock frequency	f_{XX}	25	32	50	kHz	X1, X2, crystal oscillator
	f_X	0		410	kHz	X1, external pulse input, 50% duty; $V_{DD} = 4.5$ to 6.0 V
		0		125	kHz	X1, external pulse input, 50% duty; $V_{DD} = 2.7$ V
Counter clock rise and fall time	t_{XR}, t_{XF}			0.2	µs	X1, external pulse input
Counter clock pulse width	t_{XH}, t_{XL}	1.2			µs	X1, external pulse input; $V_{DD} = 4.5$ to 6.0 V
		4.0			µs	X1, external pulse input; $V_{DD} = 2.7$ V
$\overline{\text{SCK}}$ cycle time	t_{KCY}	3.0			µs	$\overline{\text{SCK}}$ as input; $V_{DD} = 4.5$ to 6.0 V
		8.0			µs	$\overline{\text{SCK}}$ as input
		4.9			µs	$\overline{\text{SCK}}$ as output; $V_{DD} = 4.5$ to 6.0 V
		16.0			µs	$\overline{\text{SCK}}$ as output
$\overline{\text{SCK}}$ pulse width	t_{KH}, t_{KL}	1.3			µs	$\overline{\text{SCK}}$ as input; $V_{DD} = 4.5$ to 6.0 V
		4.0			µs	$\overline{\text{SCK}}$ as input
		2.2			µs	$\overline{\text{SCK}}$ as output; $V_{DD} = 4.5$ to 6.0 V
		8.0			µs	$\overline{\text{SCK}}$ as output
SI setup time to $\overline{\text{SCK}}$ ↑	t_{SIK}	300			ns	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{KSI}	450			ns	
SO delay time after $\overline{\text{SCK}}$ ↓	t_{KSO}			850	ns	$V_{DD} = 4.5$ V to 6.0 V
				1200	ns	
INT0 pulse width	t_{I0H}, t_{I0L}	10			µs	
INT1 pulse width	t_{I1H}, t_{I1L}	(Note 2)			µs	
RESET pulse width	t_{RSH}, t_{RSL}	10			µs	
RESET setup time	t_{SRS}	0			ns	
RESET hold time	t_{HRS}	0			ns	

Notes:

(1) RC network at CL1 and CL2; C = 33 pF ±5%, $|\Delta C/°C| \leqq 60$ ppm.

(2) $2 \times 10^3 \div f_{cc}$ or f_c in kHz.

AC Characteristics 2

For V_{DD} = 2.5 to 3.3 Volts

T_A = −10 to +70°C

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
System clock frequency	f_{CC}	50		80	kHz	R = 240 kΩ ±2% (Note 1)
		50	64	77	kHz	V_{DD} = 2.5 V; R = 240 kΩ ±2% (Note 1)
	f_C	10		80	kHz	CL1, external clock, 50% duty
System clock rise and fall time	t_{CR}, t_{CF}			0.2	μs	CL1, external clock
System clock pulse width	t_{CH}, t_{CL}	6.25		50	μs	CL1, external clock
Counter clock frequency	f_{XX}	25	32	50	kHz	X1, X2, crystal oscillator
	f_X	0		80	kHz	X1, external pulse input, 50% duty
Counter clock rise and fall time	t_{XR}, t_{XF}			0.2	μs	X1, external pulse input
Counter clock pulse width	t_{XH}, t_{XL}	6.25			μs	X1, external pulse input
$\overline{SCK}$ cycle time	t_{KCY}	12.5			μs	$\overline{SCK}$ as input
		25			μs	$\overline{SCK}$ as output
$\overline{SCK}$ pulse width	t_{KH}, t_{KL}	6.25			μs	$\overline{SCK}$ as input
		11.5			μs	$\overline{SCK}$ as output
SI setup time to $\overline{SCK}$ ↑	t_{SIK}	1			μs	
SI hold time after $\overline{SCK}$ ↑	t_{KSI}	1			μs	
SO delay time after $\overline{SCK}$ ↓	t_{KSO}			2	μs	
INT0 pulse width	t_{I0H}, t_{I0L}	30			μs	
INT1 pulse width	t_{I1H}, t_{I1L}	(Note 2)			μs	
RESET pulse width	t_{RSH}, t_{RSL}	30			μs	

Notes:

(1) RC network at CL1 and CL2; C = 33 pF ±5%, $|\Delta C/°C| \leqq$ 60 ppm.

(2) 2 x 10^3 ÷ f_{cc} or f_c in kHz.

Recommended R and C Values for System Clock Oscillation Circuit

T_A = −10 to +70°C

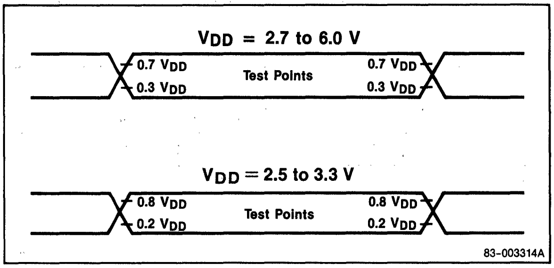
Supply Voltage Range	Recommended Values (Note 1)	Frequency Range
4.5 to 6.0 V	R = 82 kΩ ±2%	150 to 250 kHz, 200 kHz typical
2.7 to 3.3 V	R =160 kΩ ±2%	75 to 120 kHz, 100 kHz typical
2.7 to 6.0 V	R =160 kΩ± 2%	75 to 135 kHz
2.5 to 3.3 V	R = 240 kΩ ±2%	50 to 80 kHz
2.5 to 6.0 V	R = 240 kΩ ±2 %	50 to 85 kHz

Note:

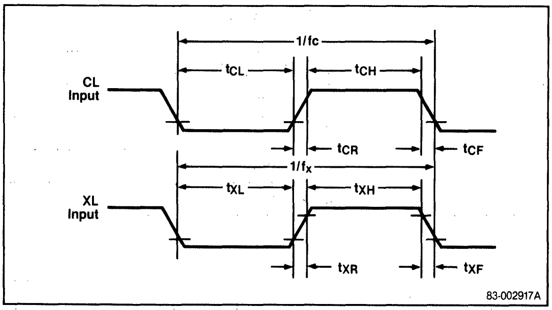
(1) C = 33 pF ±5%, $|\Delta C/°C| \leq$ 60 ppm.

Timing Waveforms

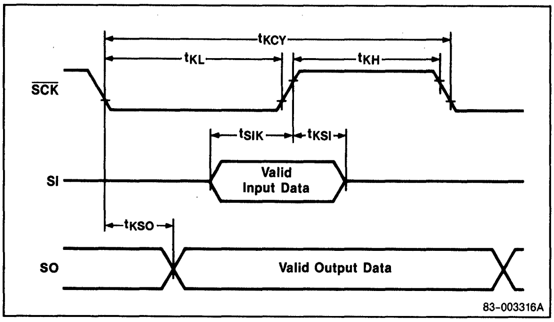
Timing Measurement Points



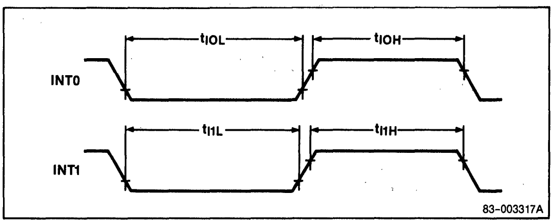
Clock Timing



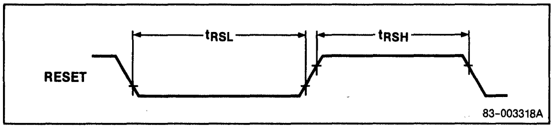
Serial Interface



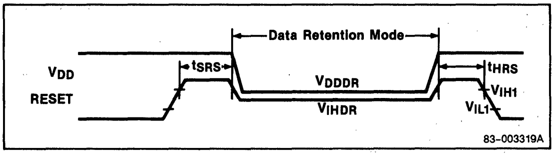
External Interrupts



Reset



Data Retention Mode

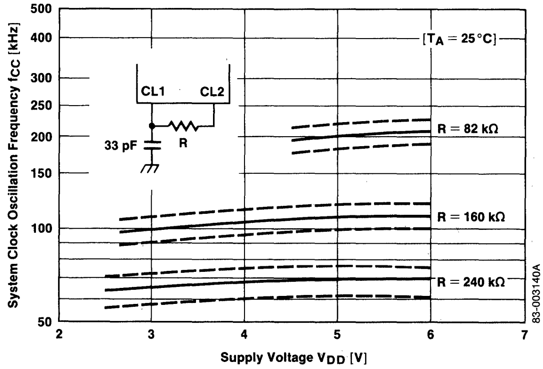


Operating Characteristics

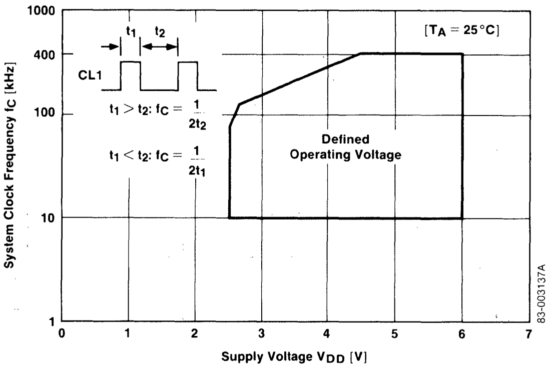
$T_A = 25°C$

3

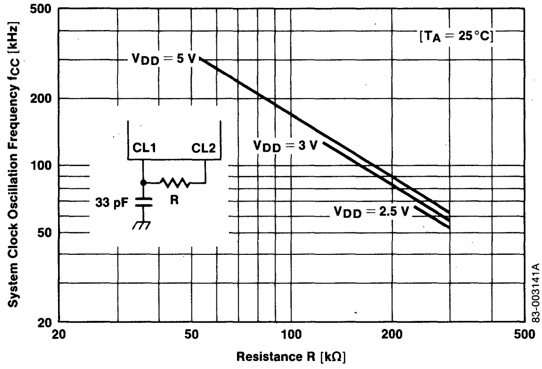
f_{CC} vs V_{DD}



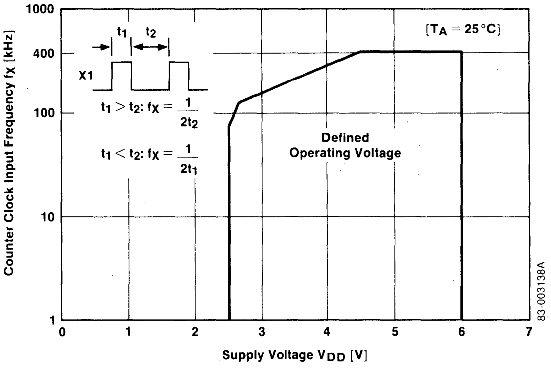
f_C vs V_{DD} [External Clock]



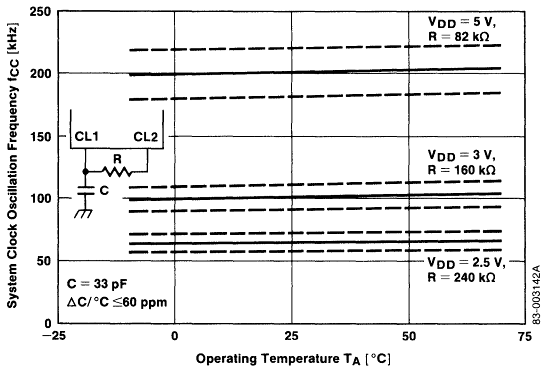
f_{CC} vs R



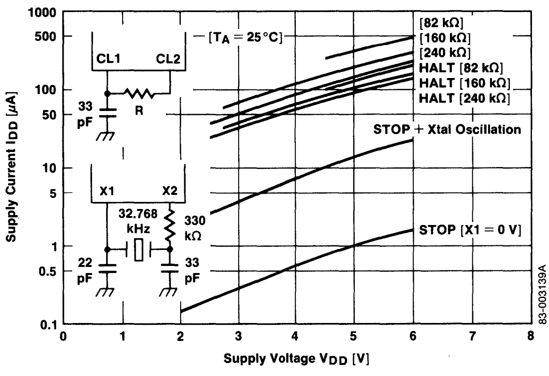
f_X vs V_{DD} [External Clock]



f_{CC} vs T_A

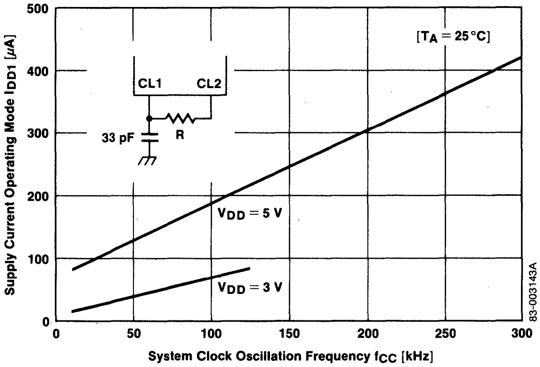


I_{DD} vs V_{DD}

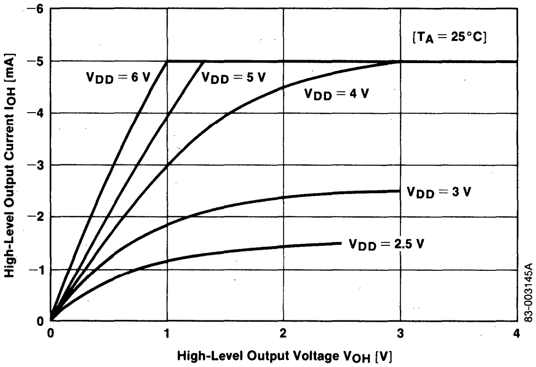


Operating Characteristics (cont)

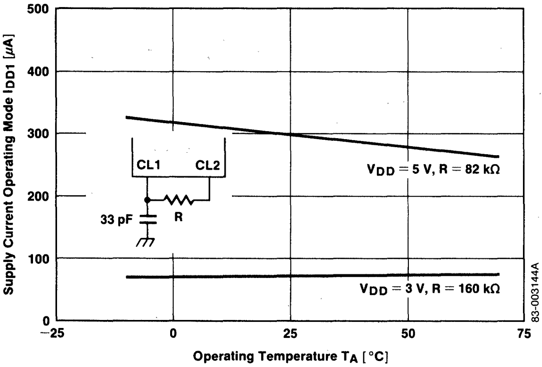
I_{DD1} vs f_{CC}



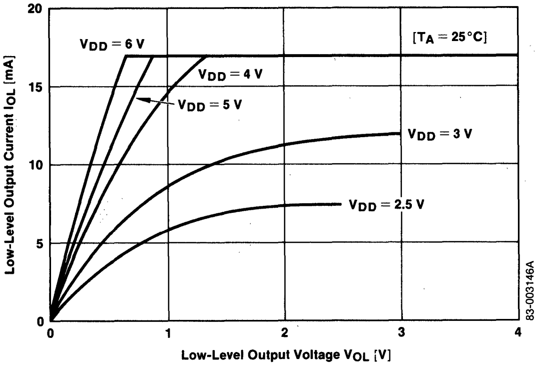
I_{OH} vs V_{OH}



I_{DD1} vs T_A



I_{OL} vs V_{OL}



NEC
NEC Electronics Inc.

μPD7507/08 4-Bit, Single-Chip CMOS Microcomputers

Description

The μPD7507 and μPD7508 4-bit, single-chip CMOS microcomputers have the μPD7500 series architecture. The subroutine stack is implemented in RAM for greater nesting depth and flexibility.

Thirty-two I/O lines are organized into eight 4-bit ports: input port/serial interface port 0, output ports 2 and 3, and I/O ports 1, 4, 5, 6, and 7.

The μPD7507 and μPD7508 execute 92 instructions of the μPD7500 series A instruction set with a 5-μs instruction cycle time.

Maximum power consumption is 900 μA at 5 V, less in the HALT and STOP low-power modes.

The μPD75CG08E is a piggyback EPROM prototyping chip that is pin-compatible with μPD7507 and μPD7508. A 2716 inserted into the top of the μPD75CG08E emulates the μPD7507's ROM. A 2732 emulates the μPD7508's ROM. When emulating the μPD7507, the user must take care to use only the first 128 RAM locations. Although the μPD7507 and μPD7508 can operate over a range of 2.5 to 6.0 V, μPD75CG08E operation is limited to 5 V ±10%.

Table 1 summarizes the differences among μPD7507, μPD7508 and μPD75CG08E.

Table 1. Features Comparison

	μPD75CG08E	μPD7507/7508
Program memory	2K x 8 EPROM (2716) 4K x 8 EPROM (2732)	2K x 8 masked ROM (7507) 4K x 8 masked ROM (7508)
Data memory	224 x 4	128 x 4 (7507) 224 x 4 (7508)
Data retention mode	No	Yes
Power supply	5 V ±10%	2.7 to 6.0 V
Package types	40-pin ceramic piggyback DIP	40-pin plastic DIP 40-pin plastic shrink DIP 52-pin plastic QFP

Features

□ Single chip microcomputer
□ Program ROM
— μPD7507: 2048 x 8-bit
— μPD7508: 4096 x 8-bit
— μPD75CG08: piggyback EPROM
□ Data RAM
— μPD7507: 128 x 4-bit
— μPD7508: 224 x 4-bit
— μPD75CG08: 224 x 4-bit
□ 8-bit timer/event counter
□ Four 4-bit general purpose registers
□ Four vectored, prioritized interrupts
□ Executes 92 instructions of μPD7500 series A instruction set
□ 5 μs instruction cycle/400 kHz external clock
□ Two standby modes
□ 32 I/O lines
□ Low-power HALT and STOP modes

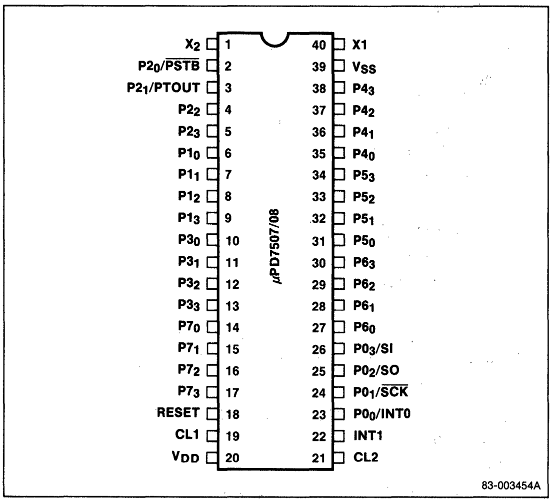
Ordering Information

*Part Number	Package Type	Max Frequency of Operation
μPD7507C	40-pin plastic DIP	410 kHz
μPD7507CU	40-pin plastic shrink DIP	410 kHz
μPD7507GC-00	52-pin plastic QFP	410 kHz
μPD7508C	40-pin plastic DIP	410 kHz
μPD7508CU	40-pin plastic shrink DIP	410 kHz
μPD7508GC-00	52-pin plastic QFP	410 kHz
μPD75CG08E	40-pin ceramic piggyback DIP	410 kHz

* A 3-digit mask identification code is added to the part number by NEC at the time of code verification.

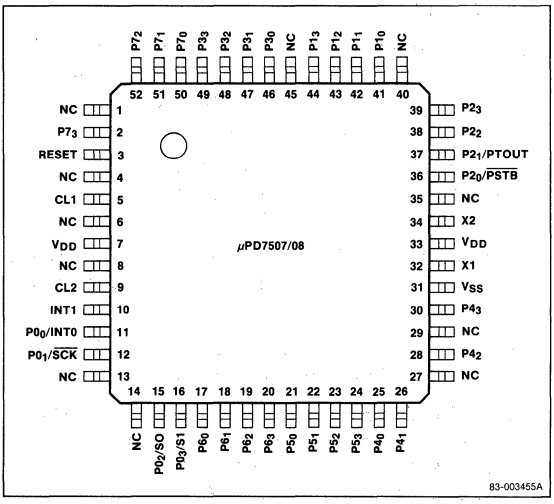
Pin Configurations

40-Pin Plastic DIP and Plastic Shrink DIP



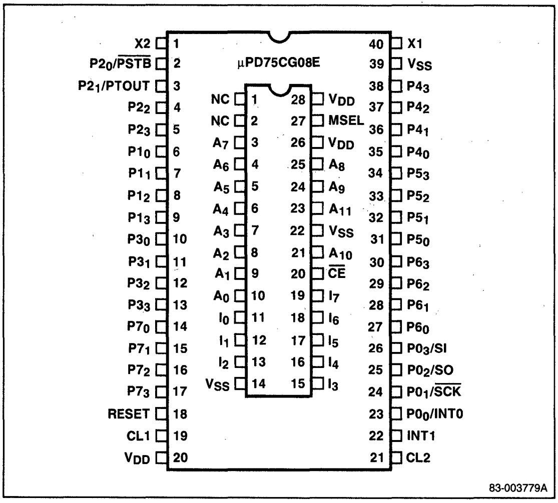
83-003454A

52-Pin Plastic QFP



83-003455A

40-Pin Ceramic Piggyback DIP



83-003779A

Pin Identification

40-Pin DIP, Shrink DIP and Piggyback DIP

No.	Symbol	Function
1, 40	X2, X1	Crystal clock/external event input port
2-5	$P2_0$/$\overline{PSTB}$, $P2_1$/PTOUT, $P2_2$, $P2_3$	Output port 2/output strobe pulse, timer out F/F signal
6-9	$P1_0$-$P1_3$	I/O port 1
10-13	$P3_0$-$P3_3$	Output port 3
14-17	$P7_0$-$P7_3$	I/O port 7
18	RESET	RESET input
19, 21	CL1, CL2	System clock inputs
20	V_{DD}	Positive power supply
22	INT1	External interrupt
23-26	$P0_0$/INT0, $P0_1$/$\overline{SCK}$, $P0_2$/SO, $P0_3$/SI	Input port 0/external interrupt, serial I/O interface
27-30	$P6_0$-$P6_3$	I/O port 6
31-34	$P5_0$-$P5_3$	I/O port 5
35-38	$P4_3$-$P4_0$	I/O port 4
39	V_{SS}	Ground

Pin Identification (cont)

28-Pin EPROM Socket on Piggyback DIP

No.	Symbol	Function
1, 2	NC	Not connected
3-10	A_7-A_0	Address bits 7-0
11-13	I_0-I_2	Data bits 0-2
14, 22	V_{SS}	Ground
15-19	I_3-I_7	Data bits 3-7
20	$\overline{CE}$	Chip enable
21, 23	A_{10}-A_{11}	Address bits 10, 11
24, 25	A_9, A_8	Address bits 9, 8
26, 28	V_{DD}	Positive power supply
27	MSEL	Memory select

52-Pin QFP

No.	Symbol	Function
1, 4, 6, 8, 13, 14, 27, 29, 35, 40, 45	NC	Not connected
2, 50-52	$P7_0$-$P7_3$	I/O port 7
3	RESET	RESET input
5, 9	CL1, CL2	System clock inputs
7	V_{DD}	Positive power supply
10	INT1	External interrupt
11, 12, 15, 16	$P0_0$/INT0, $P0_1/\overline{SCK}$, $P0_2$/SO, $P0_3$/SI	Input port 0/external interrupt, serial I/O interface
17-20	$P6_0$-$P6_3$	I/O port 6
21-24	$P5_0$-$P5_3$	I/O port 5
25, 26 28, 30	$P4_3$-$P4_0$	I/O port 4
31	V_{SS}	Ground
32, 34	X1, X2	Crystal clock/external event input
33	V_{DD}	Positive power supply
36-39	$P2_0/\overline{PSTB}$, $P2_1$/PTOUT, $P2_2$, $P2_3$	4-bit output port 2/output strobe pulse, timer out F/F signal
41-44	$P1_0$-$P1_3$	I/O port 1
46-49	$P3_0$-$P3_3$	Output port 3

Pin Functions

$P0_0$/INT0, $P0_1/\overline{SCK}$, $P0_2$/SO, $P0_3$/SI [Port 0/ External Interrupt, Serial Interface]

4-bit input port/serial I/O interface. This port can be configured as a 4-bit parallel input port or as the 8-bit serial I/O interface under control of the serial mode select register. The serial input SI, serial output SO (active low), and the serial clock $\overline{SCK}$ (active low), used for synchronizing data transfer, make up the 8-bit serial I/O interface. Line $P0_0$ is always shared with external interrupt INT0, a rising edge-triggered interrupt. If $P0_0$/INT0 is unused, it should be connected to V_{SS}. If $P0_1/\overline{SCK}$, $P0_2$/SO, or $P0_3$/SI are unused, connect them to V_{SS} or V_{DD}.

$P1_0$-$P1_3$ [Port 1]

4-bit input/three-state output port. Data output to port 1 is strobed in synchronization with a $P2_0/\overline{PSTB}$ pulse. Connect unused pins to V_{SS} or V_{DD}.

$P2_0/\overline{PSTB}$, $P2_1$/PTOUT, $P2_2$, $P2_3$ [Port 2]

4-bit latched three-state output port. Line $P2_0$ is shared with $\overline{PSTB}$, the port 1 output strobe pulse. Line $P2_1$ is shared with PTOUT, the timer out F/F signal. Leave unused pins open.

$P3_0$-$P3_3$ [Port 3]

4-bit latched three-state output port. Leave unused pins open.

$P4_0$-$P4_3$ [Port 4]

4-bit latched three-state output port. Can also perform 8-bit parallel I/O with port 5. In input mode, connect unused pins to V_{DD} or GND. In output mode, leave unused pins open.

$P5_3$-$P5_0$ [Port 5]

4-bit input/latched three-state output port. This port also performs 8-bit parallel I/O with port 4. In input mode, connect unused pins to V_{SS} or V_{DD}. In output mode, leave unused pins open.

$P6_3$-$P6_0$ [Port 6]

4-bit input/latched three-state output port. The port 6 mode select register configures individual lines as inputs or outputs. In input mode, connect unused pins to V_{SS} or V_{DD}. In output mode, leave unused pins open.

$P7_0$-$P7_3$ [Port 7]

4-bit input/latched three state output port. In input mode, connect unused pins to V_{SS} or V_{DD}. In output mode, leave unused pins open.

X2, X1 [Crystal Clock/External Event Input]

Connect a crystal oscillator circuit to input X1 and output X2 for crystal clock operation. Alternatively, connect external event pulses to input X1 and leave output X2 open for external event counting. If X1 is not used, connect it to ground. If X2 is not used, leave it open.

CL1, CL2 [System Clock Input]

Connect a 82 kΩ resistor across CL1 and CL2, and connect a 33 pF capacitor from CL1 to V_{SS}. Alternatively, connect an external clock source to CL1 and leave CL2 open.

RESET [Reset]

A high level input to this pin initializes the μPD7507/08 after power up.

INT1 [Interrupt 1]

External rising edge-triggered interrupt. Connect to V_{SS} if unused.

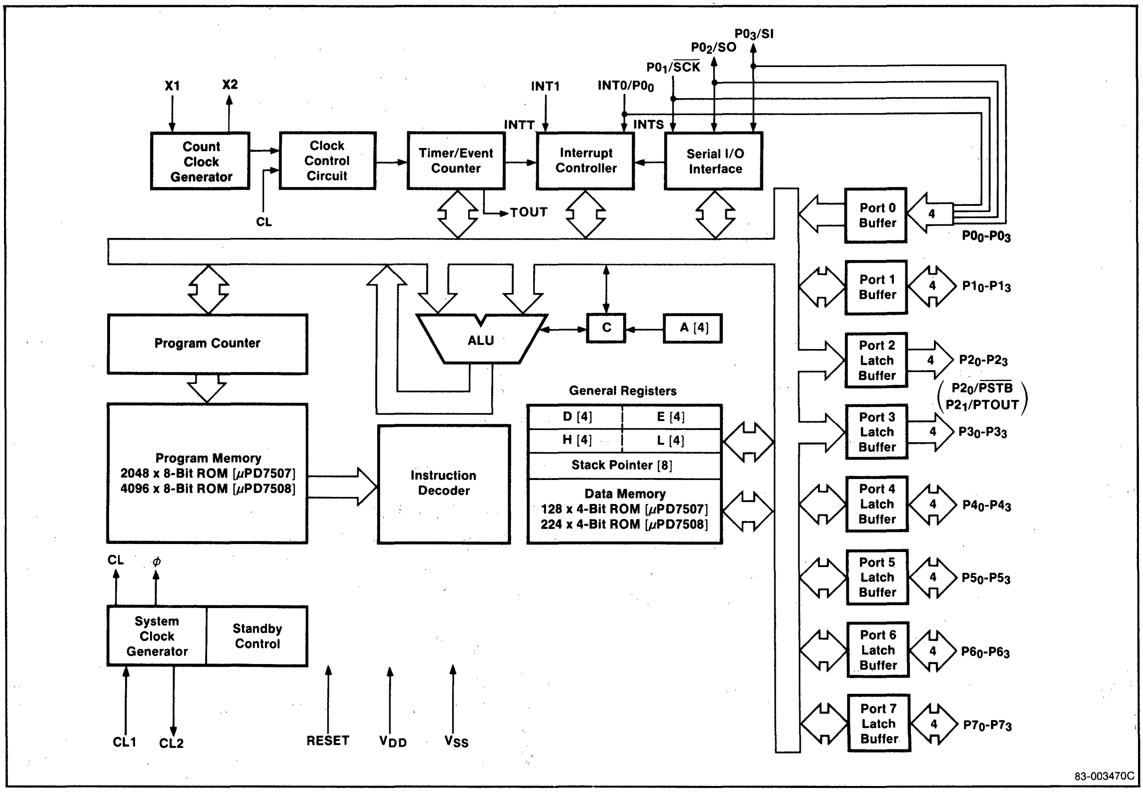
V_{DD} [Power Supply]

Positive power supply. Apply a single voltage in the range 2.7 to 6.0 V for proper operation.

V_{SS} [Ground]

Ground.

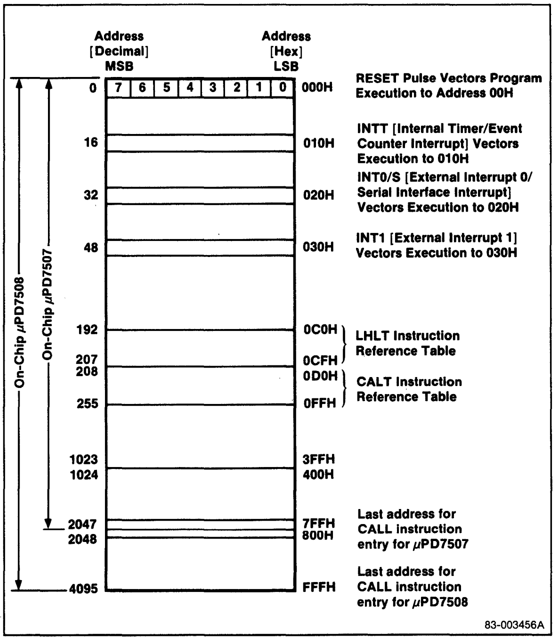
Block Diagram



Memory Map

Figure 1 shows the ROM memory map of the µPD7507/08.

Figure 1. ROM Map


Address
[Decimal]
MSB
Address
[Hex]
LSB
7 6 5 4 3 2 1 0
0
000H
RESET Pulse Vectors Program Execution to Address 00H
16
010H
INTT [Internal Timer/Event Counter Interrupt] Vectors Execution to 010H
32
020H
INT0/S [External Interrupt 0/ Serial Interface Interrupt] Vectors Execution to 020H
48
030H
INT1 [External Interrupt 1] Vectors Execution to 030H
192
0C0H
207
0CFH
LHLT Instruction Reference Table
208
0D0H
255
0FFH
CALT Instruction Reference Table
1023
3FFH
1024
400H
2047
7FFH
2048
800H
Last address for CALL instruction entry for µPD7507
4095
FFFH
Last address for CALL instruction entry for µPD7508
On-Chip µPD7508
On-Chip µPD7507
83-003456A

Clock Control Circuit

The clock control circuit consists of a 4-bit clock mode register (bits CM_1 and CM_2), prescalers 1, 2, and 3, and a multiplexer. It takes the output of the system clock generator (CL) and count clock generator circuit (X). It also selects the clock source and divides the signal according to the setting in the clock mode register. It outputs the count pulse (CP) to the timer/event counter. Figure 2 shows the clock control circuit.

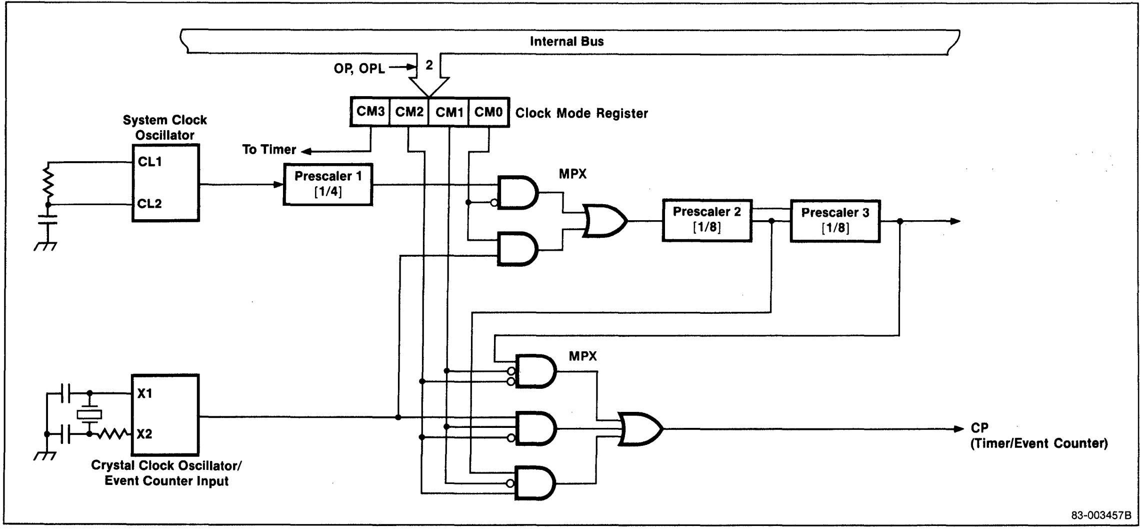
Table 2 lists the codes set in the clock mode register by the OP or OPL instruction to specify the count pulse frequency.

Table 2. Selecting the Count Pulse Frequency

CM_2	CM_1	CM_0	Frequency Selected
0	0	0	CL/256
0	0	1	X/64
0	1	0	X
0	1	1	X
1	0	0	CL/32
1	0	1	X/8
1	1	0	Not used
1	1	1	Not used

CM3	TOUT Signal
0	Disabled
1	Enabled

Figure 2. Clock Control Circuit



Timer/Event Counter

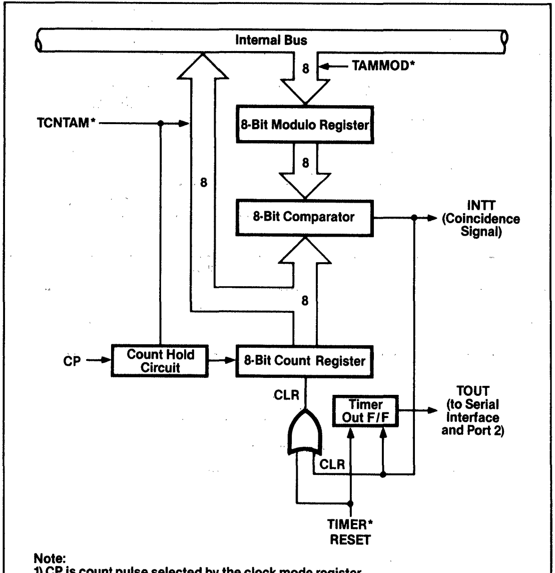
The timer/event counter consists of an 8-bit counter, an 8-bit modulo register, an 8-bit comparator, and a timer out flip-flop as shown in figure 3.

The 8-bit count register is a binary 8-bit up-counter which is incremented each time a count pulse is input. The TIMER instruction, a RESET signal, or an INTT coincidence signal clears it to 00H.

The 8-bit modulo register determines the number of counts the count register holds. The TAMMOD instruction loads the contents of the modulo register. RESET sets the modulo register to FFH.

The 8-bit comparator compares the contents of the count register and the modulo register and outputs an INTT when they are equal.

Figure 3. Timer/Event Counter



Note:
1) CP is count pulse selected by the clock mode register.
2) *Execution of instruction.

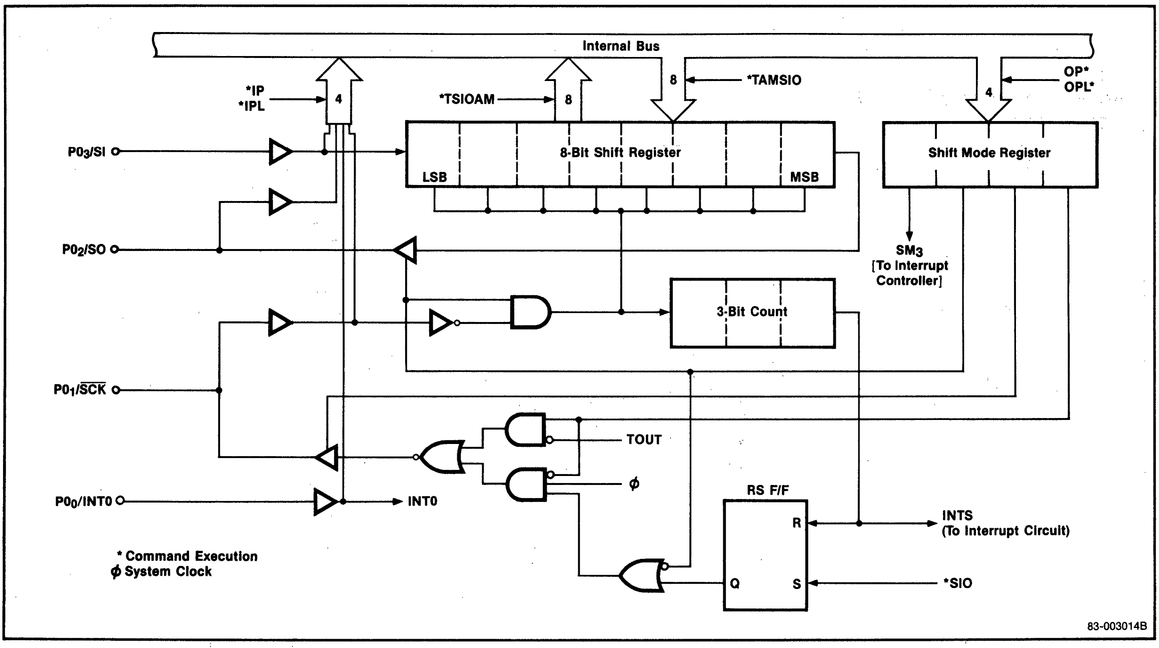
83-003600A

Serial Interface

The 8-bit serial interface allows the μPD7507/08 to communicate with peripheral devices such as the μPD7001 A/D converter, the μPD7227 dot matrix LCD controller/driver, and other microprocessors or microcomputers. Figure 4 shows the serial interface.

The serial interface consists of an 8-bit shift register, a 3-bit $\overline{SCK}$ pulse counter, the SI input port, the SO output port, the $\overline{SCK}$ serial clock I/O port, and a 4-bit serial mode select register (MSR). The MSR selects serial I/O or port 0 operation.

Figure 4. Serial Interface



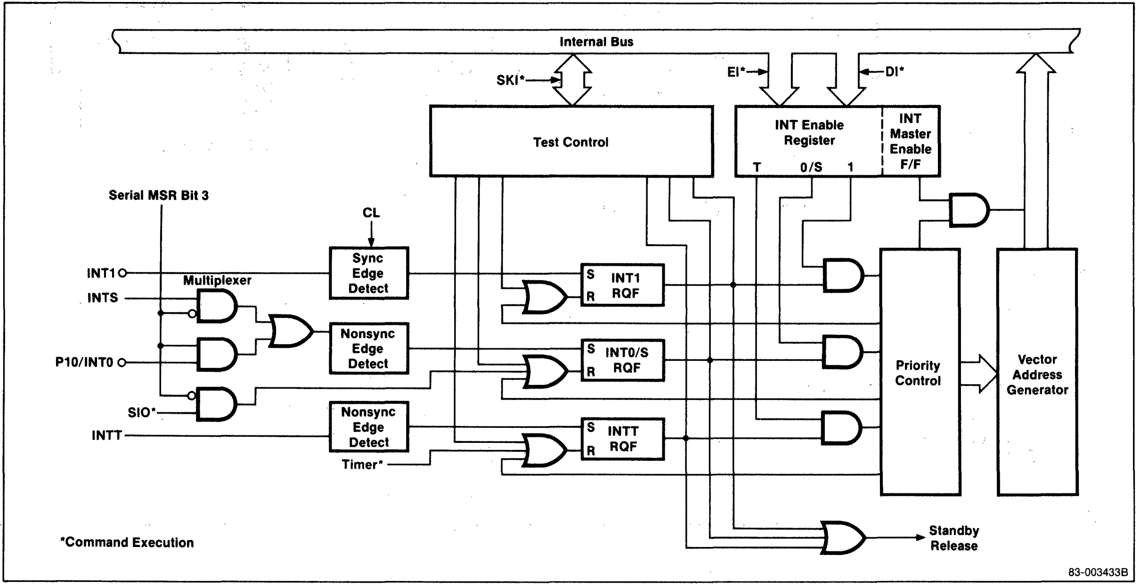
Interrupts

The µPD7507/08 has four vectored, prioritized interrupts. Two of these interrupts, INTT and INTS, are internally generated from the timer/event counter and serial interface, respectively. INT0 and INT1 are externally generated. Table 3 is a summary of the four interrupts. Figure 5 is the block diagram.

Table 3. µPD7507/08 Interrupts

Source	Function	Location	Priority	ROM Vector Address
INTT	Coincidence in timer/event counter	Internal	1	10H
INTS	Transfer complete signal from serial interface	Internal	2	20H
INT0	INT0 pin	External	2	20H
INT1	INT1 pin	External	3	30H

Figure 5. Interrupt Block Diagram



System Clock and Timing Circuitry

Timing for the μPD7507/08 is internally generated except for a frequency reference, which can be an RC circuit or an external clock source. Connect the frequency reference to the on-chip oscillator for the feedback phase shift required for oscillation. Figure 6 shows the connection for an RC circuit. Figure 7 shows the connection for an external clock source.

The internal oscillator generates a frequency in the range 60 kHz to 300 kHz depending on the frequency reference. For example, at $V_{DD} = 5$ V, an 82-kΩ resistor and a 33-pF capacitor generate a frequency of 200 kHz. The oscillation frequency is fed to the clock control circuit. It is divided by two and the resulting signal is fed to the CPU and serial interface as shown in figure 8.

Table 4 shows the operating status of the various logic blocks under the three power down-modes.

Figure 6. RC Circuit Connection

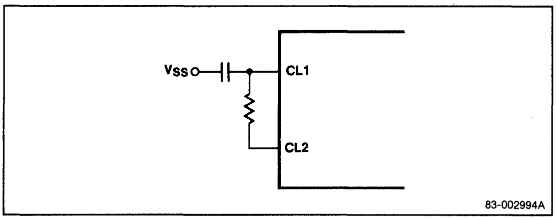


Figure 7. External Clock Source Connection

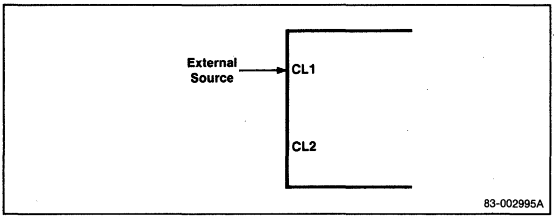


Figure 8. System Clock Circuitry

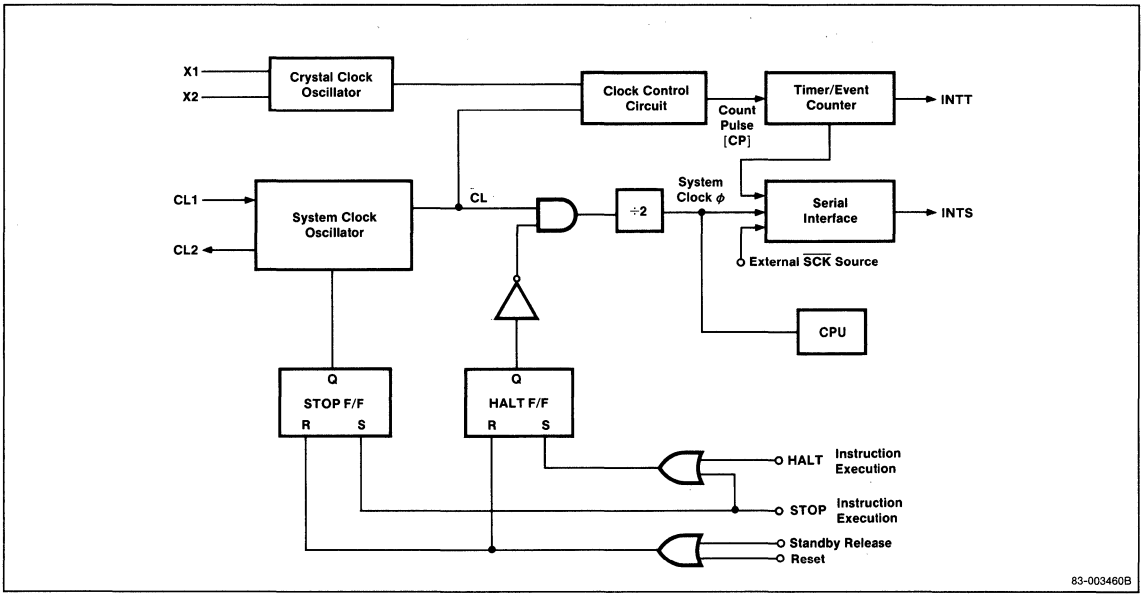


Table 4. Power-Down Operating Status

	Power-Down Mode		
Logic Block	HALT	STOP	Data Retention Mode
System clock	(Note 1)	Disabled	Disabled
X2	Normal	Normal	Disabled
CPU	Disabled	Disabled	Disabled
RAM	Data retained	Data retained	Data retained
Internal registers	Data retained	Data retained	Data retained
Timer/event counter	Normal	(Note 3)	Disabled
Serial interface	(Note 2)	(Note 2)	Disabled
INT0	Normal	Normal	Disabled
INT1	Normal	Disabled	Disabled
RESET	Normal	Normal	(Note 4)

Note:

(1) Supplied to timer/event counter but not to CPU or serial interface.

(2) Can function normally if the serial MSR is set to get the $\overline{\text{SCK}}$ signal externally or from the TOUT signal.

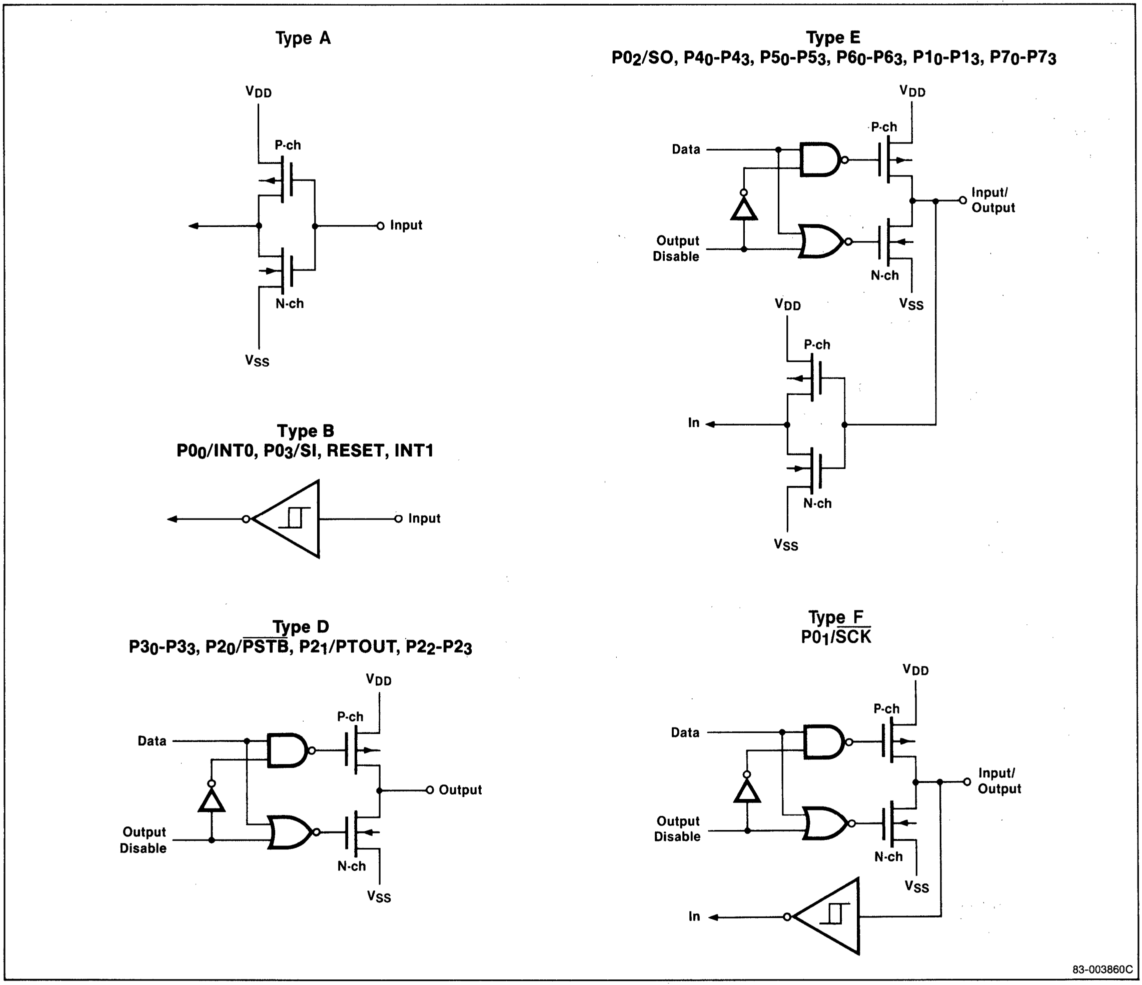
(3) Can function normally if the clock MSR is set to use X1 as the source for the count pulse.

(4) To enter the data retention mode, raise RESET while V_{DD} is lowered. To end the data retention mode, raise RESET when V_{DD} is raised, then lower it. INTT, INT0, INTS or RESET releases the STOP mode. RESET or any interrupt releases the HALT mode.

I/O Port Interfaces

Figure 9 shows the internal circuit configurations at the I/O ports.

Figure 9. ***Interface at Input/Output Ports***



Type A

Type B
$P0_0$/INT0, $P0_3$/SI, RESET, INT1

Type D
$P3_0$-$P3_3$, $P2_0$/$\overline{\text{PSTB}}$, $P2_1$/PTOUT, $P2_2$-$P2_3$

Type E
$P0_2$/SO, $P4_0$-$P4_3$, $P5_0$-$P5_3$, $P6_0$-$P6_3$, $P1_0$-$P1_3$, $P7_0$-$P7_3$

Type F
$P0_1$/$\overline{\text{SCK}}$

Absolute Maximum Ratings

$T_A = 25°C$

Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C
Power supply voltage, V_{DD}	−0.3 to +7.0 V
All input and output voltages	−0.3 to V_{DD} + 0.3 V
Output current high, I_{OH}	
One pin	−17 mA
All pins, total	−30 mA
Output current low, I_{OL}	
One pin	17 mA
Ports 1, 2, 3, 7	25 mA
Ports 4, 5, 6	25 mA

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

$T_A = 25°C$, $V_{DD} = 0$ V

		Limits			Test
Parameter	**Symbol**	**Typ**	**Max**	**Unit**	**Conditions**
Input capacitance	C_I		15	pF	f = 1 MHz; unmeasured pins returned to V_{SS}
Output capacitance	C_O		15	pF	
I/O capacitance	C_{IO}		15	pF	

DC Characteristics 1

For V_{DD} = 2.5 to 3.3 V (7507, 7508 only)

$T_A = -10$ to $+70°C$

		Limits				Test
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
Input voltage, high	V_{IH1}	0.8 V_{DD}		V_{DD}	V	Except CL1, X1
	V_{IH2}	$V_{DD} - 0.3$		V_{DD}	V	CL1, X1
	V_{IHDR}	0.9 V_{DDDR}		$V_{DDDR} + 0.2$	V	RESET, data retention mode
Input voltage, low	V_{IL1}	0		0.2 V_{DD}	V	Except CL1, X1
	V_{IL2}	0		0.3	V	CL1, X1
Output voltage, high	V_{OH}	$V_{DD} - 0.5$			V	$I_{OH} = -80$ μA
Output voltage, low	V_{OL}			0.5	V	$I_{OL} = 350$ μA
Input leakage current, high	I_{LIH1}			3	μA	Except CL1, X1; $V_I = V_{DD}$
	V_{LIH2}			10	μA	CL1, X1
Input leakage current, low	I_{LIL1}			−3	μA	Except CL1, X1; $V_I = 0$ V
	V_{LIL2}			−10	μA	CL1, X1
Output leakage current, high	I_{LOH}			3	μA	$V_O = V_{DD}$
Output leakage current, low	I_{LOL}			−3	μA	$V_O = 0$ V
Supply voltage	V_{DDDR}	2.0			V	Data retention mode
Supply current	I_{DD1}		50	250	μA	Normal operation, $V_{DD} = 3$ V ±10%; R = 240 kΩ ±2%, C = 33 pF ±5%
			35	230	μA	Normal operation, $V_{DD} = 2.5$ V; R = 240 kΩ ±2%, C = 33 pF ±5%
	I_{DD2}		0.3	10	μA	Stop mode, X1 = 0 V; $V_{DD} = 3$ V ±10%
			0.2	10	μA	Stop mode, X1 = 0 V; $V_{DD} = 2.5$ V
	I_{DDDR}		0.2	10	μA	Data retention mode, $V_{DDDR} = 2.0$ V

DC Characteristics 2

For V_{DD} = 2.7 to 6.0 V (75CGO8E, 5 V ±10%)

T_A = −10 to +70°C

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Except CL1, X1
	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	CL1, X1
	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	RESET, data retention mode, 7507/08 only
Input voltage, low	V_{IL1}	0		0.3 V_{DD}	V	Except CL1, X1
	V_{IL2}	0		0.5	V	CL1, X1
Output voltage, high	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA; V_{DD} = 4.5 to 6.0 V, 7507/08 only
		V_{DD} − 0.5			V	I_{OH} = −100 μA, 7507/08 only
	V_{OH1}	V_{DD} − 1.0			V	I_{OH} = −1.0 mA, 75CG08E only
	V_{OH2}	V_{DD} − 0.75			V	I_{OH} = −5.0 mA, 75CG08E only
Output voltage, low	V_{OL}			0.4	V	I_{OL} = 1.6 mA; V_{DD} = 4.5 to 6.0 V, 7507/08 only
				0.5	V	I_{OL} = 400 μA, 7507/08 only
				0.4	V	I_{OL} = −1.6 mA, 75CG08E only
Input current, high	I_{IH}			300	μA	75CG08E only, V_I = V_{DD}, MSEL
Input current, low	I_{IL}			−200	μA	75CG08E only, V_I = 0 V, I_0-I_7
Input leakage current, high	I_{LIH1}			3	μA	Except CL1, X1; V_I = V_{DD}
	I_{LIH2}			10	μA	CL1, X1
Input leakage current, low	I_{LIL1}			−3	μA	Except CL1, X1; V_I = 0 V
	I_{LIL2}			−10	μA	CL1, X1
Output leakage current, high	I_{LOH}			3	μA	V_O = V_{DD}
Output leakage current, low	I_{LOL}			−3	μA	V_O = 0 V
Supply voltage	V_{DDDR}	2.0			V	Data retention mode, 7507/08 only
Supply current	I_{DD1}		300	900	μA	Normal operation, V_{DD} = 5 V ±10%; R = 82 kΩ ±2%, C = 33 pF ±5%
			70	300	μA	Normal operation, V_{DD} = 3 V ±10%; R = 160 kΩ ±2%, C = 33 pF ±5%, 7507/08 only
	I_{DD2}		1.0	20	μA	Stop mode, X1 = 0 V; V_{DD} = 5 V ±10%, 7507/08 only
			0.3	10	μA	Stop mode, X1 = 0 V; V_{DD} = 3 V ±10%, 7507/08 only
			2	20	μA	Stop mode, X1 = 0 V; V_{DD} = 5 V ±10%, 75CG08E only
	I_{DDDR}		0.2	10	μA	Data retention mode V_{DDDR} = 2.0 V, 7507/08 only

AC Characteristics 1

For V_{DD} = 2.7 to 6.0 V (75CG08, 5 V ±10%)

T_A = −10 to +70°C

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
System clock frequency	f_{CC}	150	200	240	kHz	V_{DD} = 5.0 V ±10%; R = 82 kΩ ±2% (Note 1)
		75	100	120	kHz	V_{DD} = 3.0 ±10%; R = 160 kΩ ±2% (Note 1), 7507/08 only
		75		135	kHz	V_{DD} = 3.0 ±10%; R = 160 kΩ ±2% (Note 1), 7507/08 only
	f_C	10		410	kHz	CL1, external clock, 50% duty; V_{DD} = 4.5 to 6.0 V, 7507/08 only, V_{DD} = 5 V ±5%, 75CG08E only
		10		125	kHz	CL1, external clock, 50% duty; V_{DD} = 2.7 V, 7507/08 only
		10		300	kHz	CL1, external clock, 50% duty; 75CG08E only
System clock rise and fall times	t_{CR}, t_{CF}			0.2	μs	CL1, external clock
System clock pulse width	t_{CH}, t_{CL}	1.2		50	μs	CL1, external clock; V_{DD} = 4.5 to 6.0 V, 7507/08 only
		4.0		50	μs	CL1, external clock; V_{DD} = 2.7 V, 7507/08 only
		1.5		50	μs	CL1, external clock, 75CG08E only
		1.2		50	μs	CL1, external clock; V_{DD} = 5 V ±5%, 75CG08E only
Counter clock frequency	f_{XX}	25	32	50	kHz	X1, X2, crystal oscillator
	f_X	0		410	kHz	X1, external pulse input; 50% duty; V_{DD} = 4.5 to 6.0 V, 7507/08 only
		0		125	kHz	X1, external pulse input, 50% duty; V_{DD} = 2.7 V, 7507/08 only
		0		300	kHz	X1, external pulse input; 50% duty, 75CG08 only
		0		410	kHz	X1, external pulse input; 50% duty; V_{DD} = 5 V ±5%, 75CG08E only
Counter clock rise and fall times	t_{XR}, t_{XF}			0.2	μs	X1, external pulse input
Counter clock pulse width	t_{XH}, t_{XL}	1.2			μs	X1, external pulse input; V_{DD} = 4.5 to 6.0 V, 7507/08 only
		4.0			μs	X1, external pulse input; V_{DD} = 2.7 V, 7507/08 only
		1.5			μs	X1, external pulse input, 75CG08E only
		1.2			μs	X1, external pulse input; V_{DD} = 5 V ±5%, 75CG08E only
$\overline{\text{SCK}}$ cycle time	t_{KCY}	3.0			μs	$\overline{\text{SCK}}$ as input; V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±5%, 75CG08E only
		8.0			μs	$\overline{\text{SCK}}$ as input, 7507/08 only
		4.9			μs	$\overline{\text{SCK}}$ as output; V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±5%, 75CG08E only
		16.0			μs	$\overline{\text{SCK}}$ as output, 7507/08 only
		4.0			μs	$\overline{\text{SCK}}$ as input, 75CG08E only
		6.7			μs	$\overline{\text{SCK}}$ as output, 75CG08E only
$\overline{\text{SCK}}$ pulse width	t_{KH}, t_{KL}	1.3			μs	$\overline{\text{SCK}}$ as input; V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±5%, 75CG08E only
		4.0			μs	$\overline{\text{SCK}}$ as input

AC Characteristics 1 (cont)

For V_{DD} = 2.7 to 6.0 V (75CG08, 5 V ±10%)

T_A = −10 to +70°C

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
$\overline{\text{SCK}}$ pulse width	t_{KH}, t_{KL}	2.2			µs	$\overline{\text{SCK}}$ as output, V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±10%, 75CG08 only
		8.0			µs	$\overline{\text{SCK}}$ as output, 7507/08 only
		1.8			µs	$\overline{\text{SCK}}$ input, 75CG08E only
		3.0			µs	$\overline{\text{SCK}}$ as output, 75CG08E only
SI setup time to $\overline{\text{SCK}}$ ↑	t_{SIK}	300			ns	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{KSI}	450			ns	
SO delay time after $\overline{\text{SCK}}$ ↓	t_{KSO}			850	ns	V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±10%, 75CG08E only
				1200	ns	7507/08 only
Port 1 output setup time to $\overline{\text{PSTB}}$ ↑	t_{PST}	(Note 2)			µs	V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±10%, 75CG08E only
		(Note 3)			µs	7507/08 only
Port 1 output setup time to $\overline{\text{PSTB}}$ ↑	t_{STP}	100			ns	V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±10%, 75CG08E only
		100			ns	7507/08 only
$\overline{\text{PSTB}}$ pulse width	t_{STL}	(Note 2)			µs	V_{DD} = 4.5 to 6.0 V, 7507/08 only V_{DD} = 5 V ±10%, 75CG08E only
		(Note 3)			µs	7507/08 only
INT0 pulse width	t_{I0H}, t_{I0L}	10			µs	
INT1 pulse width	t_{I1H}, t_{I1L}	2/f_{CC} or 2/f_C			µs	
RESET pulse width	t_{RSH}, t_{RSL}	10			µs	
RESET setup time	t_{SRS}	0			ns	7507/08 only
RESET hold time	t_{HRS}	0			ns	7007/08 only

Note:

(1) RC network at CL1 and CL2; C = 33 pF ±5%, $|\Delta C/°C| \leq$ 60 ppm.

(2) $(10^3) \div 2(f_{CC}$ or f_C in kHz$) - 0.8$ µs.

(3) $(10^3) \div 2(f_{CC}$ or f_C in kHz$) - 2.0$ µs.

AC Characteristics 2

For V_{DD} = 2.5 to 3.3 V (7507, 7508 only)

T_A = −10 to +70°C

Parameter	Symbol	Limits			Unit	Test Conditions
		Min	Typ	Max		
System clock frequency	f_{CC}	50		80	kHz	R = 240 kΩ ±2% (Note 1)
		50	64	77	kHz	V_{DD} = 2.5 V; R = 240 kΩ ±2% (Note 1)
	f_C	10		80	kHz	CL1, external clock, 50% duty
System clock rise and fall time	t_{CR}, t_{CF}			0.2	µs	CL1, external clock
System clock pulse width	t_{CH}, t_{CL}	6.25		50	µs	CL1, external clock
Counter clock frequency	f_{XX}	25	32	50	kHz	X1, X2, crystal oscillator
	f_X	0		80	kHz	X1, external pulse input, 50% duty
Counter clock rise and fall time	t_{XR}, t_{XF}			0.2	µs	X1, external pulse input
Counter clock pulse width	t_{XH}, t_{XL}	6.25			µs	X1, external pulse input
$\overline{\text{SCK}}$ cycle time	t_{KCY}	12.5			µs	$\overline{\text{SCK}}$ as input
		25.0			µs	$\overline{\text{SCK}}$ as output
$\overline{\text{SCK}}$ pulse width	t_{KH}, t_{KL}	6.25			µs	$\overline{\text{SCK}}$ as input
		11.5			µs	$\overline{\text{SCK}}$ as output
SI setup time to $\overline{\text{SCK}}$ ↑	t_{SIK}	1			µs	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{KSI}	1			µs	
SO delay time after $\overline{\text{SCK}}$ ↓	t_{KSO}			2	µs	
Port 1 output setup time to $\overline{\text{PSTB}}$ ↑	t_{PST}	(Note 2)			µs	
Port 1 output hold time after $\overline{\text{PSTB}}$ ↑	t_{STP}	100			ns	
$\overline{\text{PSTB}}$ pulse width	t_{STL}	(Note 2)			µs	
INT0 pulse width	t_{I0H}, t_{I0L}	30			µs	
INT1 pulse width	t_{I1H}, t_{I1L}	(Note 3)			µs	
RESET pulse width	t_{RSH}, t_{RSL}	30			µs	
RESET setup time	t_{SRS}	0			ns	
RESET hold time	t_{HRS}	0			ns	

Notes:

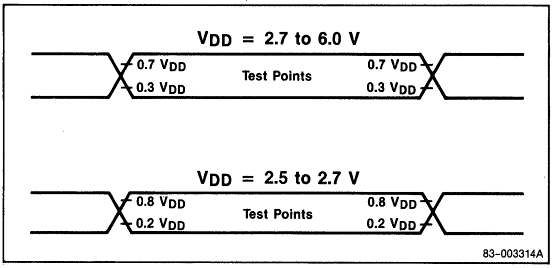
(1) RC network at CL1 and CL2; C = 33 pF ±5%, $|\Delta C/°C| \leq 60$ ppm.

(2) $10^3 \div 2$ (f_{CC} or f_C in kHz) − 2.0.

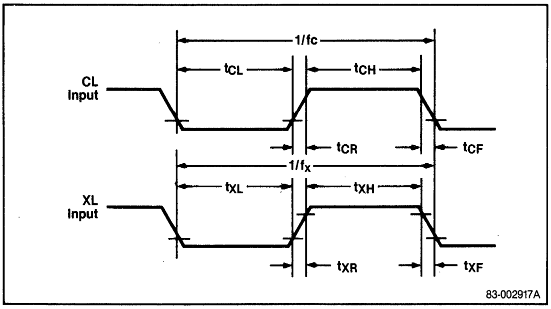
(3) $10^3 \div 2$ (f_{CC} or f_C in kHz).

Timing Waveforms

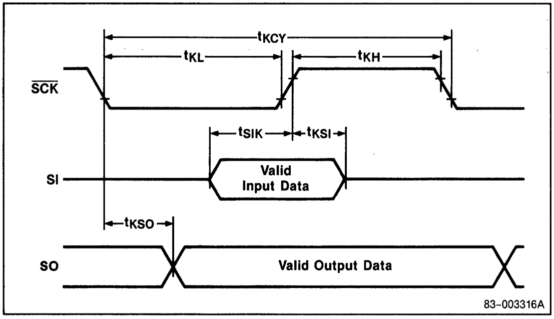
Timing Measurement Points



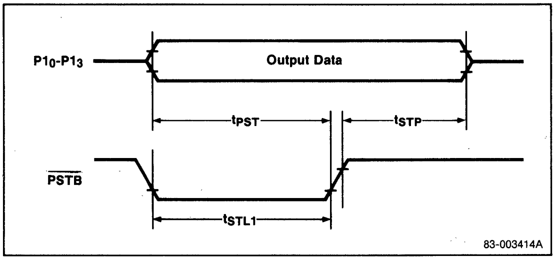
Clocks



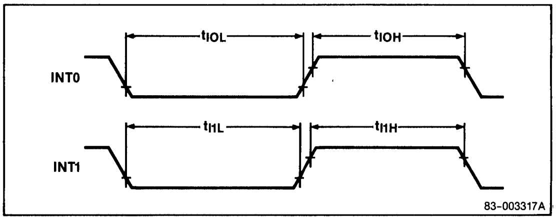
Serial Interface



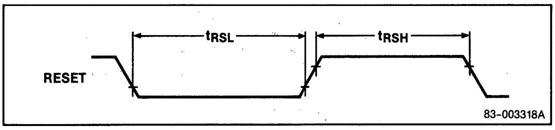
Output Strobe



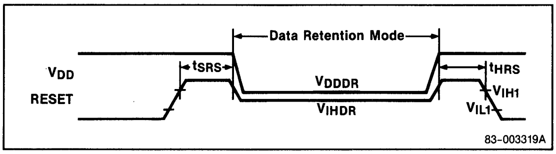
External Interrupts



RESET



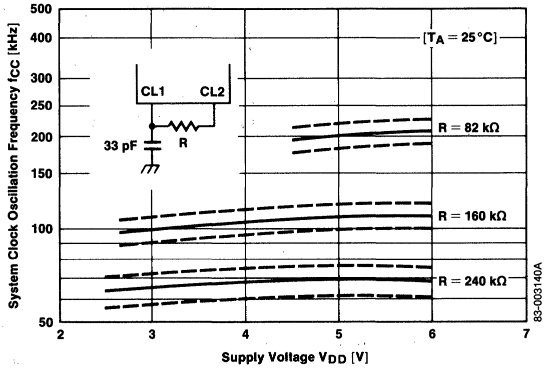
Data Retention Mode



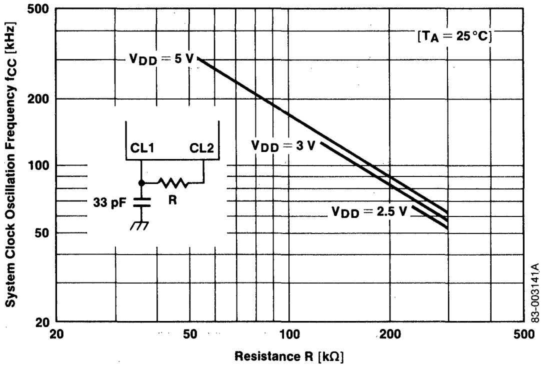
Operating Characteristics

$T_A = 25°C$

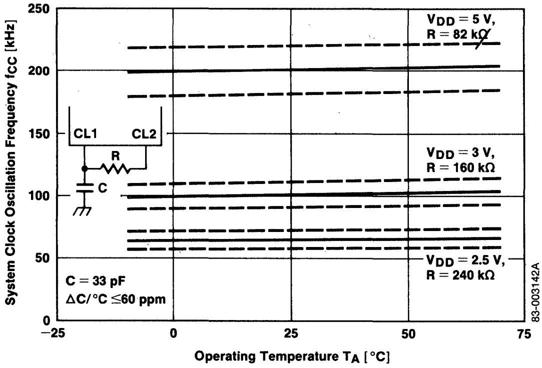
f_{CC} vs V_{DD}



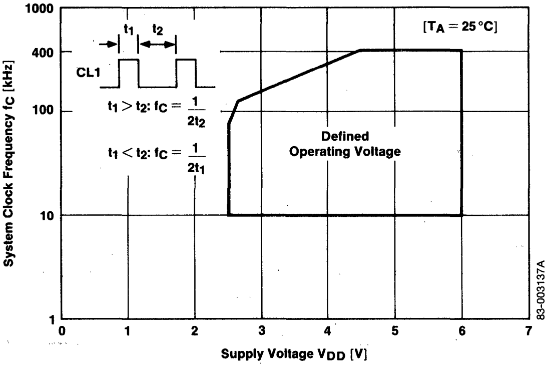
f_{CC} vs R



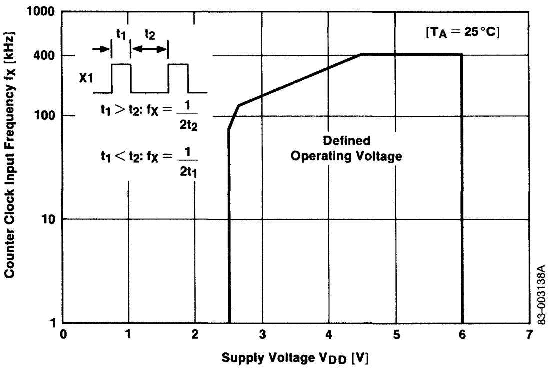
f_{CC} vs T_A



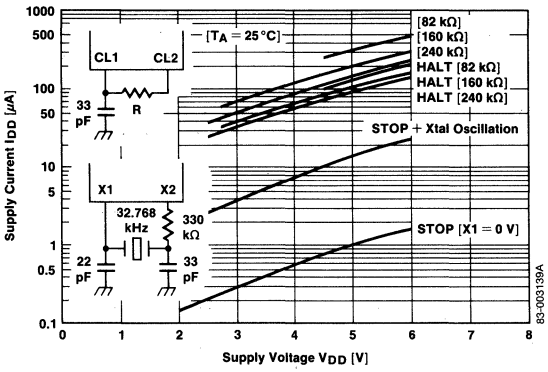
f_C vs V_{DD} [External Clock]



f_X vs V_{DD} [External Clock]



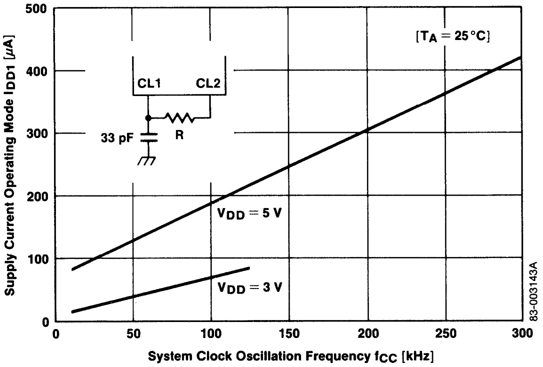
I_{DD} vs V_{DD}



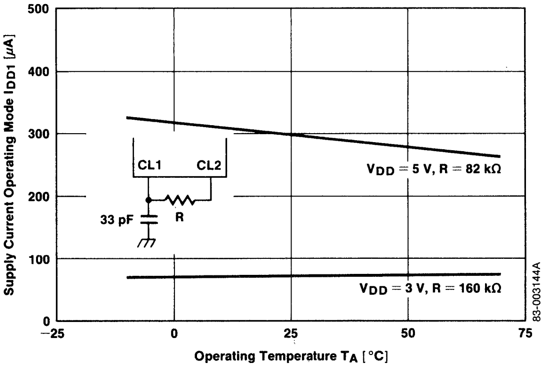
Operating Characteristics (cont)

T_A = 25°C

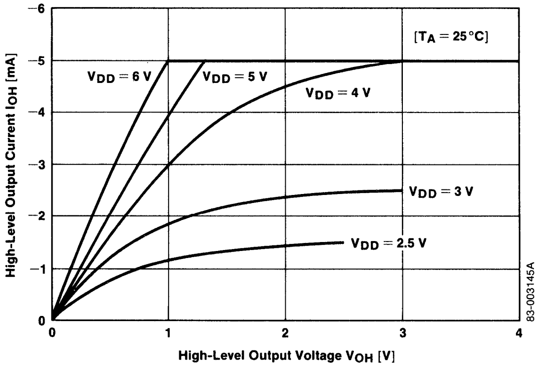
I_{DD1} vs f_{CC}



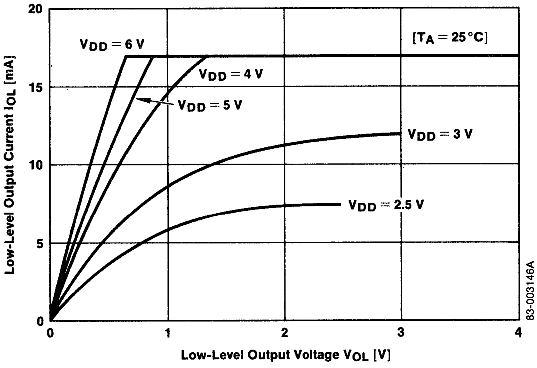
I_{DD1} vs T_A



I_{OH} vs V_{OH}



I_{OL} vs V_{OL}



NEC
NEC Electronics Inc.

μPD7507H/08H/75CG08HE 4-Bit, Single-Chip CMOS Microcomputers

Description

The μPD7507H, μPD7508H, and μPD75CG08HE are pin-compatible, high-speed (4.19 MHz), 4-bit, single-chip CMOS microcomputers with the μPD7500 series architecture. The subroutine stack is implemented in RAM for greater nesting depth and flexibility.

Thirty-two I/O lines are organized into eight 4-bit ports: input port/serial interface port 0, output ports 2 and 3, and I/O ports 1, 4, 5, 6, and 7.

The μPD7507H and μPD7508H execute 92 instructions of the μPD7500 series A instruction set with a 2.86-μs instruction cycle time.

Maximum power consumption is 3 mA at 5 V and less in the HALT and STOP low-power modes.

The 75CG08HE is a piggyback EPROM prototyping chip that is pin-compatible with 7507H and 7508H. A 2716 plugged into the top of the 75CG08HE emulates the ROM of a 7507H. A 2732 emulates the ROM of 7508H. When emulating the 7507H, the user must take care to use only the first 128 RAM locations. Although 7507H and 7508H can operate over a range of 2.7 to 6.0 V, 75CG08HE is limited to 5 V ±10%. Table 1 summarizes the differences among 7507H, 7508H, and 75CG08HE.

Features

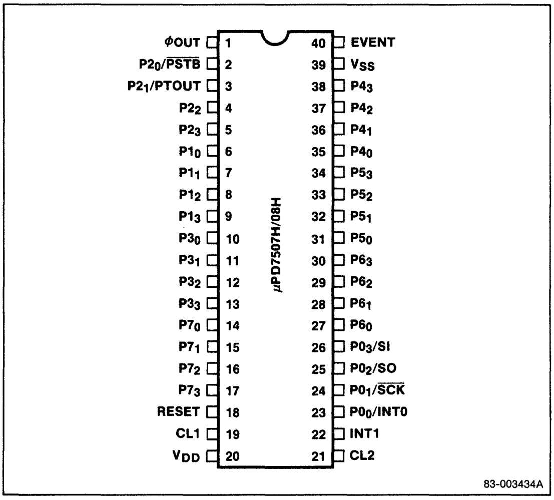
- □ Single-chip microcomputer
- □ Program ROM
  - — μPD7507H: 2048 x 8-bit
  - — μPD7508H: 4096 x 8-bit
  - — μPD75CG08HE: piggyback EPROM
- □ Data RAM
  - — μPD7507H: 128 x 4-bit
  - — μPD7508H: 224 x 4-bit
  - — μPD75CG08HE: 224 x 4-bit
- □ 8-bit timer/event counter
- □ Four 4-bit general purpose registers
- □ Four vectored, prioritized interrupts
- □ Executes 92 instructions of 7500 series A instruction set
- □ 2.86-μs instruction cycle/4.19-MHz external clock
- □ Two standby modes
- □ 32 I/O lines
- □ LED direct drive (ports 2-5; 16 lines)
- □ Low power HALT and STOP modes

Ordering Information

Part No.	Package Type	Max Frequency of Operation
μPD7507HC	40-pin plastic DIP	4.19 MHz
μPD7507HCU	40-pin plastic shrink DIP	4.19 MHz
μPD7507HGB-22	44-pin plastic QFP	4.19 MHz
μPD7508HC	40-pin plastic DIP	4.19 MHz
μPD7508HCU	40-pin plastic shrink DIP	4.19 MHz
μPD7508HGB-22	44-pin plastic QFP	4.19 MHz
μPD75CG08HE	40-pin ceramic piggyback DIP	4.19 MHz

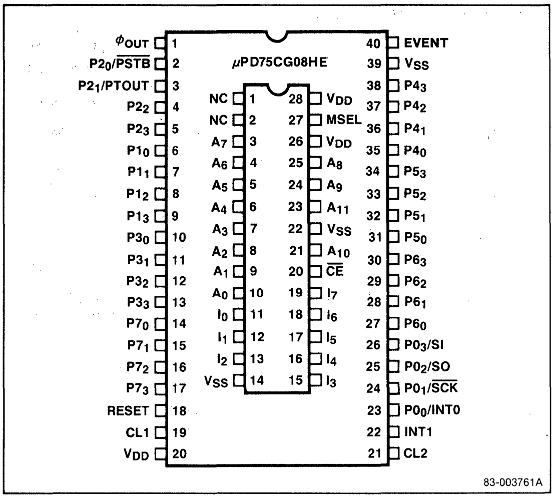
Pin Configurations

40-Pin Plastic DIP and Plastic Shrink DIP

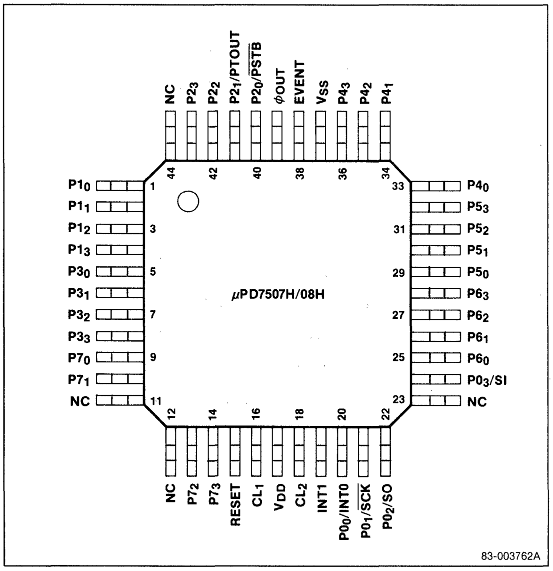


Pin Configurations (cont)

40-Pin Ceramic Piggyback DIP



44-Pin Plastic QFP



Pin Identification

40-Pin DIP, Shrink DIP, and Piggyback DIP

No.	Symbol	Function
1	ϕ_{OUT}	f_{CC}/12 square wave
2-5	P2$_0$/$\overline{PSTB}$ P2$_1$/PTOUT, P2$_2$, P2$_3$	Output port 2/output strobe pulse, timer out F/F signal
6-9	P1$_0$-P1$_3$	I/O port 1
10-13	P3$_0$-P3$_3$	Output port 3
14-17	P7$_0$-P7$_3$	I/O port 7
18	RESET	RESET input
19, 21	CL1, CL2	System clock inputs
20	V$_{DD}$	Positive power supply
22	INT1	External interrupt
23-26	P0$_0$/$\overline{INT0}$, P0$_1$/$\overline{SCK}$, P0$_2$/SO P0$_3$/SI	Input port 0/external interrupt, serial I/O interface
27-30	P6$_0$-P6$_3$	I/O port 6
31-34	P5$_0$-P5$_3$	I/O port 5
35-38	P4$_0$-P4$_3$	I/O port 4
39	V$_{SS}$	Ground
40	EVENT	External event input port

44-Pin QFP

No.	Symbol	Function
1-4	P1$_0$-P1$_3$	I/O port 1
5-8	P3$_0$-P3$_3$	Port 3 output
9, 10, 13, 14	P7$_0$-P7$_3$	I/O port 7
11-12	NC	Not connected
15	RESET	RESET input
16, 18	CL1, CL2	System clock inputs
17	V$_{DD}$	Positive power supply
19	INT1	External interrupt 1
20	P0$_0$/INT0	Port 0 input/Interrupt 0
21	P0$_1$/$\overline{SCK}$	Port 0 input/Serial clock I/O
22	P0$_2$/SO	Port 0 input/Serial output
23	NC	Not connected
24	P0$_3$/SI	Port 0 input/Serial output
25-28	P6$_0$-P6$_3$	I/O port 6
29-32	P5$_0$-P5$_3$	I/O port 5
33-36	P4$_0$-P4$_3$	I/O port 4

Pin Identification (cont)

44-Pin QFP (cont)

No.	Symbol	Function
37	V_{SS}	Ground
38	EVENT	External event input
39	ϕ_{OUT}	f_{CC}/12 square wave
40	P2$_0$/$\overline{\text{PSTB}}$	Port 2 output/Output strobe pulse
41	P2$_1$/PTOUT	Port 2 output/Timer out F/F signal
42, 43	P2$_2$, P2$_3$	Port 2 output
44	NC	Not connected

28-Pin EPROM Socket on Piggyback DIP

No.	Symbol	Function
1, 2	NC	Not connected
3-10	A_7-A_0	Address bits 7-0
11-13	I_0-I_2	Data bits 0-2
14, 22	V_{SS}	Ground
15-19	I_3-I_7	Data bits 3-7
20	$\overline{\text{CE}}$	Chip enable
21, 23	A10, A11	Address bits 10, 11
24, 25	A9, A8	Address bits 9, 8
26, 28	V_{DD}	Positive power supply
27	MSEL	Memory select

Pin Functions

P0$_0$/INT0, P0$_1$/$\overline{\text{SCK}}$, [Port 0/External Interrupt, Serial Interface] P0$_2$/SO, P0$_3$/SI

4-bit input port/serial I/O interface. This port can be configured as a 4-bit parallel input port or as the 8-bit serial I/O interface under control of the serial mode select register. The serial input SI, serial output SO, and the serial clock $\overline{\text{SCK}}$ (active low) used for synchronizing data transfer make up the 8-bit serial I/O interface. Line P0$_0$ is always shared with external interrupt INT0, a rising edge-triggered interrupt. If P0$_0$/INT0 is unused, it should be connected to V_{SS}. If P0$_1$/$\overline{\text{SCK}}$, P0$_2$/SO, or P0$_2$/SI are unused, connect them to V_{SS} or V_{DD}.

P1$_0$-P1$_3$ [Port 1]

4-bit input/three-state output port. Data output to port 1 is strobed in synchronization with a P2$_0$/$\overline{\text{PSTB}}$ pulse. Connect unused pins to V_{SS} or V_{DD}.

P2$_0$/$\overline{\text{PSTB}}$, P2$_1$/PTOUT, P2$_2$, P2$_3$ [Port 2]

4-bit latched three-state output port. Line P2$_0$ is shared with $\overline{\text{PSTB}}$, the port 1 output strobe pulse. Line P2$_1$ is shared with PTOUT, the timer out F/F signal. Leave unused pins open.

P3$_0$-P3$_3$ [Port 3]

4-bit latched three-state output port. Leave unused pins open.

P4$_0$-P4$_3$ [Port 4]

4-bit latched three-state output port. Can also perform 8-bit parallel I/O with port 5. In input mode, connect unused pins to V_{DD} or GND. In output mode, leave unused pins open.

P5$_3$-P5$_0$ [Port 5]

4-bit input/latched three-state output port. This port also performs 8-bit parallel I/O with port 4. In input mode, connect unused pins to V_{SS} or V_{DD}. In output mode, leave unused pins open.

P6$_3$-P6$_0$ [Port 6]

4-bit input/latched three-state output port. The port 6 mode select register configures individual lines as inputs or outputs. In input mode, connect unused pins to V_{SS} or V_{DD}. In output mode, leave unused pins open.

P7$_0$-P7$_3$ [Port 7]

4-bit input/latched three-state output port. In input mode, connect unused pins to V_{SS} or V_{DD}. In output mode, leave unused pins open.

ϕ_{OUT} [Clock Out]

Outputs a square wave with frequency f_{CC}/12.

EVENT [External Event Input]

Pulses on this line are counted by the timer/event counter and an interrupt is generated when a predetermined count is reached.

CL1, CL2 [System Clock Input]

The system clock can be generated by connecting a crystal or a ceramic resonator across CL1 and CL2 and capacitors from each side of the crystal to ground. Alternatively a clock signal can be input to CL1 and its invert to CL2. See figure 1.

RESET [Reset]

A high level input to this pin initializes the μPD7507H/08H after power up.

INT1 [Interrupt 1]

External rising edge-triggered interrupt. Connect to V_{SS} if unused.

V_{DD} [Power Supply]

Positive power supply. Apply a single voltage in the range 2.7 to 6.0 V for proper operation.

V_{SS} [Ground]

Ground.

Table 1. Features Comparison

	μPD75CG08H	μPD7507H/7508H
Program memory	2K x 8 EPROM (2716) 4K x 8 EPROM (2732)	2K x 8 masked ROM (7507H) 4K x 8 masked ROM (7508H)
Data memory	224 x 4	128 x 4 (7507H) 224 x 4 (7508H)
Data retention mode	Use more current than 7507H, 7508H	Yes
Power supply	5 V ±10%	2.7 to 6.0 V
Package types	40-pin ceramic piggyback DIP	40-pin plastic DIP 40-pin plastic shrink DIP 44-pin plastic QFP

Block Diagram

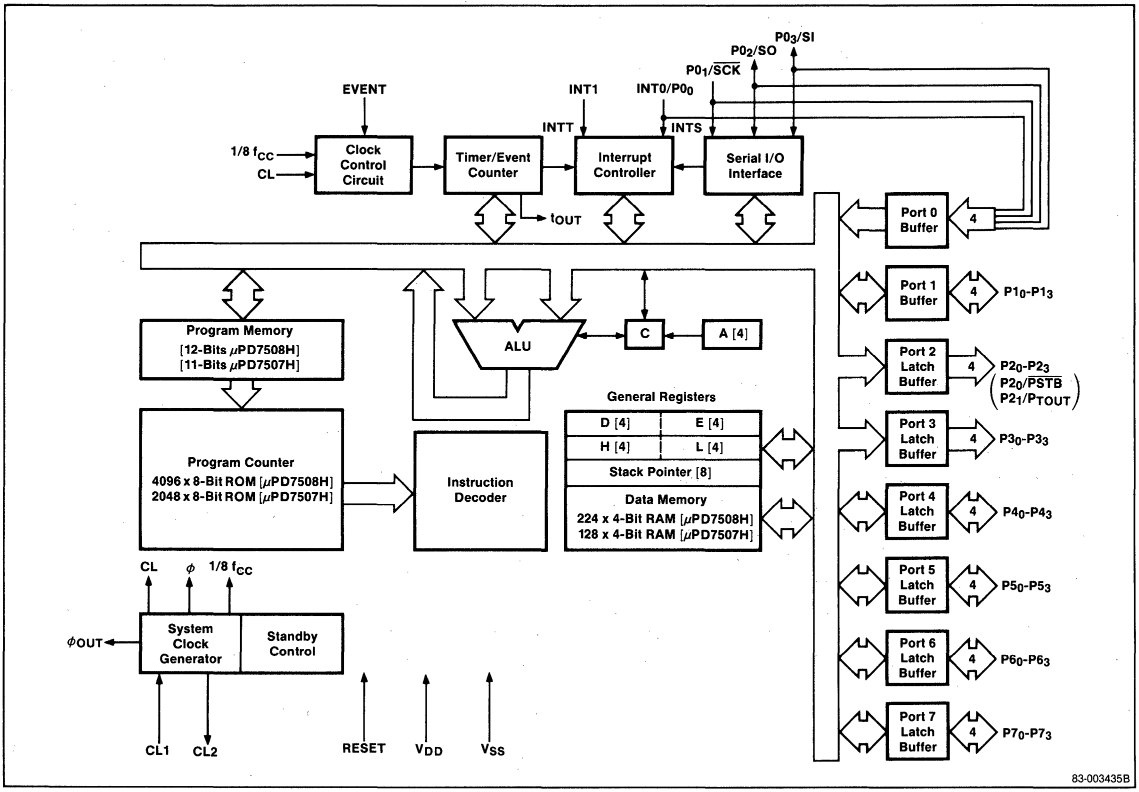
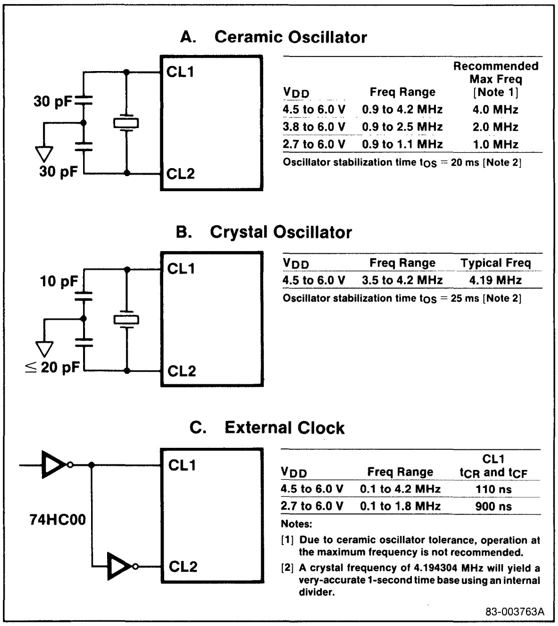


Figure 1. System Clock Options



A. Ceramic Oscillator

V_{DD}	Freq Range	Recommended Max Freq [Note 1]
4.5 to 6.0 V	0.9 to 4.2 MHz	4.0 MHz
3.8 to 6.0 V	0.9 to 2.5 MHz	2.0 MHz
2.7 to 6.0 V	0.9 to 1.1 MHz	1.0 MHz

Oscillator stabilization time t_{OS} = 20 ms [Note 2]

B. Crystal Oscillator

V_{DD}	Freq Range	Typical Freq
4.5 to 6.0 V	3.5 to 4.2 MHz	4.19 MHz

Oscillator stabilization time t_{OS} = 25 ms [Note 2]

C. External Clock

V_{DD}	Freq Range	CL1 t_{CR} and t_{CF}
4.5 to 6.0 V	0.1 to 4.2 MHz	110 ns
2.7 to 6.0 V	0.1 to 1.8 MHz	900 ns

Notes:

[1] Due to ceramic oscillator tolerance, operation at the maximum frequency is not recommended.

[2] A crystal frequency of 4.194304 MHz will yield a very-accurate 1-second time base using an internal divider.

83-003763A

Memory Map

Figure 2 shows the ROM program map of the 7507H/7508H.

Clock Control Circuit

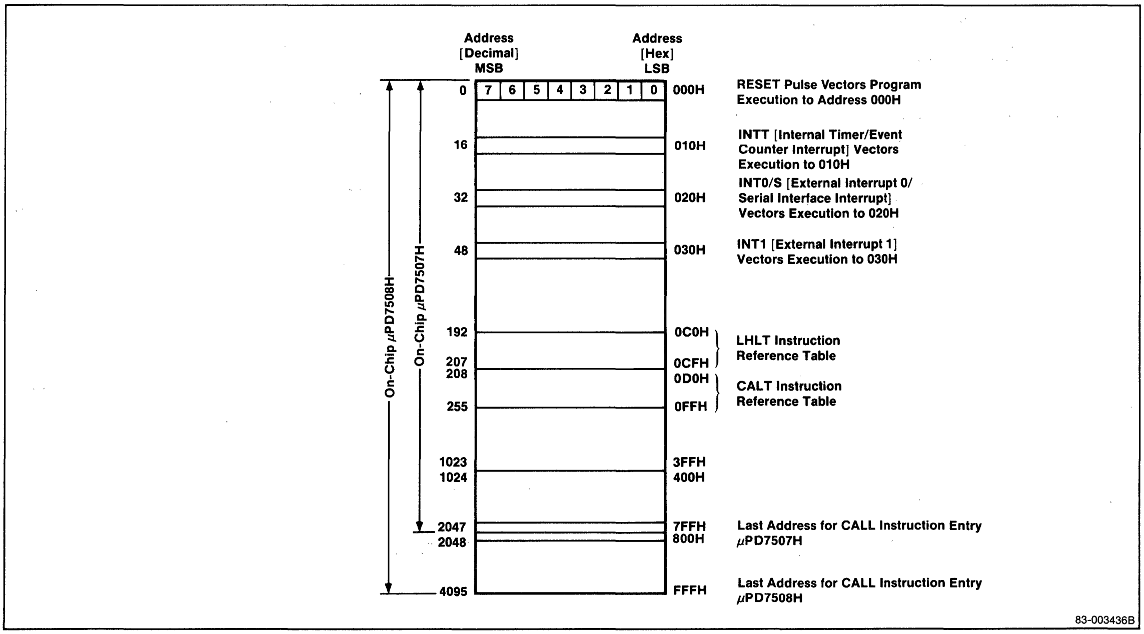
The clock control circuit consists of a 4-bit clock mode register (bits CM_0-CM_3), prescalers 1, 2, and 3, and a multiplexer. It takes the output of the system clock generator (CL) and external EVENT input. It also selects the clock source and divides the signal according to the setting in the clock mode register. It outputs the count pulse (CP) to the timer/event counter. Figure 3 shows the clock control circuit.

Table 2 lists the codes set in the clock mode register by the OP or OPL instruction to specify the count pulse frequency. Bit CM_3 controls the timer out F/F; it is disabled when the bit is 0 and output when the bit is 1.

Table 2. Selecting the Count Pulse Frequency

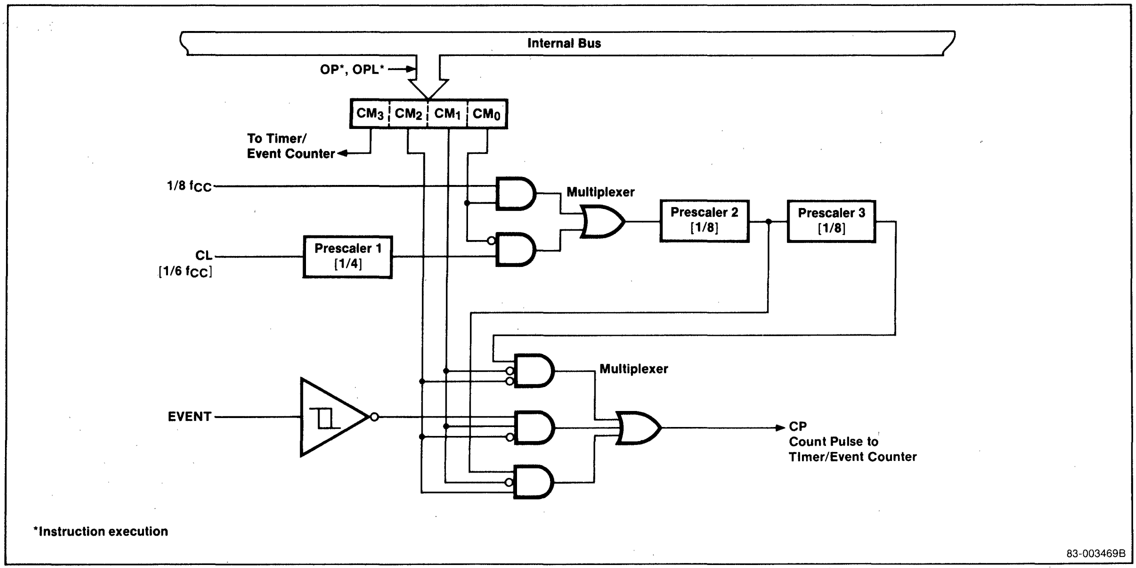
CM_2	CM_1	CM_0	Frequency Selected
0	0	0	f_{CC}/1536 (or CL/256)
0	0	1	f_{CC}/512 = (f_{CC}/8) (1/64)
0	1	0	EVENT input
0	1	1	Not used
1	0	0	f_{CC}/192 (or CL/32)
1	0	1	f_{CC}/64 = (f_{CC}/8) (1/8)
1	1	0	Not used
1	1	1	Not used

Figure 2. ROM Map


Address [Decimal] MSB
Address [Hex] LSB
7 6 5 4 3 2 1 0
0
000H
RESET Pulse Vectors Program Execution to Address 000H
16
010H
INTT [Internal Timer/Event Counter Interrupt] Vectors Execution to 010H
32
020H
INT0/S [External Interrupt 0/ Serial Interface Interrupt] Vectors Execution to 020H
48
030H
INT1 [External Interrupt 1] Vectors Execution to 030H
192
0C0H
207
0CFH
LHLT Instruction Reference Table
208
0D0H
255
0FFH
CALT Instruction Reference Table
1023
3FFH
1024
400H
2047
7FFH
2048
800H
Last Address for CALL Instruction Entry μPD7507H
4095
FFFH
Last Address for CALL Instruction Entry μPD7508H
On-Chip μPD7508H
On-Chip μPD7507H

83-003436B

Figure 3. Clock Control Circuit



*Instruction execution

Timer/Event Counter

The timer/event counter consists of an 8-bit counter, an 8-bit modulo register, an 8-bit comparator, and a timer out flip flop as shown in figure 4.

The 8-bit count register is a binary 8-bit up counter, which is incremented each time a count pulse is input. The TIMER instruction, a RESET signal, or an INTT coincidence signal clears it to 00H.

The 8-bit modulo register determines the number of counts the count register holds. The TAMMOD instruction loads the contents of the modulo register. RESET sets the modulo register to FFH.

The 8-bit comparator compares the contents of the count register and the modulo register and outputs an INTT one clock pulse after they are equal.

Serial Interface

The 8-bit serial interface allows the μPD7507H/08H to communicate with peripheral devices such as the μPD7001 A/D converter, the μPD7227 dot matrix LCD controller/driver, and other microprocessors or microcomputers.

The serial interface consists of an 8-bit shift register, a 3-bit $\overline{\text{SCK}}$ pulse counter, the SI input port, the SO output port, the $\overline{\text{SCK}}$ serial clock I/O port, and a 4-bit serial mode select register (MSR). The MSR selects serial I/O or port 0 operation.

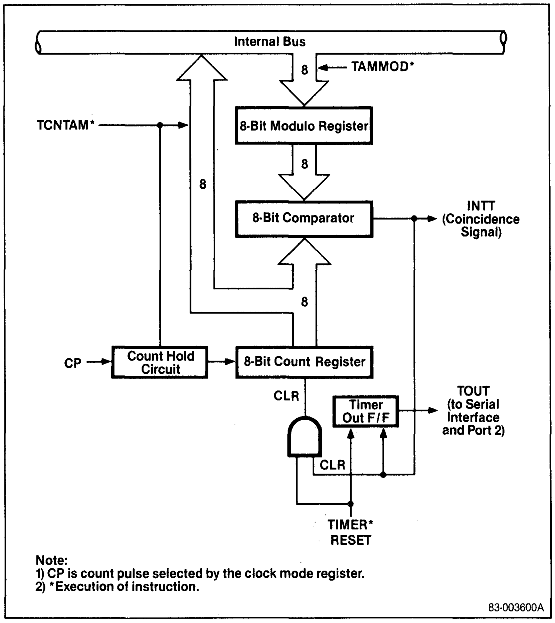
Interrupts

The μPD7507H/08H has four vectored, prioritized interrupts. Two of these interrupts, INTT and INTS, are internally generated from the timer/event counter and serial interface, respectively. INT0 and INT1 are externally generated. Table 3 is a summary of the four interrupts.

Table 3. μPD7507H/08H Interrupts

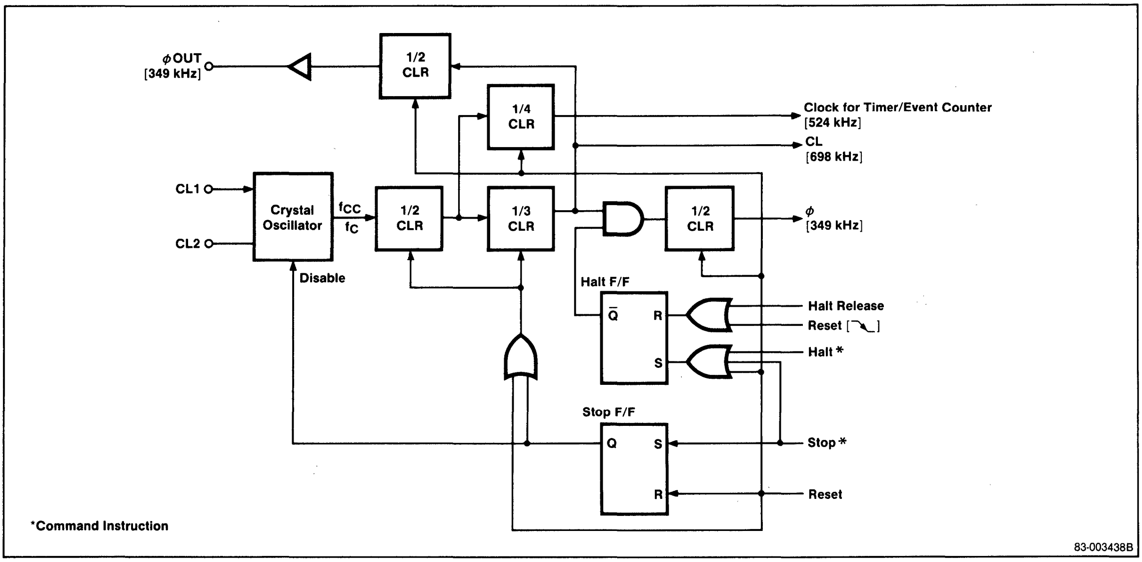
Source	Function	Location	Priority	ROM Vector Address
INTT	Coincidence in timer/event counter	Internal	1	10H
INTS	Transfer complete signal from serial interface	Internal	2	20H
INT0	INT0 pin	External	2	20H
INT1	INT1 pin	External	3	30H

Figure 4. Timer/Event Counter



Note:
1) CP is count pulse selected by the clock mode register.
2) *Execution of instruction.

Figure 5. System Clock Circuitry



*Command Instruction

System Clock and Timing Circuitry

There are four time bases available for the µPD7507H/08H. Table 4 shows these bases and the frequencies generated.

The CPU clock is used by the CPU and serial interface. The system clock is used by the timer/event counter and the INT1 signal.

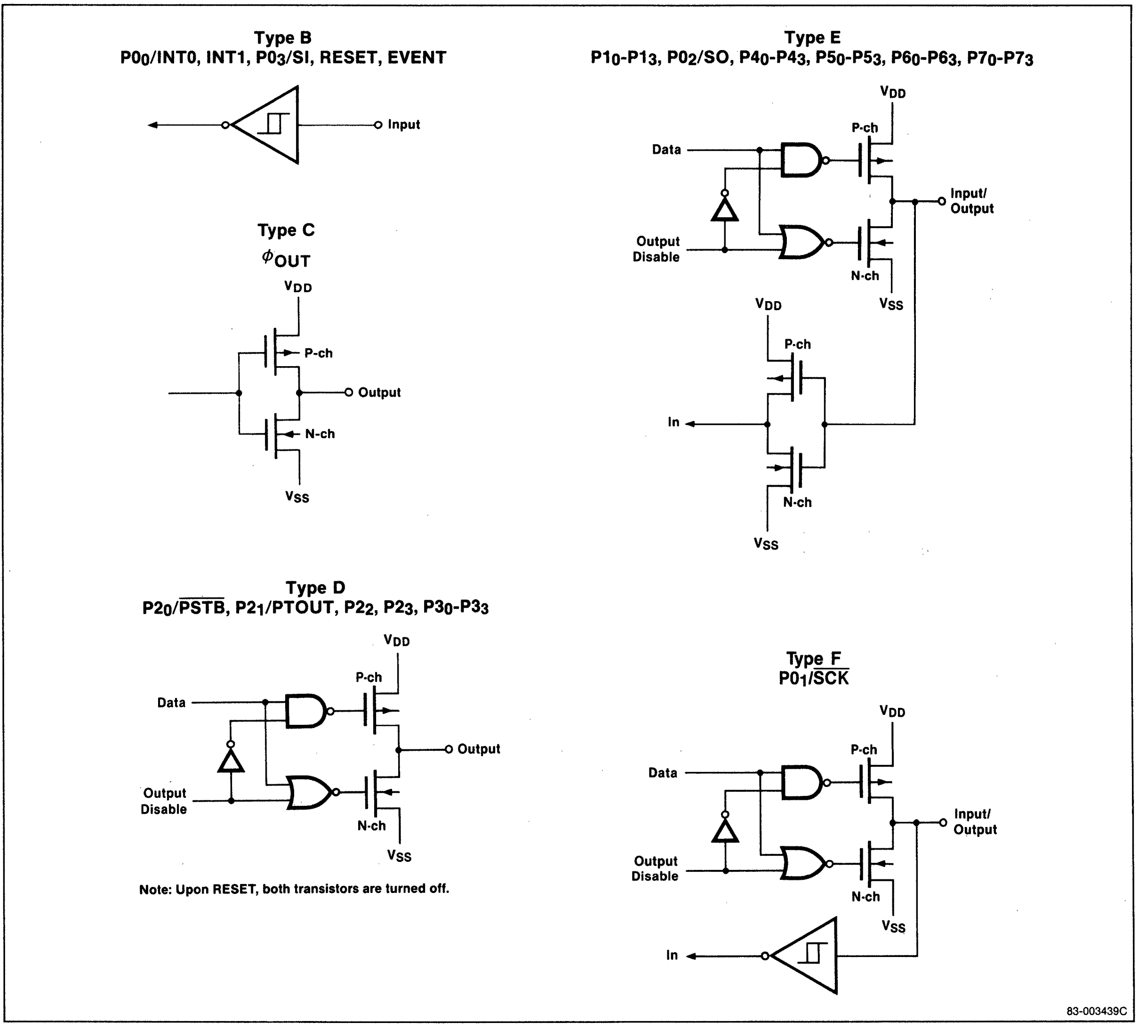
Table 4. µPD7507H/08H Time Bases

Base	Symbol	Frequency	Derivation
System clock	CL	698 kHz	$f_{CC}/6$ (4.19 MHz/6)
CPU clock	ϕ	349 kHz	$f_{CC}/12$ (4.19 MHz/12)
External clock	ϕ_{OUT}	349 kHz	$f_{CC}/12$ (4.19 MHz/12)
Timer/event counter clock	—	524 kHz	$f_{CC}/8$ (4.19 MHz/8)

I/O Port Interfaces

Figure 6 shows the internal circuit configurations at the I/O ports.

Figure 6. Interface at Input/Output Ports



Note: Upon RESET, both transistors are turned off.

Absolute Maximum Ratings

$T_A = 25°C$

Operating temperature, T_{OPT}	−10 to 70°C
Storage temperature, T_{STG}	−65 to 150°C
Power supply voltage, V_{DD}	−0.3 to +7.0 V
All input and output voltages	−-0.3 to V_{DD} + 0.3 V
Output current, high, I_{OH}	
One pin	−5 mA
All pins, total	−20 mA
Output current, low, I_{OL}	
One pin	17 mA
Ports 6, 7	20 mA
Total ports	200 mA

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

$T_A = 25°C$, $V_{DD} = 0$ V

		Limits			Test
Parameter	Symbol	Typ	Max	Unit	Conditions
Input capacitance	C_I		15	pF	f = 1 MHz; unmeasured pins returned to V_{SS}.
Output capacitance	C_O		15	pF	
I/O capacitance	C_{IO}		15	pF	

DC Characteristics

$T_A = -10$ to $+70$°C; $V_{DD} = 2.7$ to 6.0 V (5 V ±10% for 75CG08HE)

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Except CL1, CL2
	V_{IH2}	$V_{DD} - 0.5$		V_{DD}	V	CL1, CL2
	V_{IHDR}	0.9 V_{DDDR}		$V_{DDDR} + 0.2$	V	RESET, data retention mode, μPD7507H/08H only
Input voltage, low	V_{IL1}	0		0.3 V_{DD}	V	Except CL1, CL2
	V_{IL2}	0		0.5	V	CL1, CL2
Output voltage, high	V_{OH}	V_{DD} −1.0			V	$I_{OH} = -1.0$ mA; $V_{DD} = 4.5$ to 6.0 V; except A_{11}/V_{PP}, for μPD75CG08HE
		$V_{DD} - 0.5$			V	$I_{OL} = -100$ μA
		$V_{DD} - 0.75$			V	A_{11}/V_{PP}; $I_{OH} = -5$ mA (μPD75CG08HE only)
Output voltage, low	V_{OL}		0.5	1.5	V	$I_{OL} = 12$ mA; $V_{DD} = 4.5$ to 6.0 V; Ports 2-5
				0.4	V	$I_{OL} = 1.6$ mA; $V_{DD} = 4.5$ to 6.0 V; Ports 6-7
				0.5	V	$I_{OL} = 400$ μA
High level input current (MSEL)	I_{IH}	$V_{IN} = V_{DD}$		300	μA	μPD75CG08HE only
Low level input current (I_0-I_7)	I_{IL}	$V_{IN} = 0$ V		−200	μA	μPD75CG08HE only
Input leakage current, high	I_{LIH1}			3	μA	Except CL1, CL2; $V_I = V_{DD}$
	I_{LIH2}			20	μA	CL1, CL2; $V_I = V_{DD}$
Input leakage current, low	I_{LIL1}			−3	μA	Except CL1, CL2; $V_I = 0$ V
	I_{LIL2}			−20	μA	CL1, CL2; $V_I = 0$ V
Output leakage current, high	I_{LOH}			3	μA	$V_O = V_{DD}$
Output leakage current, low	I_{LOL}			−3	μA	$V_O = 0$ V
Supply voltage	V_{DDDR}	2.0		6.0	V	Data retention mode, μPD7507H/08H only
Supply current	I_{DD1}		900 (1) 1000 (2)	3000 (1) 3000 (2)	μA μA	Normal operation, $V_{DD} = 4.5$ to 6.0 V; $f_{CC} = 4.19$ MHz
			150 (2)	700 (2)	μA	Normal operation, $V_{DD} = 2.7$ to 3.3 V; $f_{CC} = 1$ MHz, μPD7507H/08H only
	I_{DD2}		350 (1) 500 (2)	800 (1) 1100 (2)	μA μA	HALT mode, X1 = 0 V; $V_{DD} = 4.5$ to 6.0 V; $f_{CC} = 4.19$ MHz
			70 (2)	180 (2)	μA	HALT mode, X1 = 0 V; $V_{DD} = 2.7$ to 3.3 V; $f_{CC} = 1$ MHz, μPD7507H/08H only
	I_{DD3}		0.1	10	μA	STOP mode, μPD7507H/08H only
			0.5	50	μA	STOP mode, μPD75CG08HE only

Notes:

(1) Crystal oscillation; C1 = C2 = 10 pF.

(2) Ceramic oscillation; C1 = C2 = 30 pF.

AC Characteristics

$T_A = -10$ to $+70°C$; $V_{DD} = 2.7$ to 6.0 V (5 V ±10% for 75CG08HE)

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
System cycle time	t_{CY}	2.86		120	μs	$V_{DD} = 4.5$ to 6.0 V
		11		120	μs	
EVENT input frequency	f_E	0		700	kHz	$V_{DD} = 4.5$ to 6.0 V
		0		150	kHz	
EVENT input high	t_{EH}	0.7			μs	$V_{DD} = 4.5$ to 6.0 V
EVENT input low	t_{EL}	3.3			μs	
$\overline{\text{SCK}}$ cycle time	t_{KCY}	2.5			μs	$\overline{\text{SCK}}$ as input; $V_{DD} = 4.5$ to 6.0 V
		10			μs	$\overline{\text{SCK}}$ as input
		2.86			μs	$\overline{\text{SCK}}$ as output; $V_{DD} = 4.5$ to 6.0 V
		11			μs	$\overline{\text{SCK}}$ as output
$\overline{\text{SCK}}$ pulse width	t_{KH}, t_{KL}	1.1			μs	$\overline{\text{SCK}}$ as input; $V_{DD} = 4.5$ to 6.0 V
		4.5			μs	$\overline{\text{SCK}}$ as input
		1.3			μs	$\overline{\text{SCK}}$ as output; $V_{DD} = 4.5$ to 6.0 V
		5.0			μs	$\overline{\text{SCK}}$ as output
SI setup time to $\overline{\text{SCK}}$ ↑	t_{SIK}	300			ns	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{KSI}	450			ns	
SO delay time after $\overline{\text{SCK}}$ ↓	t_{KSO}			850	ns	$V_{DD} = 4.5$ to 6.0 V
				1200	ns	
Port 1 output setup time to $\overline{\text{PSTB}}$ ↑	t_{PST}	(Note 1)			ns	$V_{DD} = 4.5$ to 6.0 V
		(Note 2)			ns	
Port 1 output hold time after $\overline{\text{PSTB}}$ ↑	t_{STP}	80			ns	
$\overline{\text{PSTB}}$ pulse width	t_{SWL}	(Note 1)			ns	$V_{DD} = 4.5$ to 6.0 V
		(Note 2)			ns	
INT0 pulse width	t_{I0H}, t_{I0L}	10			μs	
INT1 pulse width	t_{I1WH}, t_{I1WL}	1 (Note 3)			t_{CY}	
RESET pulse width	t_{RSH}, t_{RSL}	10			μs	
RESET setup time	t_{SRS}	0			ns	
Clock stabilization time	t_{OS}	25			ms	$V_{DD} = 4.5$ V

Notes:

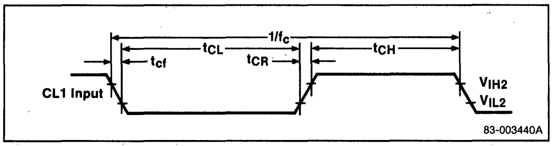
(1) $(3 \div f_{CC}$ or $f_C) - 350$.

(2) $(3 \div f_{CC}$ or $f_C) - 1000$.

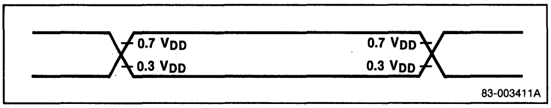
(3) $t_{CY} = 12 \div f_{CC}$ or f_C.

Timing Waveforms

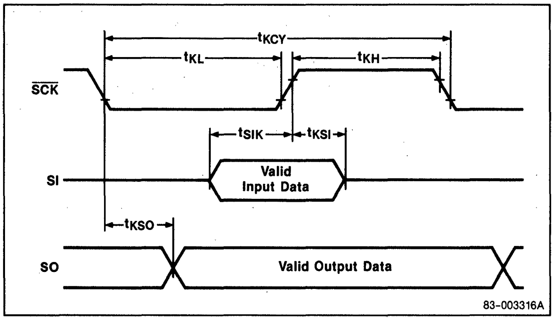
Clocks



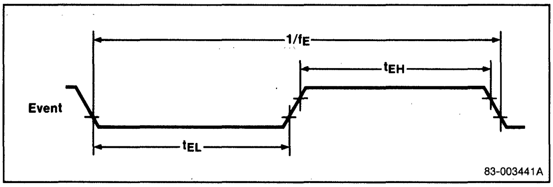
Timing Measurement Points



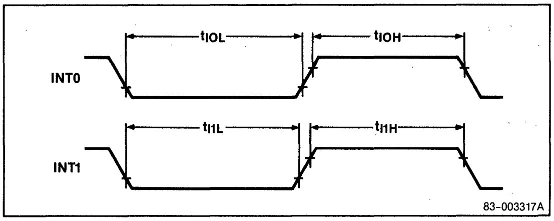
Serial Interface



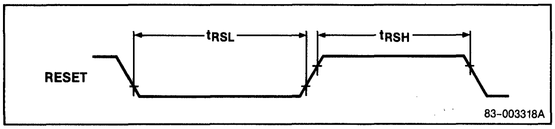
EVENT Input



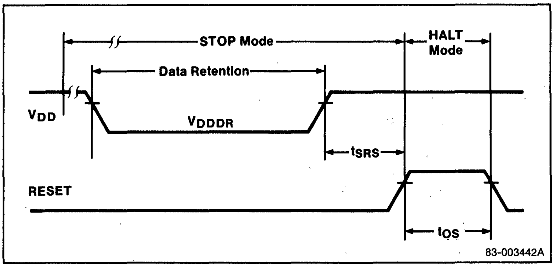
External Interrupts



Reset



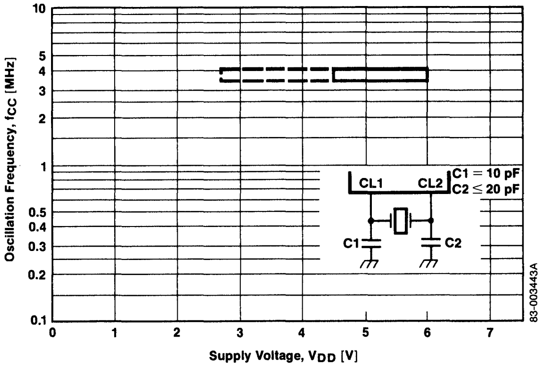
STOP Mode



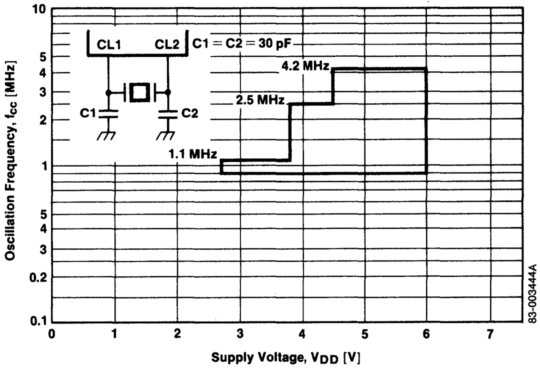
Operating Characteristics (cont)

$T_A = 25\,°C$

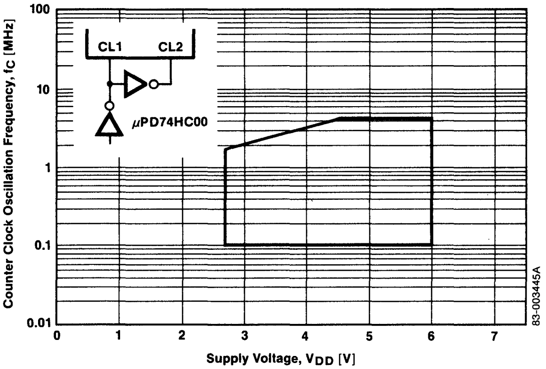
Oscillator Frequency vs Supply Voltage



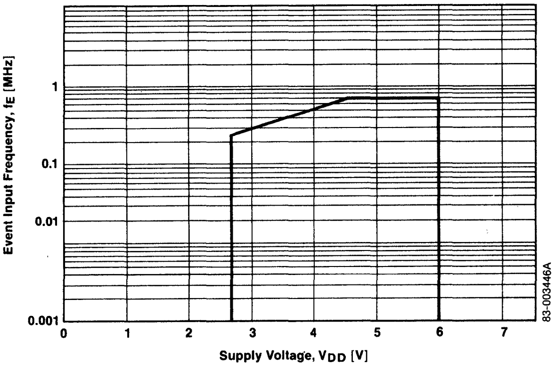
Oscillator Frequency vs Supply Voltage



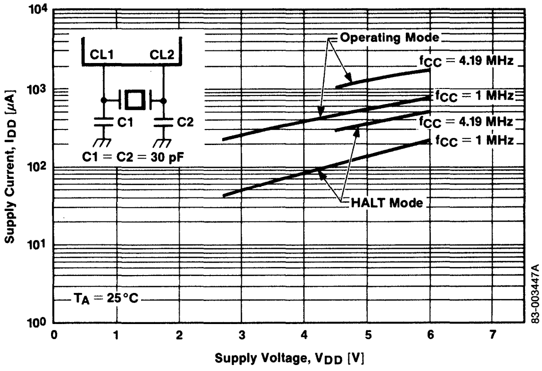
Clock Frequency vs Supply Voltage



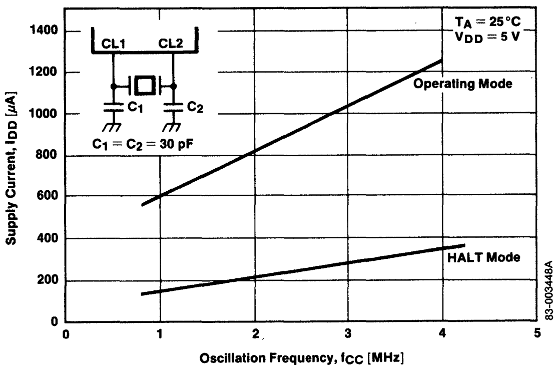
Event Frequency vs Supply Voltage



Supply Current vs Supply Voltage

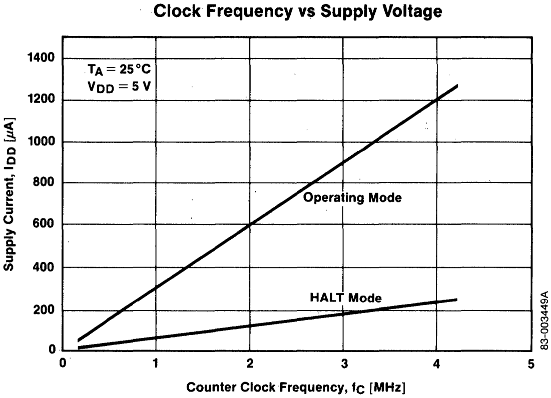


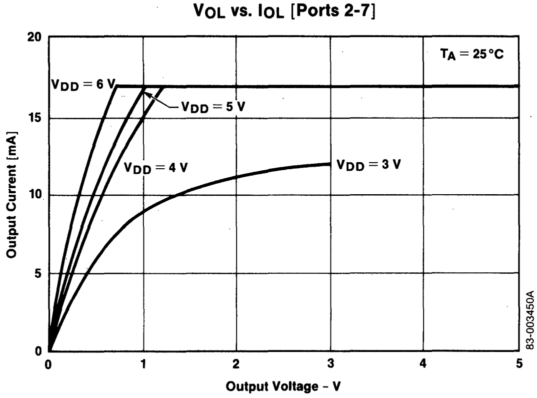
Oscillator Frequency vs Supply Voltage

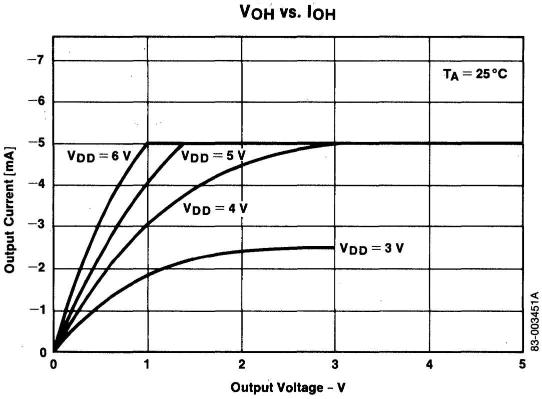


Operating Characteristics (cont)

$T_A = 25°C$







NEC
NEC Electronics Inc.

μPD7527A/28A/75CG28E 4-Bit, Single-Chip CMOS Microcomputers With FIP® Driver

Description

The μPD7527A, μPD7528A, and μPD75CG28E are 4-bit, single-chip CMOS microcomputers with the μPD7500 architecture and FIP direct-drive capability.

The μPD7527A contains a 2048 x 8-bit ROM and a 128 x 4-bit RAM. The μPD7528A contains a 4096 x 8-bit ROM and 160 x 4-bit RAM.

The μPD7527A/28A contains two 4-bit general purpose registers located outside RAM. The subroutine stack is implemented in RAM for greater depth and flexibility. The μPD7527A/28A typically executes 67 instructions with a 5 μs instruction cycle time.

The μPD7527A/28A has one external and two internal edge-triggered hardware-vectored interrupts. It also contains an 8-bit timer/event counter and an 8-bit serial interface to help reduce software requirements.

Thirty-one high-voltage lines are organized into the 3-bit output port 2, the 4-bit output ports 3, 8, and 9, and the 4-bit I/O ports 4, 5, 10, and 11.

The low power consumption CMOS process allows the use of a power supply between 2.7 and 6.0 V. Current consumption is less than 3.0 mA maximum, and can be further reduced in the halt and stop power-down modes.

The μPD75CG28E is a piggyback EPROM version of the μPD7527A/28A. Pin-compatible and function-compatible with the final, masked versions of the μPD7527A/28A, the μPD75CG28E is used for prototyping and for aiding in program development.

Features

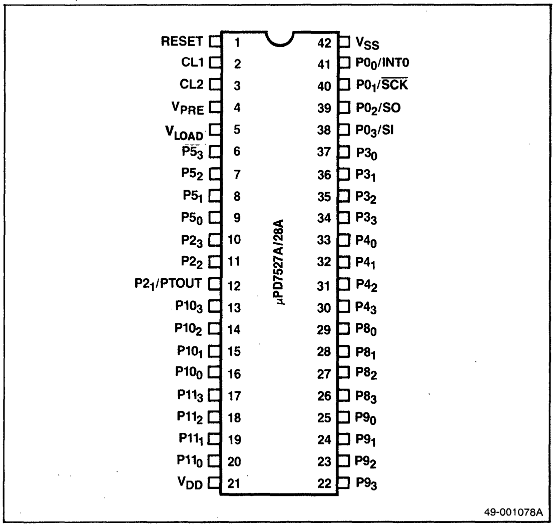
□ 67 instructions
□ Instruction cycle:
  — Internal clock: 3.3 μs/600 kHz, 5 V
  — External clock: 3.3 μs/600 kHz, 5 V
□ Upwardly compatible with the μPD7500 series product family
□ 4,096 × 8-bit ROM (μPD7528A/75CG28E)
  2,048 × 8-bit ROM (μPD7527A)
□ 160 × 4-bit RAM (μPD7528A/75CG28E)
  128 × 4-bit RAM (μPD7527A)
□ 35 I/O lines
□ 31 high-voltage output lines that can directly drive a vacuum fluorescent display (FIP)
□ Can select either a pull-down resistor or open-drain output per 31 high-voltage outputs (mask optional)
□ Vectored interrupts: one external, two internal
□ 8-bit timer/event counter
□ 8-bit serial interface
□ Standby function (HALT, STOP)
□ Data retention mode
□ Zero-cross detector on P0$_0$/INT0 input (mask optional)
□ System clock (μPD7527A/7528A/75CG28E): on-chip RC oscillator
□ CMOS technology
□ Low power consumption
□ Single power supply
  — μPD7527A/7528A: 2.7 to 6.0 V
  — μPD75CG28E: 5.0 V

Ordering Information

Part Number	Package Type	Max Frequency of Operation
μPD7527AC / 28AC	42-pin plastic DIP	610 kHz
μPD7527ACU / 28ACU	42-pin plastic shrink DIP	610 kHz
μPD75CG28E	42-pin ceramic piggyback DIP	500 kHz

Pin Configurations

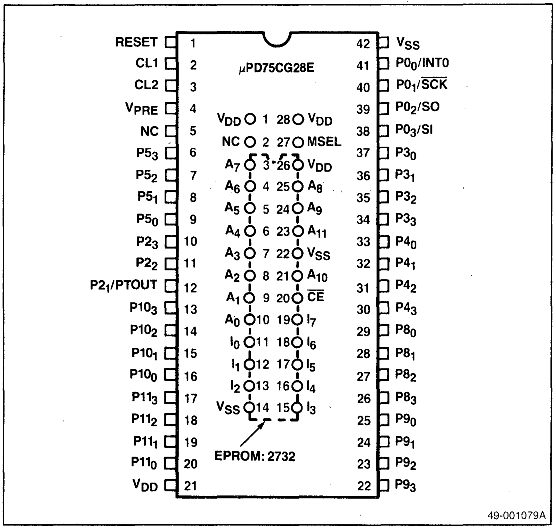
μPD7527A/28A, 42-Pin Plastic DIP or Shrink DIP



FIP is the registered trademark for NEC's fluorescent indicator panel (vacuum fluorescent display).

Pin Configurations (cont)

μPD75CG28E, 42-Pin Ceramic Piggyback DIP



Pin Identification

μPD7527A/28A and μPD75CG28E

No.	Symbol	Function
1	RESET	Reset input
2, 3	CL1, CL2	Clock pins
4	V_{PRE}	High-voltage predriver supply
5	V_{LOAD}	High-voltage option resistor supply 7527A / 28A only
6-9	$P5_0$–$P5_3$	High-voltage I / O port 5
10, 12	$P2_3$, $P2_2$ $P2_1$ / PTOUT	High-voltage output port 2, and output port from timer / event counter (PTOUT)
13–16	$P10_0$–$P10_3$	High-current, high-voltage I / O port 10
17–20	$P11_0$–$P11_3$	High-voltage, high-current I / O port 11
21	V_{DD}	Positive power supply
22–25	$P9_0$–$P9_3$	High-voltage, high-current output port 9
26–29	$P8_0$–$P8_3$	High-voltage, high-current output port 8
30–33	$P4_0$–$P4_3$	High-voltage I / O port 4
34–37	$P3_0$–$P3_3$	High-voltage output port 3
38 39 40 41	$P0_3$ / SI $P0_2$ / SO $P0_1$ / $\overline{SCK}$ $P0_0$ / INTO	4-bit input of port 0; or serial data input (SI), serial data output (SO), serial clock I / O ($\overline{SCK}$), and external interrupt input (INTO) or zero-cross detect input ($P0_0$).
42	V_{SS}	Ground

μPD75CG28E EPROM

No.	Symbol	Function
1	V_{DD}	Connection to pin 21 of μPD75CG28E
2	NC	No connection
3-10, 21, 24, 25	A_0–A_{10}	EPROM address output
11-13, 15-19	I_0–I_7	Data read input from the EPROM
14	V_{SS}	Connection to EPROM GND pin
20	$\overline{CE}$	Chip enable output
22	V_{SS}	Supplies EPROM $\overline{OE}$ signal
23	A_{11}	Program counter MSB output
26	V_{DD}	Supplies V_{CC} to the EPROM
27	MSEL	Mode select input
28	V_{DD}	Supplies high-level signal to MSEL

Note:

(1) Output drivers on ports 2-5 and 8-11 are mask-optional. Accordingly, either an open-drain output or a pull-down resistor can be selected. V_{LOAD} is suitable for an output driver with a pull-down resistor.

(2) Ports 2-5 are suitable as FIP segment signal outputs, and ports 8-11 are suitable for FIP digit signal outputs.

(3) Ports 8-11 have high-current drive capability and can drive an LED directly.

Pin Functions, μPD7527A/28A and μPD75CG28E

RESET

System reset (input).

CL1, CL2

Connection to the RC oscillator. CL1 is the external clock input.

V_{PRE}

Negative power supply for high-voltage output predrivers (for ports 2-5, 8-11).

V_{LOAD}

Negative power supply for optional load resistors (pull-down resistors) of high-voltage output drivers (for ports 2-5, 8-11). This pin is only on the μPD7527A/28A.

$P5_3$-$P5_0$

4-bit, high-voltage I/O port 5.

$P2_1$-$P2_3$

3-bit, high-voltage output port 2.

PTOUT

Output port from the timer/event counter.

$P10_3$-$P10_0$

4-bit, high-voltage, high-current I/O port 10. Capable of bit set/reset by SPBL/RPBL instructions.

$P11_3$-$P11_0$

4-bit, high-voltage, high-current I/O port 11. Capable of bit set/reset by SPBL/RPBL instructions.

V_{DD}

Positive power supply.

$P9_3$-$P9_0$

4-bit, high-voltage, high-current output port 9. Capable of bit set/reset by SPBL/RPBL instructions.

$P8_3$-$P8_0$

4-bit, high-voltage, high-current output port 8. Capable of bit set/reset by SPBL/RPBL instructions.

$P4_3$-$P4_0$

4-bit, high-voltage I/O port 4.

$P3_3$-$P3_0$

4-bit, high-voltage output port 3.

$P0_0$-$P0_3$

4-bit input port 0. $P0_0$ is also used as the zero-cross detection input.

SI

Serial data input.

SO

Serial data output.

$\overline{\text{SCK}}$

I/O serial clock.

INT0

External interrupt input.

V_{SS}

Ground.

Pin Functions, μPD75CG28E EPROM

MSEL

Changes the addressing area of the external EPROM and the on-chip RAM (with a pull-down resistor). Connecting a jumper between socket pins 27 (MSEL) and 28 (V_{DD}) selects μPD7527A mode (2-Kbyte EPROM, 128 x 4-bit RAM). Leaving MSEL open selects μPD7528A mode (4-Kbyte EPROM, 160 x 4-bit RAM).

A_0-A_{10}

Output the low-order 11 bits of the program counter (PC_0-PC_{10}). Used as EPROM address signals.

A_{11}

When MSEL is high level, A_{11} outputs high-level signals. When MSEL is open, A_{11} outputs the MSB of the PC, which is used as the most significant address signal of the 4-Kbyte EPROM 2732.

I_0-I_7

Input data read from the EPROM.

$\overline{\text{CE}}$

Outputs the chip enable signal to the EPROM.

V_{DD}

Pin 26 is electrically equivalent to the bottom V_{DD} pin and is used to supply V_{CC} to the EPROM. Pin 28 is electrically equivalent to the bottom V_{DD} pin and is used to supply the high level signal to MSEL. Pin 1 connects to pin 21 of μPD75CG28E.

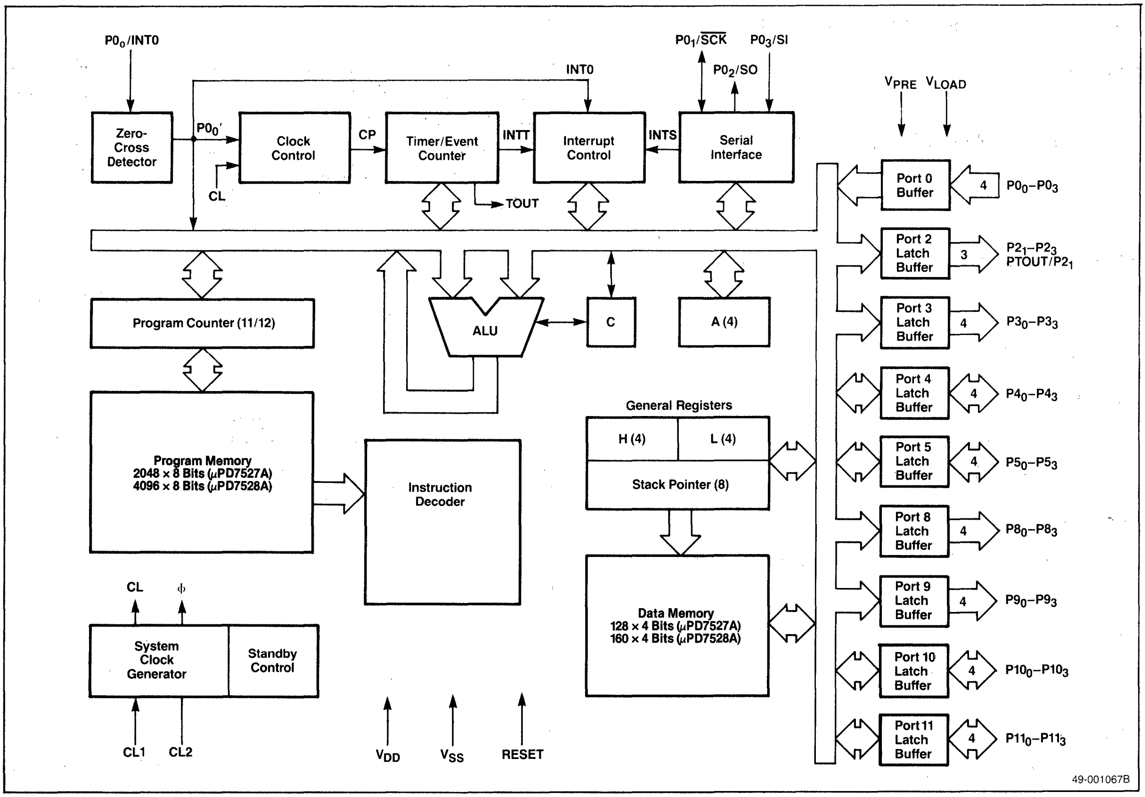
V_{SS}

Pin 14 is electrically equivalent to the bottom V_{SS} pin in voltage, and is connected to the EPROM GND pin. Pin 22 is electrically equivalent to the bottom V_{SS} pin and is used to supply the $\overline{\text{OE}}$ signal to the EPROM.

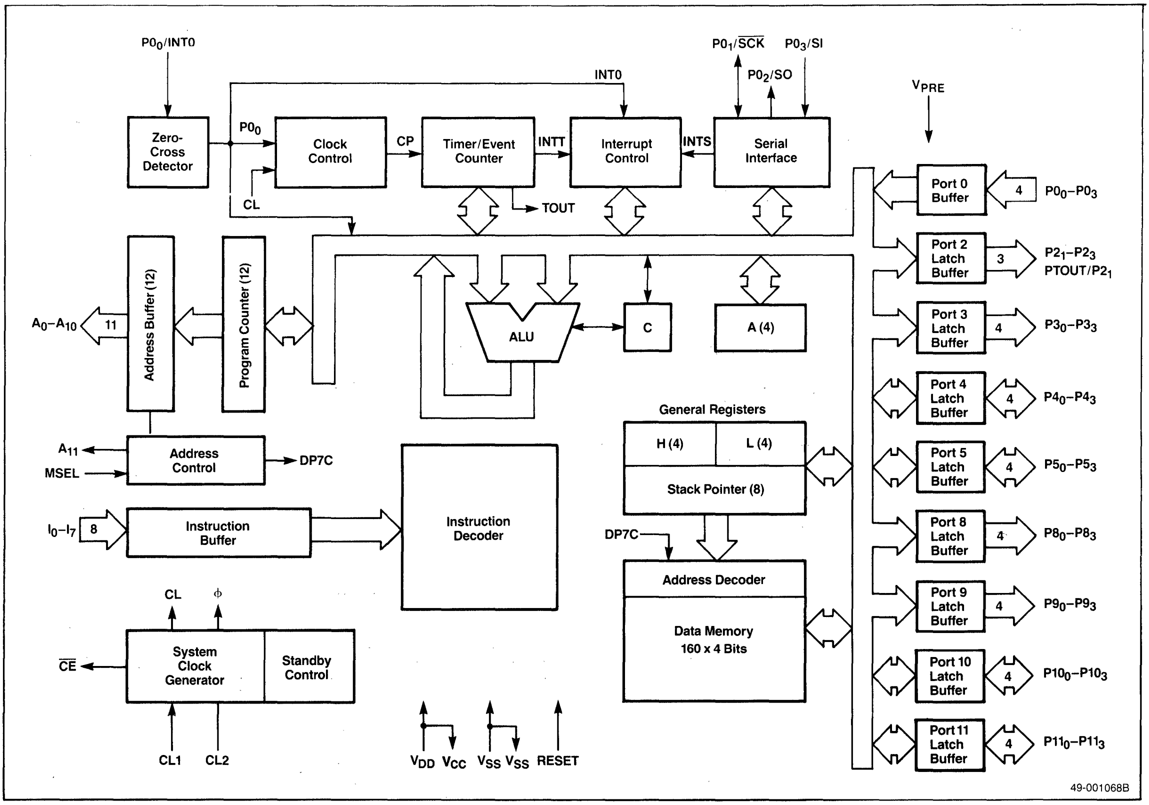
Instruction Set

Refer to the User's Manual. The instruction set appears also as subset A4 in the data sheet for the μPD7500 series of single-chip microcomputers.

Block Diagram, μPD7527A/28A



Block Diagram, µPD75CG28E



Absolute Maximum Ratings

$T_A = 25°C$

Power supply voltage, V_{DD}	−0.3 to +7 V
Power supply voltage, V_{LOAD} (µPD7527A / 28A)	V_{DD} −40 V to V_{DD} +0.3 V
Power supply voltage, V_{PRE}	V_{DD} −12 V to V_{DD} +0.3 V
Input voltage, except ports 4, 5, 10, 11, V_{IN}	−0.3 V to V_{DD} +0.3 V
Input voltage, ports 4, 5, 10, 11, V_{IN}	V_{DD} −40 V to V_{DD} +0.3 V
Output voltage, except ports 2-5, 8-11, V_O	−0.3 V to V_{DD} +0.3 V
Output voltage, ports 2-5, 8-11, V_O	V_{DD} −40 V to V_{DD} +0.3 V
Output current high, per pin: P0₁, P0₂; I_{OH}	−15 mA
Output current high, per pin: ports 2-5, 8-11; I_{OH}	−30 mA
Output current high, ports 3, 4, 8, 9 total, I_{OH}	−55 mA
Output current high, ports 2, 5, 10, 11 total, I_{OH}	−55 mA
Output current low, per pin, I_{OL}	15 mA
Output current low, all ports total, I_{OL}	15 mA
Operating temperature, T_{OPT}	−10°C to +70°C
Storage temperature, T_{STG}	−65°C to +150°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of the specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

µPD7527A/28A

$T_A = -10°C$ to $+70°C$, $V_{DD} = +2.7$ V to 6.0 V

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage, low	V_{IL1}	0		0.3 V_{DD}	V	Port 0, RESET
	V_{IL2}	0		0.5	V	CL1
	V_{IL3}	$V_{DD}-35$		0.3 V_{DD}	V	Ports 4, 5, 10, 11
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Port 0, RESET
	V_{IH2}	$V_{DD}-0.5$		V_{DD}	V	CL1
	V_{IH3}	0.7 V_{DD}		V_{DD}	V	Ports 4, 5, 10, 11; 4.5 V ≤ V_{DD} ≤ 6.0 V
		$V_{DD}-0.5$		V_{DD}	V	Ports 4, 5, 10, 11; 2.7 V ≤ V_{DD} ≤ 4.5 V
Output voltage, low	V_{OL}			0.4	V	PO_1, PO_2; 4.5 V ≤ V_{DD} ≤ 6.0 V; I_{OL} = 1.6 mA
				0.5	V	PO_1, PO_2; I_{OL} = 400 µA
Output voltage, high	V_{OH}	$V_{DD}-2.0$			V	Ports 2–5, I_{OH} = −4 mA (Note 1)
		$V_{DD}-2.0$			V	Ports 8–11, I_{OH} = −10 mA (Note 1)
		$V_{DD}-2.0$			V	Ports 2–5, I_{OH} = −2 mA (Note 2)
		$V_{DD}-2.0$			V	Ports 8–11, I_{OH} = −5 mA (Note 2)
		$V_{DD}-1.0$			V	PO_1, PO_2; I_{OH} = −1 mA (Note 3)
		$V_{DD}-0.5$			V	PO_1, PO_2; I_{OH} = −100 µA
Input leakage current, low	I_{LIL1}			−3	µA	V_{IN} = 0 V; PO_0–PO_3 (Note 4)
	I_{LIL2}			−40	µA	V_{IN} = 0 V; PO_0 (Note 5)
	I_{LIL3}			−10	µA	V_{IN} = 0 V; CL1
	I_{LIL4}			−10	µA	$V_{IN} = V_{DD}-35$ V; ports 4, 5, 10, 11
Input leakage current, high	I_{LIH1}			3	µA	$V_{IN} = V_{DD}$; PO_0–PO_3 (Note 4)
	I_{LIH2}			40	µA	$V_{IN} = V_{DD}$; PO_0 (Note 5)
	I_{LIH3}			10	µA	$V_{IN} = V_{DD}$; CL1
	I_{LIH4}			80	µA	$V_{IN} = V_{DD}$; ports 4, 5, 10, 11

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Output leakage current, low	I_{LOL1}			−3	µA	V_O = 0 V; PO_1, PO_2
	I_{LOL2}			−10	µA	$V_O = V_{DD}-35$ V; ports 2–5, 8–11
Output leakage current, high	I_{LOH1}			3	µA	$V_O = V_{DD}$; except ports 4, 5, 10, 11
	I_{LOH2}			80	µA	$V_O = V_{DD}$; ports 4, 5, 10, 11
Supply current, normal operation	I_{DD1}		1.0	3.0	mA	V_{DD} = 5 V ± 10%, R = 39 kΩ
			0.4	1.0	mA	V_{DD} = 3 V, R = 82 kΩ
Supply current, HALT mode (Note 6)	I_{DD2}		200	600	µA	V_{DD} = 5 V ± 10%, R = 39 kΩ (Note 4)
			60	200	µA	V_{DD} = 3 V, R = 82 kΩ (Note 4)
			210	640	µA	V_{DD} = 5 V ± 10%, R = 39 kΩ (Note 5)
			67	230	µA	V_{DD} = 3 V, R = 82 kΩ (Note 5)
Supply current, STOP mode (Note 6)	I_{DD3}		0.1	10	µA	V_{DD} = 3 V (Note 4)
			10	40	µA	V_{DD} = 5 V ± 10% (Note 5)
			7	30	µA	V_{DD} = 3 V (Note 5)
On-chip pull-down resistance	R_L	80	140	220	kΩ	$V_{DD}-V_{LOAD}$ = 35 V

Note:

(1) $V_{PRE} = V_{DD} - 9$ V ± 1 V. The circuit in figure 5 is recommended.

(2) V_{PRE} = 0 V. V_{DD} = 4.5 V to 6.0 V.

(3) V_{DD} = 4.5 V to 6.0 V.

(4) Without zero-cross detector.

(5) With zero-cross detector.

(6) Ports 4, 5, 10, 11 are low level output or low level input.

DC Characteristics (cont)

µPD75CG28E

T_A = −10°C to +70°C, V_{DD} = +5 V ±10%

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage, low	V_{IL1}	0		0.3 V_{DD}	V	Port 0, RESET
	V_{IL2}	0		0.5	V	CL1
	V_{IL3}	V_{DD} − 35		0.3 V_{DD}	V	Ports 4, 5, 10, 11
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Port 0, RESET
	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	CL1
	V_{IH3}	0.7 V_{DD}		V_{DD}	V	Ports 4, 5, 10, 11
Output voltage, low	V_{OL}			0.4	V	PO_1, PO_2; I_{OL} = 1.6 mA
				0.5	V	PO_1, PO_2; I_{OL} = 400 µA
Output voltage, high	V_{OH}	V_{DD} − 2.0			V	Ports 2–5, I_{OH} = −4 mA (1)
		V_{DD} − 2.0			V	Ports 8–11, I_{OH} = −10 mA(1)
		V_{DD} − 2.0			V	Ports 2–5, I_{OH} = −2 mA(2)
		V_{DD} − 2.0			V	Ports 8–11, I_{OH} = −5 mA(2)
		V_{DD} − 1.0			V	PO_1, PO_2; I_{OH} = −1 mA
Input current, low (I_0–I_7)	I_{IL}			−200	µA	V_{IN} = 0 V
Input current, high (MSEL)	I_{IH}			300	µA	V_{IN} = V_{DD}

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input leakage current, low	I_{LIL1}			−3	µA	V_{IN} = 0 V; $P0_0$–$P0_3$
	I_{LIL2}			−40	µA	V_{IN} = 0 V; $P0_0$
	I_{LIL3}			−10	µA	V_{IN} = 0 V; CL1
	I_{LIL4}			−10	µA	V_{IN} = V_{DD} − 35 V; ports 4, 5, 10, 11
Input leakage current, high	I_{LIH1}			3	µA	V_{IN} = V_{DD}; $P0_0$–$P0_3$
	I_{LIH2}			40	µA	V_{IN} = V_{DD}; $P0_0$
	I_{LIH3}			10	µA	V_{IN} = V_{DD}; CL1
	I_{LIH4}			80	µA	V_{IN} = V_{DD}; ports 4, 5, 10, 11
Output leakage current, low	I_{LOL1}			−3	µA	V_O = 0 V; PO_1, PO_2
	I_{LOL2}			−10	µA	V_O = V_{DD} − 35 V; ports 2–5, 8–11
Output leakage current, high	I_{LOH1}			3	µA	V_O = V_{DD}; except ports 4, 5, 10, 11
	I_{LOH2}			80	µA	V_O = V_{DD}; ports 4, 5, 10, 11
Supply current, normal operation	I_{DD1}		1.0	3.0	mA	R = 39 kΩ
Supply current, HALT mode(3)	I_{DD2}		210	630	µA	R = 39 kΩ
Supply current, STOP mode(3)	I_{DD3}		10	50	µA	

Note:

(1) V_{PRE} = V_{DD} − 9 V +1 V. The circuit in figure 6 is recommended.

(2) V_{PRE} = 0 V

(3) Ports 4, 5, 10, 11 are output off or low input.

Figure 1. Recommended Circuit, µPD7527A/7528A

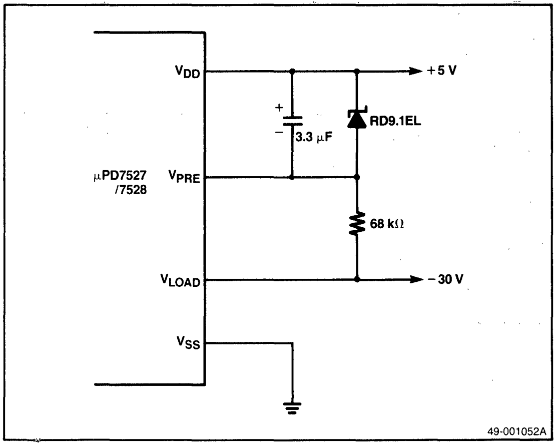
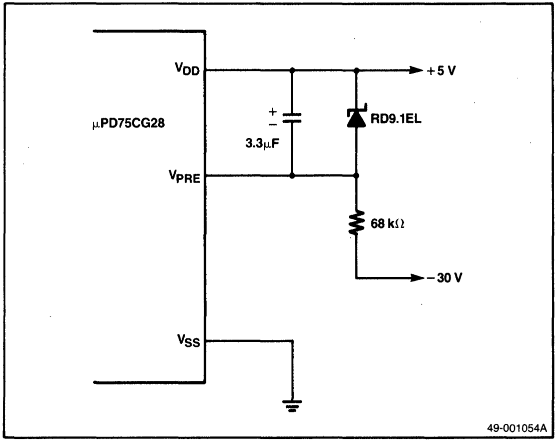


Figure 2. Recommended Circuit, µPD75CG28E

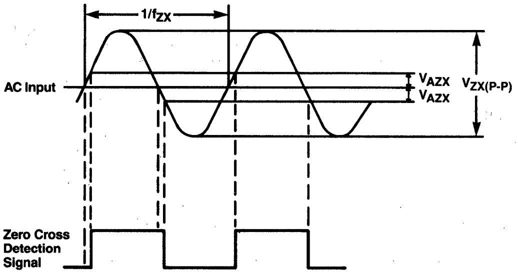


Zero-Cross Detection Characteristics

μPD7527A/28A: T_A = −10°C to +70°C, V_{DD} = 4.5 V to 6.0 V
μPD75CG28E: T_A = −10°C to +70°C, V_{DD} = +5 V ±10%

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Zero-cross detection input voltage	V_{ZX}(P-P)	1		3	V_{P-P}	AC coupled, C = 0.1 μF
Zero-cross accuracy	V_{AZX}			±100	mV	50 Hz to 60 Hz sine wave
Zero-cross detection input frequency	f_{ZX}	45		1000	Hz	

Zero-Cross Detection Waveform



Note: In the above waveforms, both 0-to-1 and 1-to-0 transitions of the zero-cross detection signal delay from the low-to-high and high-to-low transitions of the AC input signal, respectively. However, it is possible that the zero-cross detection leads low-to-high and/or high-to-low transition(s) of the AC input signal.

49-001055A

Capacitance

T_A = 25°C, V_{DD} = 0 V, f = 1.0 MHz, Unmeasured pins returned to GND

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input capacitance	C_I			15	pF	$P0_0$, $P0_3$
Output capacitance	C_O			15	pF	Port 2
				35	pF	Ports 3, 8, 9
I/O capacitance	C_{IO}			15	pF	$P0_1$, $P0_2$
				35	pF	Ports 4, 5, 10, 11

AC Characteristics

μPD7527A/28A

T_A = −10°C to +70°C, V_{DD} = +2.7 V to 6.0 V

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Cycle time (Note 1)	t_{CY}	3.3		200	μs	V_{DD} = 4.5 V to 6.0 V
		6.9		200	μs	
$P0_0$ event input frequency	f_{P0}	0		610	kHz	V_{DD} = 4.5 V to 6.0 V
		0		290	kHz	
$P0_0$ input rise time	t_{POR}			0.1	μs	
$P0_0$ input fall time	t_{POF}			0.1	μs	
$P0_0$ input pulse width, low	t_{POL}	1.63			μs	
$P0_0$ input pulse width, high	t_{POH}	0.72			μs	V_{DD} = 4.5 V to 6.0 V
$\overline{SCK}$ cycle time	t_{KCY}	3.0			μs	Input; V_{DD} = 4.5 V to 6.0 V
		3.3			μs	Output; V_{DD} = 4.5 V to 6.0 V
		6.9			μs	Input
		8.0			μs	Output
$\overline{SCK}$ pulse width, low	t_{KL}	3.35			μs	Input
		3.9			μs	Output
$\overline{SCK}$ pulse width, high	t_{KH}	1.4			μs	Input; V_{DD} = 4.5 V to 6.0 V
		1.55			μs	Output; V_{DD} = 4.5 V to 6.0 V
SI set-up time (to rising-edge of $\overline{SCK}$)	t_{SIK}	300			ns	
SI hold time (after rising-edge of $\overline{SCK}$)	t_{KSI}	450			ns	
SO output delay time (after falling-edge of $\overline{SCK}$)	t_{KSO}			850	ns	V_{DD} = 4.5 V to 6.0 V
				1200	ns	
INT0 pulse width, high, low	t_{IOH}, t_{IOL}	10			μs	
RESET pulse width, high, low	t_{RSH}, t_{RSL}	10			μs	

AC Characteristics (cont)

μPD75CG28E

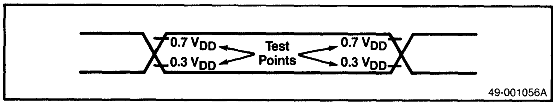
$T_A = -10°C$ to $+70°C$, $V_{DD} = +5 V \pm 10\%$

		Limits				Test
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
Cycle time (Note 1)	t_{CY}	4.0		200	μs	
$P0_0$ event input frequency	f_{P0}	0		500	kHz	
$P0_0$ input rise time	t_{POR}			0.2	μs	
$P0_0$ input fall time	t_{POF}			0.2	μs	
$P0_0$ input pulse width, high, low	t_{POH}, t_{POL}	0.8			μs	
$\overline{\text{SCK}}$ cycle time	t_{KCY}	3.0			μs	Input
		4.0			μs	Output
$\overline{\text{SCK}}$ pulse width, low	t_{KL}	1.8			μs	Output
$\overline{\text{SCK}}$ pulse width, high	t_{KH}	1.3			μs	Input
SI set-up time (to rising-edge of $\overline{\text{SCK}}$)	t_{SIK}	300			ns	
SI hold time (after rising-edge of $\overline{\text{SCK}}$)	t_{KSI}	450			ns	
SO output delay time (after falling-edge of $\overline{\text{SCK}}$)	t_{KSO}			850	ns	
INT0 pulse width, high, low	t_{IOH}, t_{IOL}	10			μs	
RESET pulse width, high, low	t_{RSH}, t_{RSL}	10			μs	
Data input delay time from address	t_{ACC}			700	ns	
Data input delay time from $\overline{\text{CE}}$	t_{CE}			700	ns	
Input hold time after address	t_{IH}	0			ns	

Note:

(1) $t_{CY} = 2/f_{CC}$ or $2/f_C$

AC Waveform Measurement Points (Except CL1)



Oscillation Characteristics

μPD7527A/28A

$T_A = -10°C$ to $+70°C$, $V_{DD} = 2.7 V$ to $6.0 V$

Parameter	Symbol	Min	Typ	Max	Unit	Conditons
System clock oscillation frequency (Note 1)	f_{CC}	300	400	500	kHz	R = 39 kΩ ±2% V_{DD} = 4.5 V to 6.0 V
		150	200	250	kHz	R = 82 kΩ ±2%
System clock CL1 input frequency (Note 2)	f_C	10		610	kHz	V_{DD} = 4.5 V to 6.0 V
		10		290	kHz	
CL1 input rise time (Note 2)	t_{CR}			0.1	μs	
CL1 input fall time (Note 2)	t_{CF}			0.1	μs	
CL1 input pulse width, low (Note 2)	t_{CL}	0.7		50	μs	
CL1 input pulse width, high (Note 2)	t_{CH}	1.63		50	μs	V_{DD} = 4.5 V to 6.0 V

μPD75CG28E

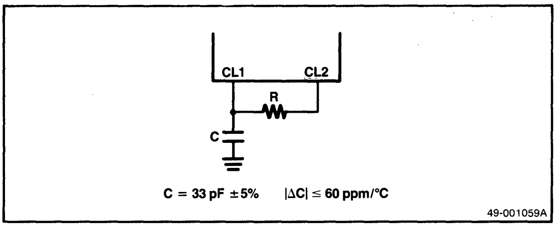
$T_A = -10°C$ to $+70°C$, $V_{DD} = 5 V \pm 10\%$

		Limits				Test
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
System clock oscillation frequency (Note 1)	f_{CC}	300	400	500	kHz	R=39 kΩ ± 2%
		110	150	190	kHz	R=110 kΩ ± 2%
System clock CL1 input frequency (Note 2)	f_C	10		500	kHz	
CL1 input rise time (Note 2)	t_{CR}			0.2	μs	
CL1 input fall time (Note 2)	t_{CF}			0.2	μs	
CL1 input pulse width, high, low	t_{CH}, t_{CL}	0.8		50	μs	

Note:

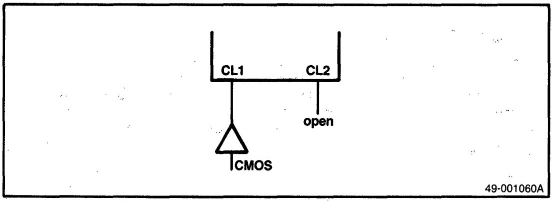
(1) R, C (see figure 3). (2) External clock (see figure 4).

Figure 3. Recommended RC Oscillator Circuit



C = 33 pF ±5% |ΔC| ≤ 60 ppm/°C

Figure 4. Recommended External Clock Circuit



Stop Mode Low Voltage Data Retention Characteristics

μPD7527A/28A

$T_A = -10°C$ to $+70°C$

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention supply current	I_{DDDR}		0.3	10	μA	$V_{DDDR} = 2$ V (Note 1)
			7	30	μA	$V_{DDDR} = 2$ V (Note 2)
Data retention RESET input voltage high	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} +0.2	V	
RESET set-up time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	

μPD75CG28E

$T_A = -10°C$ to $+70°C$

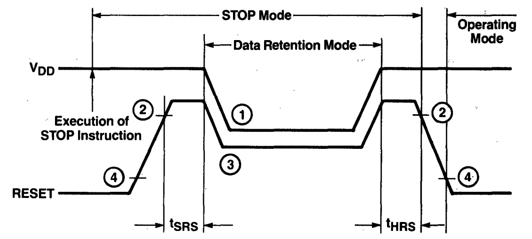
Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Data retention supply voltage	V_{DDDR}	2.0		5.5	V	
Data retention supply current	I_{DDDR}		7	30	μA	$V_{DDDR} = 2$ V
Data retention RESET input voltage high	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} +0.2	V	
RESET set-up time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	

Note:

(1) Without zero-cross detector

(2) With zero-cross detector

Data Retention Mode Timing



① V_{DDDR} ② V_{IH1} ③ V_{IHDR} ④ V_{IL1}

Note: In data retention mode, all inputs should be made lower level than V_{DDDR}.

49-001077A

μPD75CG28E EPROM Interface

A 4-Kbyte EPROM (2732) plugs into socket pins on top of the μPD75CG28E. A high input to MSEL selects μPD7527A mode and fixes the A_{11} output high level in order to access the upper 2-Kbytes of the 4-Kbyte EPROM. When MSEL is open, μPD7528A mode is selected. All EPROM addresses can be accessed because A_{11} functions as the MSB of the address. Figure 5 shows the address control unit. Figures 6 and 7 show the μPD75CG28E connected with the 2732.

Figure 8 shows the EPROM read timing. Data is read into the instruction buffer at the end of the T4 state. The chip enable ($\overline{CE}$) signal is made active during 2 states (T3, T4) in order to decrease the power consumption of the EPROM.

Figure 5. Address Control Unit

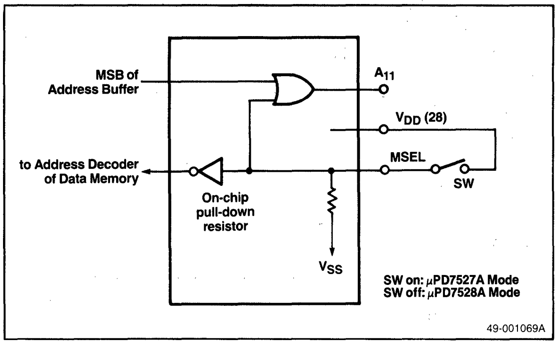


Figure 6. Connection with the 2732 (μPD7527A Mode)

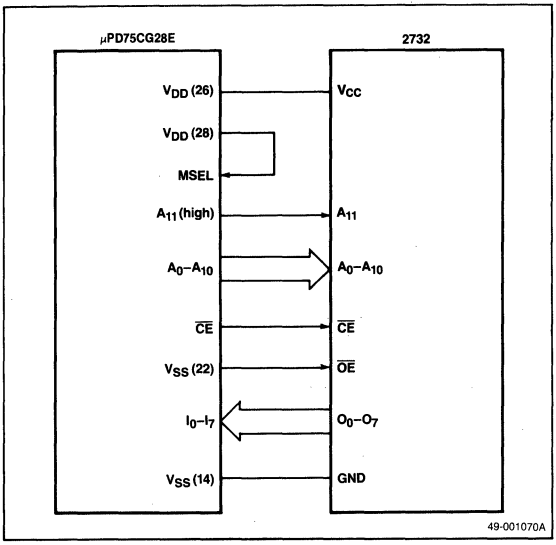


Figure 7. Connection with the 2732 (μPD7528A Mode)

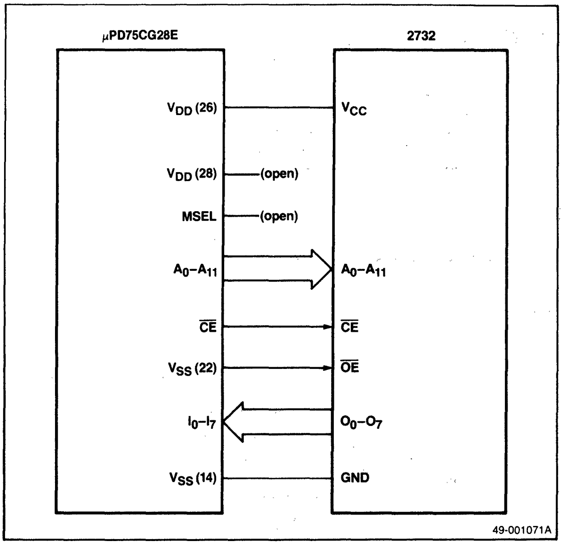
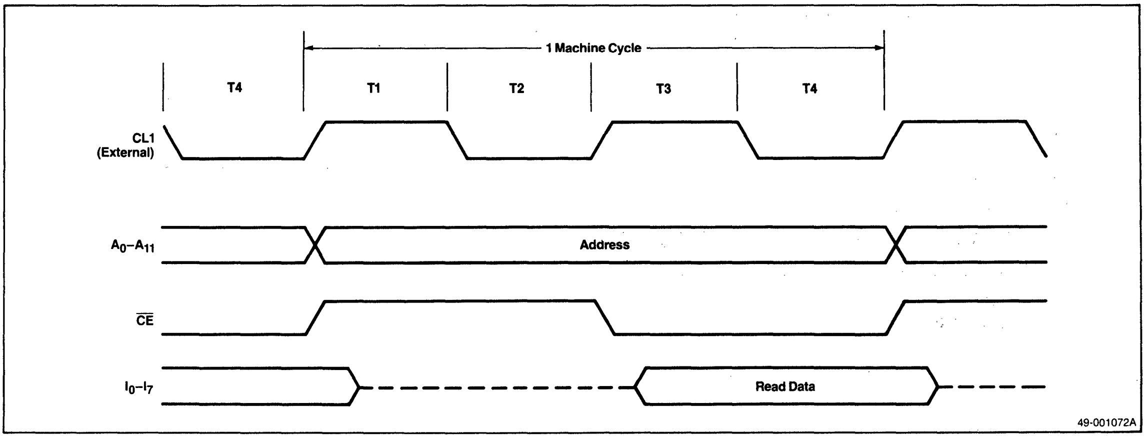
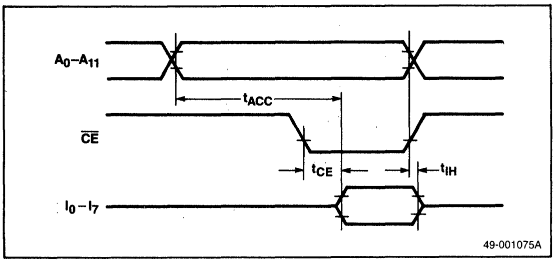


Figure 8. EPROM Read Timing

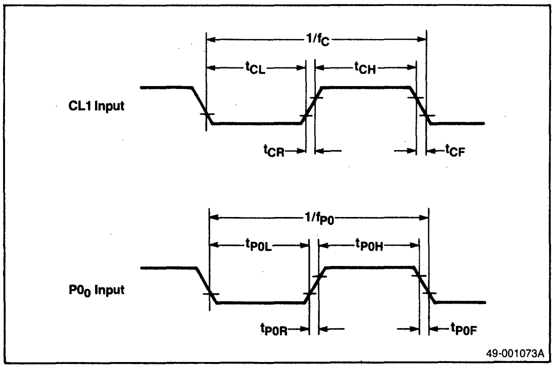


Timing Waveforms

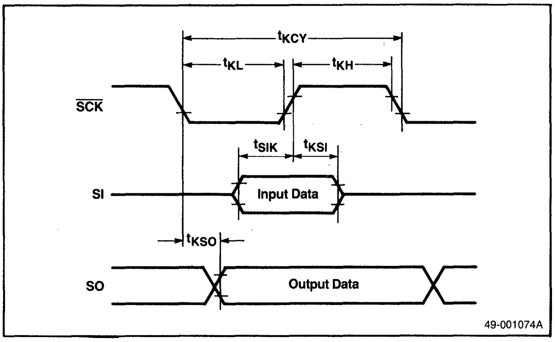
EPROM (μPD75CG28E only)



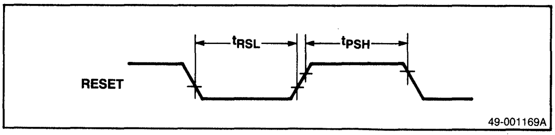
Clock



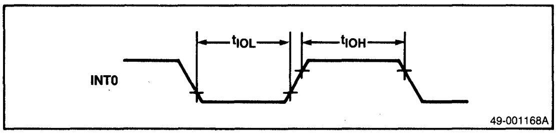
Serial Interface



Interrupt Input



Reset Input



Differences Among the μPD7527A/28A/CG28E

	μPD75CG28E	μPD7527A	μPD7528A
Program memory	4 Kbyte EPROM (2732) connectable on top	On-chip 2 Kbyte ROM	On-chip 4 Kbyte ROM
Data memory (RAM)	160 × 4	128 × 4	160 × 4
High-voltage output lines	All open-drain outputs	On-chip load capacitor or open drain output (bit by bit, mask optional)	
V_{LOAD} pin	No		
Zero-cross detection	Yes	Mask optional	
Package	42-pin ceramic piggyback DIP bottom pin compatible with μPD7527A / 28A	42-pin plastic DIP 42-pin plastic shrink DIP	
Power supply	5 V	2.7 V to 6.0 V	

NEC
NEC Electronics Inc.

μPD7533/75CG33E 4-Bit, Single-Chip CMOS Microcomputers With A/D Converter

Description

The μPD7533 is a 4-bit, single-chip CMOS microcomputer with a 4-channel, 8-bit A/D converter, 8-bit timer/event counter, and an 8-bit serial interface. The μPD7533 has 30 I/O lines, 8 of which can be used to directly drive LEDs. The μPD7533 executes 67 instructions of the μPD7500 series "A" instruction set.

The A/D converter has various temperature monitoring applications that can be used with household electrical appliances, such as air conditioners and electric ovens. Other applications include health monitoring equipment and cameras.

The μPD75CG33E consists of a 28-pin socket "piggybacked" on the lower 42-pin ceramic DIP. This socket is configured to hold either a 2732A or 2764 EPROM. For engineering purposes, programs can be tried and debugged before ROM code submission.

Features

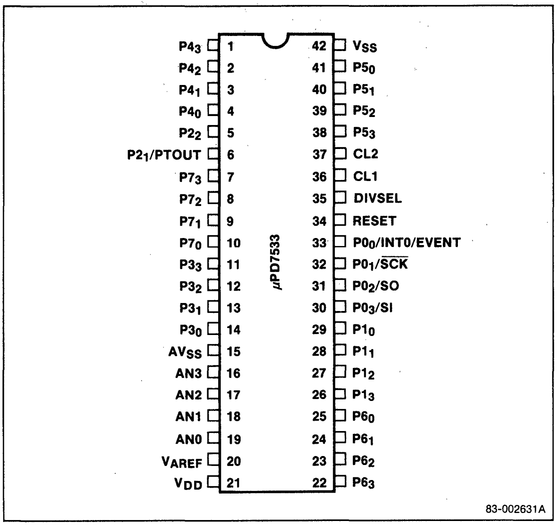
- □ 4-bit single chip microcomputer
- □ 67 instructions (subset of μPD7500 series set A)
- □ Instruction cycle
  - — 5 μs at 5 V, 400-kHz clock at ceramic oscillation, DIVSEL = high
  - — 10 μs at 5 V, 400-kHz clock at ceramic oscillation, DIVSEL = low
- □ Program memory (ROM): 4096 words x 8 bits
  - — External in the μPD75CG33E
- □ Data memory (RAM): 160 words x 4 bits
- □ 8 high current output lines for LED direct drive
- □ Input/output ports
  - — Two 4-bit input ports
  - — One 2-bit output port
  - — One 4-bit output port
  - — Three 4-bit input/output ports (two of these can function in 8-bit units)
  - — One 4-bit input/output port usable at bit level
- □ Interrupts: two internal and one external
- □ 8-bit serial interface
- □ Standby operation
  - — STOP mode
  - — HALT mode
- □ On-chip system clock oscillator
  - — Ceramic resonator
  - — Full or 1/2 oscillation frequency
- □ CMOS technology
- □ Low power consumption
- □ Single power supply

Ordering Information

Part Number	Package Type	Maximum Frequency of Operation
μPD7533C	42-pin plastic DIP	510 kHz
μPD7533CU	42-pin plastic shrink DIP	510 kHz
μPD7533G-22	44-pin plastic QFP	510 kHz
μPD75CG33E	42-pin ceramic piggyback DIP	510 kHz

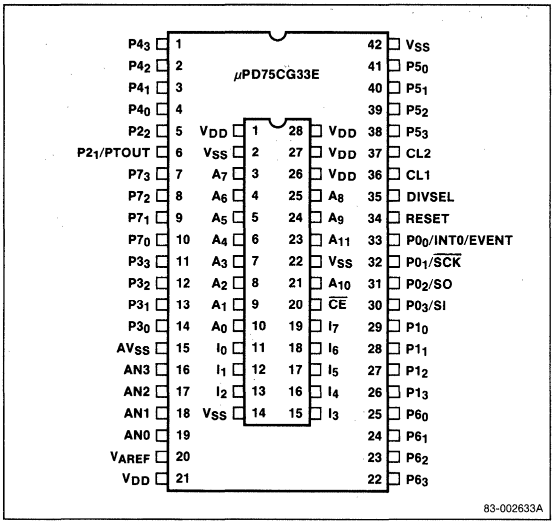
Pin Configurations

42-Pin Plastic DIP or Plastic Shrink DIP

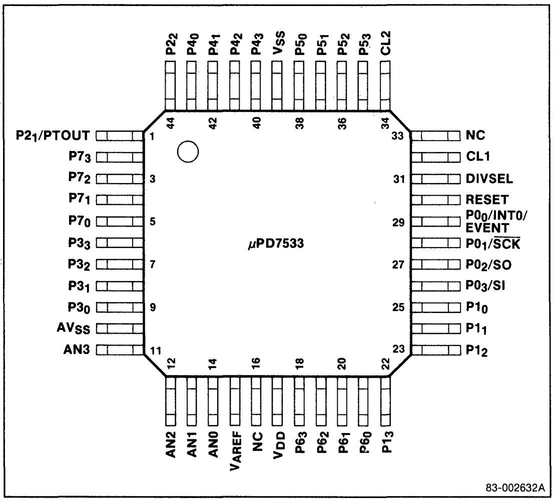


Pin Configurations (cont)

42-Pin Ceramic Piggyback DIP



44-Pin Plastic QFP



Pin Identification

42-Pin DIP, Shrink DIP, and Piggyback DIP

No.	Symbol	Function
1-4	$P4_3$-$P4_0$	I/O port 4
5, 6	$P2_2$, $P2_1$/PTOUT	Port 2 output
7-10	$P7_3$-$P7_0$	I/O port 7
11-14	$P3_3$-$P3_0$	Port 3 output
15	A_{VSS}	A/D converter ground
16-19	AN3-AN0	Analog input
20	V_{AREF}	A/D reference voltage input
21	V_{DD}	Positive power supply
22-25	$P6_3$-$P6_0$	I/O port 6
26-29	$P1_3$-$P1_0$	Port 1 input
30	$P0_3$/SI	Port 0 input/Serial input
31	$P0_2$/SO	Port 0 input/Serial output
32	$P0_1$/$\overline{SCK}$	Port 0 input/(I/O) Serial clock
33	$P0_0$/INT0/EVENT	Port 0 input/Interrupt 0/Event input
34	RESET	RESET input
35	DIVSEL	System clock selection input
36, 37	CL1, CL2	External clock input/System clock terminal
38-41	$P5_3$-$P5_0$	I/O port 5
42	V_{SS}	Ground

Pin Identification (cont)

44-Pin QFP

No.	Symbol	Function
1, 44	$P2_1$/PTOUT, $P2_2$	Port 2 output
2-5	$P7_3$-$P7_0$	I/O port 7
6-9	$P3_3$-$P3_0$	Port 3 output
10	A_{VSS}	A/D converter ground
11-14	AN3-AN0	Analog input
15	V_{AREF}	A/D reference voltage input
17	V_{DD}	Positive power supply
18-21	$P6_3$-$P6_0$	I/O port 6
22-25	$P1_3$-$P1_0$	Port 1 input
26	$P0_3$/SI	Port 0 input/Serial input
27	$P0_2$/SO	Port 0 input/Serial output
28	$P0_1$/$\overline{SCK}$	Port 0 input/(I/O) Serial clock
29	$P0_0$/INT0/EVENT	Port 0 input/Interrupt 0/Event input
30	RESET	RESET input
31	DIVSEL	System clock selection input
32, 34	CL1, CL2	External clock input/System clock
35-38	$P5_3$-$P5_0$	I/O port 5
39	V_{SS}	Ground
40-43	$P4_3$-$P4_0$	I/O port 4
16, 33	NC	No connect

28-Pin EPROM Socket on 42-pin Piggyback DIP

No.	Symbol	Function
1, 26-28	V_{DD}	Positive power supply
2, 14, 22	V_{SS}	Ground
20	$\overline{CE}$	Chip enable output
3-10, 21, 23-25	A_0-A_{11}	Address bus
11-13, 15-19	I_0-I_7	Data bus

Pin Functions

$P0_0$-$P0_3$ [Port 0]

$P0_0$-$P0_3$ function as port 0. $P0_0$ also functions as a count pulse input pin for the timer/event counter (EVENT) or as interrupt 0 (INT0). $P0_1$ also functions as a serial clock input/output pin ($\overline{SCK}$) for the serial interface. $P0_2$ functions as a serial data output pin (SO) and pins $P0_3$ as a serial data input pin (SI). The $P0_1$/$\overline{SCK}$ and $P0_2$/SO pins are three-state input/output.

The shift mode register (SM_0-SM_3) determines the operation mode of the port 0 input/output pins; however, the data on $P0_0$-$P0_3$ can be loaded into the accumulator at any time by executing a port input instruction (IP/IPL). This is possible even when $P0_1$-$P0_3$ are functioning as the serial interface.

After a RESET, $P0_0$-$P0_3$ become input ports (high impedance).

$P1_0$-$P1_3$ [Port 1]

$P1_0$-$P1_3$ function as port 1. Execution of an IP or IPL instruction reads data present on $P1_0$-$P1_3$ into the accumulator. Tie any unused lines of $P1_0$-$P1_3$ to V_{DD} or V_{SS}.

$P2_1$-$P2_2$ [Port 2]

$P2_1$-$P2_2$ function as port 2 with an output latch. When an output instruction (OP/OPL) to port 2 is executed, the middle 2 bits (A_1 and A_2) of the accumulator are latched by the output latch and, at the same time, output to $P2_1$-$P2_2$.

After being written once, the output latch contents remain until they are rewritten by an output instruction or a reset. The status of the corresponding output signal also remains. After a reset, the output latch contents become undefined, all output signals are disabled, and the output drivers are turned off.

$P2_1$ is also used as an output pin (PTOUT) for the timer-out F/F signal (PTOUT). Bit 3 (CM_3) of the clock mode register controls the PTOUT output. When CM_3 is 1, TOUT is ORed with the $P2_1$ output latch contents and sent to the output driver. Therefore, to output the $P2_1$ output latch contents, reset CM_3 to 0 to inhibit the TOUT signal.

Note that soon after the RESET signal is asserted, CM_3 is reset and TOUT is inhibited. However, since the output latch contents are undefined after a reset, to output the TOUT signal, first write 0 in the $P2_1$ output latch and then set CM_3 to 1 to output TOUT.

P3$_0$-P3$_3$ [Port 3]

P3$_0$-P3$_3$ function as port 3 with an output latch. When an output instruction to port 3 is executed, the accumulator contents are latched and output.

Once data is written in the output latch, the data is held until the next output instruction to port 3 is executed or RESET is asserted. After a reset, the output latch contents become undefined and the output driver is turned off.

P4$_0$-P4$_3$ [Port 4]
P5$_0$-P5$_3$ [Port 5]

P4$_0$-P4$_3$ function as port 4 and P5$_0$-P5$_3$ function as port 5. When an input instruction is executed, the data on these pins is read into the accumulator. When an output instruction is executed, the accumulator contents are latched and output. After the data is written into the latch, it is held until the next output instruction to ports 4 or 5 is executed, or RESET is asserted.

Ports 4 and 5 can work as a pair enabling data (input with the IP54 instruction and output with the OP54 instruction) in 8-bit units. The high four bits of data are from the accumulator and the low four bits are from memory (addressed by HL).

Ports 4 and 5 automatically set in the input mode (high impedance output) after a reset or when the input instructions to these ports are executed. After a reset, the output latch contents become undefined. Both ports 4 and 5 can drive LEDs directly.

Note that after the port changes from output mode to input mode, the data on the line is unstable when the input instruction that changes the mode is first executed. It is strongly recommended that you re-execute the input instruction considering the input/output mode switching time. This will insure reading stable data.

The bit manipulation instruction affects the specified bit only. So when the output latch contents are undefined, (immediately after a reset), initialize the output latch contents with an output instruction before the bit manipulation instruction is executed.

P6$_0$-P6$_3$ [Port 6]

P6$_0$-P6$_3$ function as the 4-bit input latched, three-state output port. The individual lines can be programmed as either inputs or outputs.

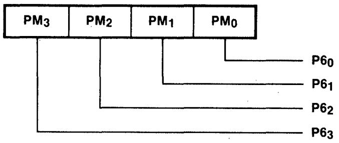
In input mode, data present at this port is read into the accumulator by the execution of an IP or IPL instruction. Accumulator data written to this port by the execution of an OP, OPL, ANP, or ORP instruction is statically latched, and remains unchanged until rewritten. This data, however, is not output since the output buffer is disabled and placed in the high impedance state.

In output mode, accumulator data written to the specified port line by the execution of the OP, OPL, ANP, or ORP instruction is statically latched and output to the P6$_n$ pin. Data present at P6$_n$ is read into the accumulator by the execution of the IP or IPL instruction, making it possible to read the contents of the P6$_n$ output latch.

All lines of port 6 are initialized to the high impedance state at Reset. Leave any unused lines open (if outputs) or tied to V$_{DD}$ or V$_{SS}$ (if inputs).

The port 6 mode select register (MSR) controls the function of the individual port 6 lines. The execution of the OP or OPL instruction loads the port 6 MSR with the accumulator contents. The 4-bit immediate data operand or the contents of the L register must be set to 0EH. Figure 1 shows the format of the port 6 MSR.

Figure 1. Port 6 MSR Format



PM$_n$	Port 6 Selection
0	P6$_n$ input [output buffer high-impedance]
1	P6$_n$ output [output buffer on]

83-002634A

$P7_0$-$P7_3$ [Port 7]

Port 7 is a 4-bit input or latched three-state output port. The execution of an IP or IPL instruction execution reads data present at this port into the accumulator. Accumulator data written to this port by the execution of an OP, OPL, ANP, or ORP instruction is statically latched and remains unchanged until rewritten.

Upon reset, all lines are initialized to the high-impedance state. Leave any unused lines open (if outputs) or tied to V_{DD} or V_{SS} (if inputs).

AN0-AN3 [A/D Input Terminal]

AN0-AN3 are the 4-channel A/D converter input terminals. The A/D converter uses a successive approximation method.

V_{AREF} [A/D Converter Positive Reference]

The voltage on V_{AREF} determines the full scale analog voltage.

AV_{SS} [A/D Converter Ground]

AV_{SS} is the ground for the A/D circuit.

CL1, CL2 [Clock]

CL1 and CL2 connect external oscillator elements to the system clock. Connect a ceramic resonator to these pins. If an external clock is used, place a buffer between the clock source and the CL1 and CL2 pins.

When connecting the oscillation parts to the CL1 and CL2 pins, use the shortest wiring possible. Ground the capacitor as close to the V_{SS} pin as possible.

DIVSEL [System Clock Divider Selection Input]

DIVSEL selects whether the system clock runs at ceramic oscillation frequency, or at one-half the ceramic oscillation frequency. If a logic 0 (V_{SS}) is connected to DIVSEL, the system clock is one-fourth the ceramic oscillation. If DIVSEL is high, then the system clock will be one-half of the ceramic oscillation.

RESET [Reset]

A high on RESET activates this input.

V_{DD} [Power Supply]

V_{DD} is the positive power supply pin.

V_{SS} [Ground]

V_{SS} is the ground pin.

Pin Functions, μPD75CG33 EPROM

A_0-A_{11} [EPROM Address]

A_0-A_{11} output the contents of the EPROM program address counter. A reset leaves A_0-A_{11} undefined.

I_0-I_7 [Data Bus]

I_0-I_7 input the contents of the EPROM data bus.

$\overline{CE}$ [Chip Enable]

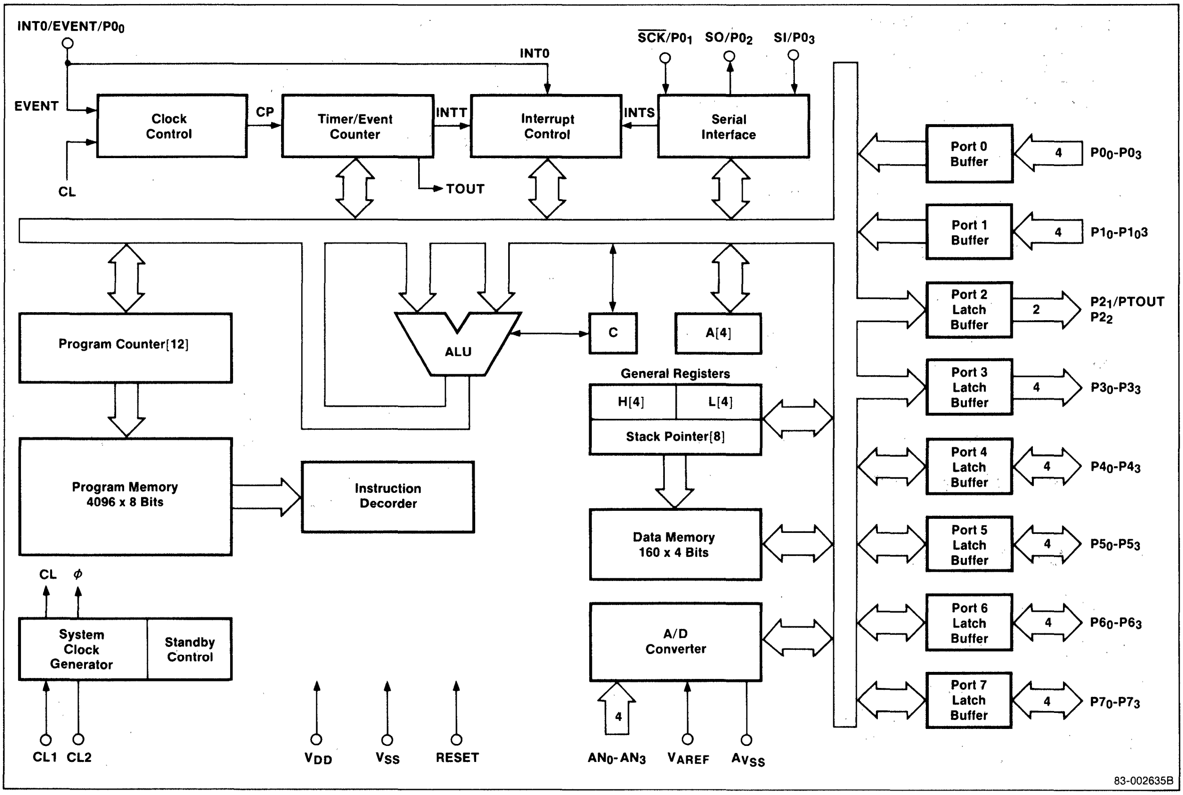
$\overline{CE}$ outputs the EPROM chip enable signal. (Active low.)

V_{DD} [Power Supply], V_{SS} [Ground]

V_{DD} is the positive power supply pin with the same voltage as the lower portion pin 21. V_{SS} is the ground pin with the same voltage as the lower portion pin 42. The following voltages are supplied to the 2764 or 2732A pins from V_{DD} or V_{SS}.

Pin Number			
2764	2732A	Symbol	Voltage
1	20	V_{PP}	V_{DD} pin 21 = +5 V
28	24	V_{CC}	V_{DD} pin 21 = +5 V
22	20	$\overline{OE}$	V_{SS} pin 42 = 0 V
2	—	A_{12}	V_{DD} pin 21 = +5 V
14	12	V_{SS}	V_{SS} pin 42 = 0 V

Block Diagram



Absolute Maximum Ratings

$T_A = 25°C$

Power supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_I	−0.3 V to V_{DD} + 0.3 V
Output voltage, V_O	−0.3 V to V_{DD} + 0.3 V
High level output current, I_{OH}	−17 mA (1 pin)
	−20 mA (all output ports)
Low level output current, I_{OL}	17 mA (1 pin)
	80 mA ports 2,3,4,7 (total pins)
	80 mA ports 0,5,6
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C
A/D V_{SS}, AV_{SS}	−0.3 to +0.3 V
A/D reference, V_{AREF}	−0.3 V to V_{DD}

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

$T_A = 25°C$, $V_{DD} = 0$ V

		Limits			
Parameter	Symbol	Min	Max	Unit	Test Conditions
Input capacitance	C_{IN}		15	pF	f =1 MHz Unmeasured pins are 0 V.
Output capacitance	C_{OUT}		15	pF	
I/O capacitance	C_{IO}		15	pF	

DC Characteristics

T_A = −10 to +70°C, V_{DD} = 2.7 to 6.0 V, DIVSEL = 1

		Limits			Test
Parameter	**Symbol**	**Min**	**Max**	**Unit**	**Conditions**
High level input voltage (other than CL1, CL2)	V_{IH1}	0.7 V_{DD}	V_{DD}	V	Conditions specified by oscillation characteristics
High level input voltage (CL1, CL2)	V_{IH2}	V_{DD} − 0.5	V_{DD}	V	
Low level input voltage (other than CL1, CL2)	V_{IL1}	0	0.3 V_{DD}	V	
Low level input voltage (CL1, CL2)	V_{IL2}	0	0.5	V	
High level output voltage	V_{OH}	V_{DD} − 1.0		V	V_{DD} = 4.5–6.0 V I_{OH} = −1 mA except P63
		V_{DD} − 0.5		V	I_{OH} = −100 µA
		V_{DD} − 0.5	V_{DD} − 0.2	V	V_{DD} = 4.5–6.0 V I_{OH} = −2 mA (P63 only)
Low level output voltage	V_{OL}	0.6 (typ)	2.0	V	V_{DD} = 4.5–6.0 V I_{OL} = 15 mA
		0.7 (typ)	2.5	V	(75CG33: V_{DD} = 4.5 - 6.0 V)
			0.4	V	I_{OL} = 1.6 mA
			0.5	V	I_{OL} = 400 µA
High level input leakage current (other than CL1, CL2)	I_{LIH1}		3	µA	V_{IN} = V_{DD}
High level input leakage current (CL1, CL2)	I_{LIH2}		20	µA	V_{IN} = V_{DD}
Low level input leakage current (other than CL1, CL2)	I_{LIL1}		−3	µA	V_{IN} = 0 V
Low level input leakage current (CL1, CL2)	I_{LIL2}		−20	µA	V_{IN} = 0 V
High level output leakage current	I_{LOH}		3	µA	V_{OUT} = V_{DD}
Low level output leakage current	I_{LOL}		−3	µA	V_{OUT} = 0 V

DC Characteristics

T_A = −10 to +70°C, V_{DD} = 2.7 to 6.0 V, DIVSEL = 1

		Limits			Test
Parameter	**Symbol**	**Min**	**Max**	**Unit**	**Conditions**
Supply current	I_{DD1}	1.0 (typ)	3.0	mA	Operating mode: f_{CC} = 500 kHz
	I_{DD2}	250 (typ)	750	µA	HALT mode: f_{CC} = 500 kHz
		300 (typ)	900	µA	(75CG33: V_{DD} = 4.5 − 6.0 V; f_{CC} = 500 kHz)
	I_{DD3}	0.1 (typ)	10	µA	STOP mode
		25 (typ)	200	µA	(75CG33: V_{DD} = 4.5 − 6.0 V)

AC Characteristics

T_A = −10 to +70°C, V_{DD} = 2.7 to 6.0 V

		Limits			Test
Parameter	**Symbol**	**Min**	**Max**	**Unit**	**Conditions**
Cycle time	t_{CY}	3.92	200	µs	V_{DD} = 4.5–6.0 V
		9.52	200	µs	
EVENT input frequency	f_E	0	510	kHz	V_{DD} = 4.5–6.0 V
		0	210	kHz	
EVENT input high duration	t_{EH}	0.8		µs	V_{DD} = 4.5–6.0V
EVENT input low duration	t_{EL}	2.2		µs	
SCK cycle time	t_{KCY}	4.0		µs	Input V_{DD} = 4.5–6.0 V
		3.92		µs	Output V_{DD} = 4.5–6.0 V
		10.0		µs	Input
		9.52		µs	Output
SCK high, low level duration	t_{KH}, t_{KL}	1.8		µs	Input V_{DD} = 4.5–6.0 V
		1.76		µs	Output V_{DD} = 4.5–6.0 V
		4.8		µs	Input
		4.6		µs	Output
SI setup time (SCK high)	t_{SIK}	300		ns	
SI hold time (SCK high)	t_{KSI}	450		ns	
SCK low to SO output delay time	t_{KSO}		850	ns	V_{DD} = 4.5–6.0 V
			1200		
INT0 high, low level duration	t_{IOH}, t_{IOL}	10		µs	
RESET high, low level duration	t_{RSH}, t_{RSL}	10		µs	

Data Memory, STOP Mode Data Retention Characteristics

$T_A = -10$ to $+70$°C

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention supply current	I_{DDDR}		0.1	10	μA	$V_{DDDR} = 2.0$ V
RESET setup time	t_{SRS}	0			μs	
Oscillation stabilizing time	t_{OS}	20			ms	Ceramic resonator: when V_{DD} greater than 4.5 V

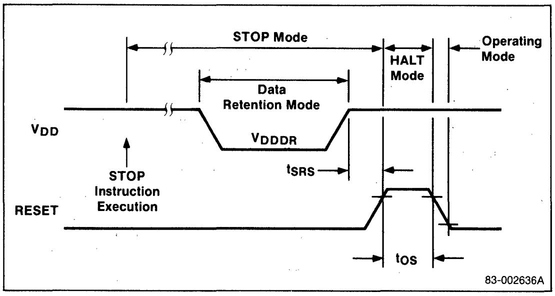
A/D Converter Characteristics

$T_A = -10$ to $+70$°C, $V_{DD} = +5.0$ V ±5%,
$V_{SS} = A_{VSS} = 0$ V, $V_{AREF} = V_{DD} - 0.5$ V to V_{DD}

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Resolution		8			Bits	
Absolute accuracy				±1.5	LSB	
Conversion time	t_{CONV}	9			t_{CYC}*	$V_{DD} - 0.5 \leq V_{AREF} \leq V_{DD}$
Sampling time	t_{SAMP}		1		t_{CYC}*	
Analog input voltage	V_{IAN}	A_{VSS}		V_{AREF}	V	
Analog input impedance	R_{AN}		1000		MΩ	
V_{AREF} current	I_{AREF}	0.4	1	2	mA	

* $t_{CYC} = \frac{2}{f_{CC}}$ (DIVSEL = 1)

Data Retention Timing



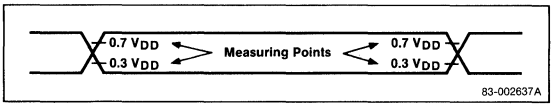
Oscillator Characteristics

$T_A = -10$ to $+70$°C, $V_{DD} = 2.7$ to 6.0 V, DIVSEL = 1

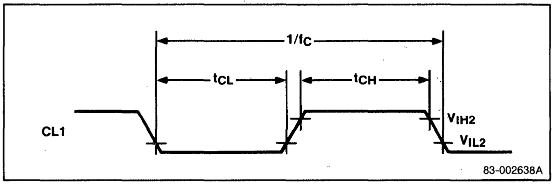
Oscillation	Configuration	Parameter	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Ceramic	See figure 3	Oscillation frequency (f_{CC})	390	500	510	kHz	$V_{DD} = 4.5$ to 6.0 V
			390	500	510	kHz	$V_{DD} = 4.0$ to 6.0 V
			390	500	510	kHz	$V_{DD} = 3.0$ to 6.0 V DIVSEL = 0
			390	400	410	kHz	$V_{DD} = 2.7$ to 6.0 V DIVSEL = 0
		Stabilization time	20			ms	V_{DD} greater than 4.5 V
External clock	See figure 3	CL1 input frequency	10		510	kHz	$V_{DD} = 4.5$ to 6.0 V
			10		210	kHz	
		CL1 input high, low level duration (t_{CH}, t_{CL})	1.0		50	μs	$V_{DD} = 4.5$ to 6.0 V
			1.0		50	μs	

Timing Waveforms

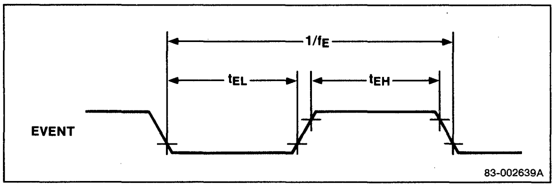
AC Timing Measuring Points (Except CL1)



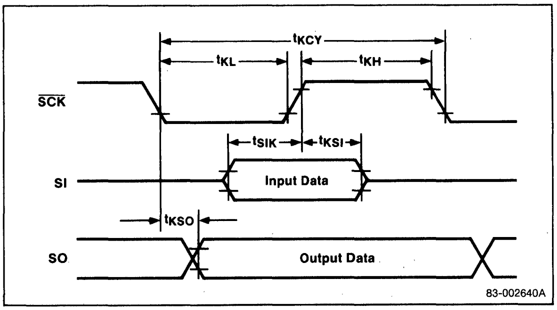
Clock Timing



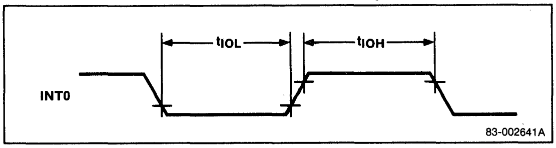
EVENT Timing



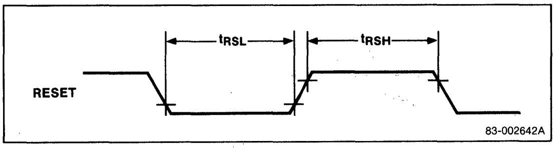
Serial Transfer Timing



Interrupt Input Timing



RESET Input Timing



Functional Description

System Clock Generator

The ceramic oscillator circuit generates the system clock for the μPD7533. Figure 2 shows that the oscillator circuit for the μPD7533 includes a ceramic oscillator, two divide-by-two circuits, the DIVSEL input, and control circuitry for the standby modes, HALT and STOP.

Figure 3 shows that the ceramic oscillator requires that a ceramic resonator be connected to the CL1 and CL2 pins. An external clock can also be input at CL1. In this case, the oscillator operates as an inverted buffer.

Figure 2 shows that the output frequency from the ceramic oscillator connects either directly to the clock selector or via a divide-by-two circuit. The selector is controlled by the DIVSEL line. If DIVSEL is low, the divide-by-two frequency is selected. This option is used during a low power operating mode. If DIVSEL is high, then the direct frequency is chosen. The output of the selector is used as system clock (CL), and is also divided by two to supply the CPU clock (ϕ).

Table 1 shows how DIVSEL selects the system and CPU clocks, and machine cycle timing.

Table 1. Clock Selection

DIVSEL	System Clock (CL)	CPU Clock (ϕ)	Machine Cycle
Low	200 kHz	100 kHz	10 μs
High	400 kHz	200 kHz	5 μs

Standby Control

The HALT F/F and the STOP F/F comprise the control circuitry for standby mode (figure 2). The STOP F/F is set by the STOP instruction. When the STOP F/F is set, the ceramic oscillator stops. The rising edge of the RESET input resets the STOP F/F.

The HALT instruction sets the HALT F/F and inhibits the input of the half-frequency divider which generates the CPU clock. As a result, only the CPU clock is stopped in HALT mode. The RELEASE signal resets the HALT F/F. RELEASE becomes active when any interrupt request flag is set, or at the falling edge of the RESET input.

While RESET is active, the HALT F/F is set, and the chip goes into the HALT mode. At a power-on Reset, the ceramic oscillation is driven when the RESET input signal becomes high.

Figure 3. Clock Driver Configuration

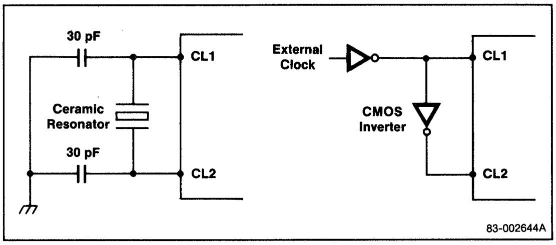
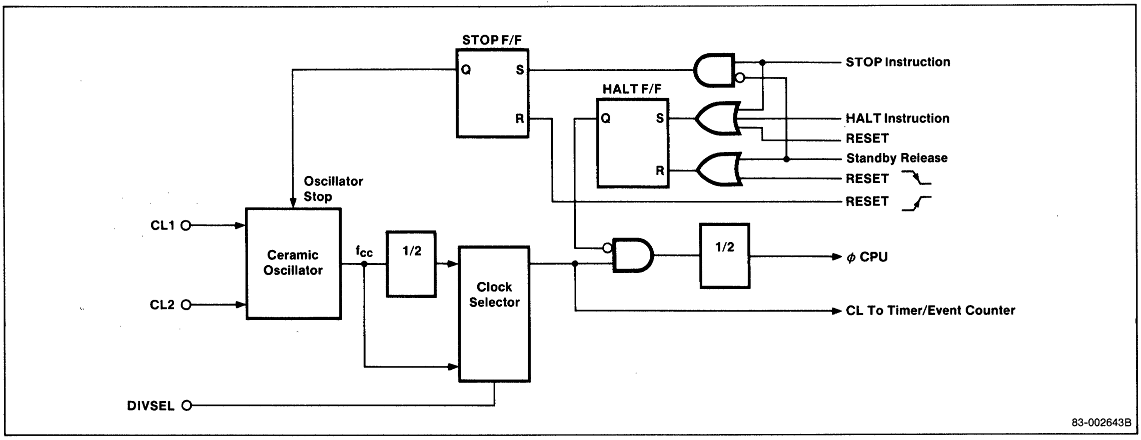


Figure 2. System Clock Generator



It takes a short period of time for the oscillator output to become stable. To prevent errors due to an unstable clock, the HALT F/F is set to inhibit the CPU clock while the RESET input is high. Therefore, the high-level pulse width for the RESET input should be wide enough to cover the required time for the ceramic resonator oscillation to stabilize.

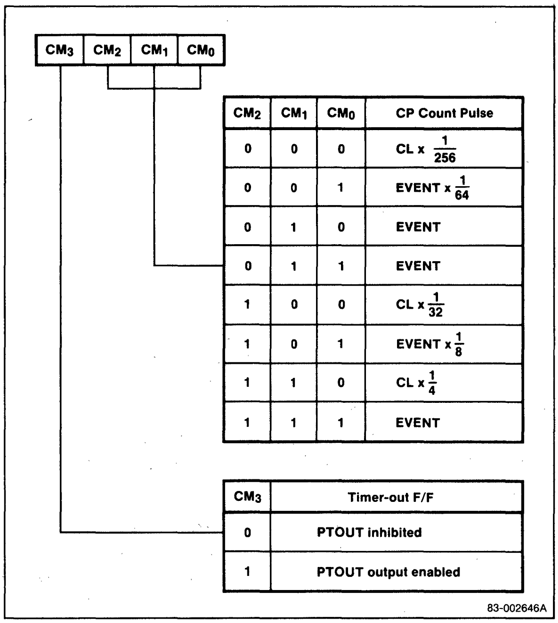
Clock Control

Figure 4 shows that the clock controller contains a 4-bit clock mode register (CM0-CM3), prescalers 1-3, and multiplexers. The clock controller selects the clock sources and prescalers, and supplies the count pulses (CP) to the timer/event counter. The clock sources are the system clock generator output (CL) or the EVENT pulse.

The OP 12 or OPL (L = 12) instruction sets codes in the clock mode register. CM3 designates the output of the timer-out signals. If CM3 = 1, the output of the timer-out F/F (TOUT) is available at the PTOUT (P21) pin.

Figure 5 shows the format of the clock mode register.

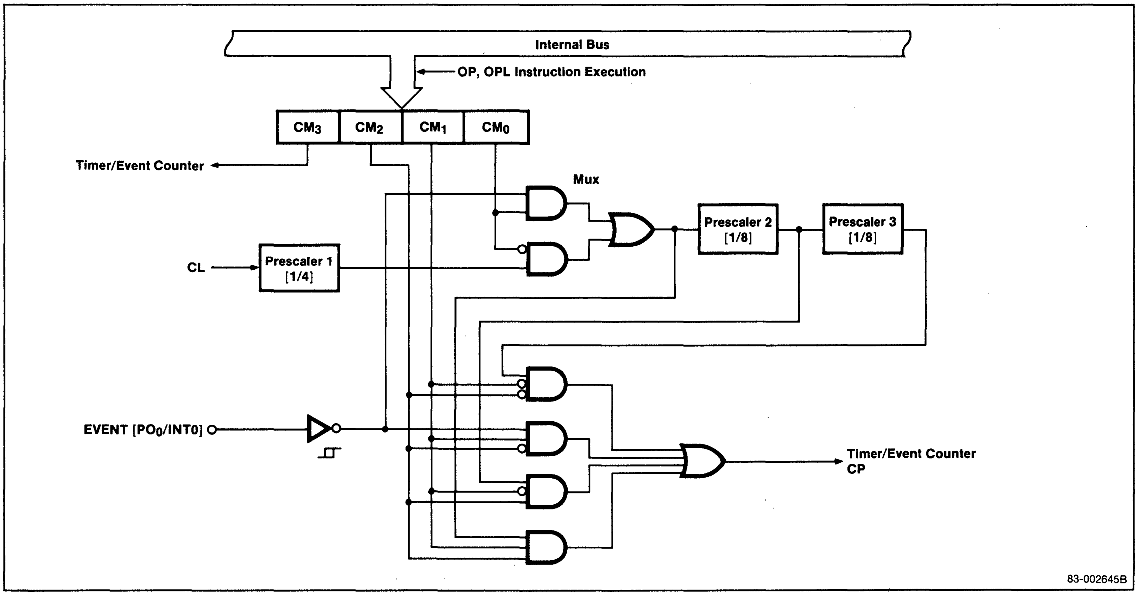
Figure 5. Format of Clock Mode Register



CM_2	CM_1	CM_0	CP Count Pulse
0	0	0	CL x $\frac{1}{256}$
0	0	1	EVENT x $\frac{1}{64}$
0	1	0	EVENT
0	1	1	EVENT
1	0	0	CL x $\frac{1}{32}$
1	0	1	EVENT x $\frac{1}{8}$
1	1	0	CL x $\frac{1}{4}$
1	1	1	EVENT

CM_3	Timer-out F/F
0	PTOUT inhibited
1	PTOUT output enabled

Figure 4. Clock Controller Block Diagram



Timer/Event Counter

Figure 6 shows the timer/event counter has an 8-bit count register, 8-bit modulo register, an 8-bit comparator, and a timer-out flip flop.

Timer Operation

After the TAMMOD instruction sets a count value in the modulo register and the TIMER instruction clears the contents of the count register, the timer starts counting count pulses (CP). If an external clock is used, the count pulses are synchronized with the rising edge of CL1 or the $P0_0$ input.

When the value of the modulo register equals the value of the count register, the comparator generates a coincidence signal (INTT) to set an interrupt request flag. Then it clears the count register to repeat the counting. In this manner, the timer functions as an interval timer whose interval is set by the modulo register.

Regardless of any instructions, the count pulses are always input into the count register, updating the count value. If the contents of the count register are equal to those of the modulo register, the INTT request flag is then set. For this reason, inhibit INTT interrupts when not using the timer.

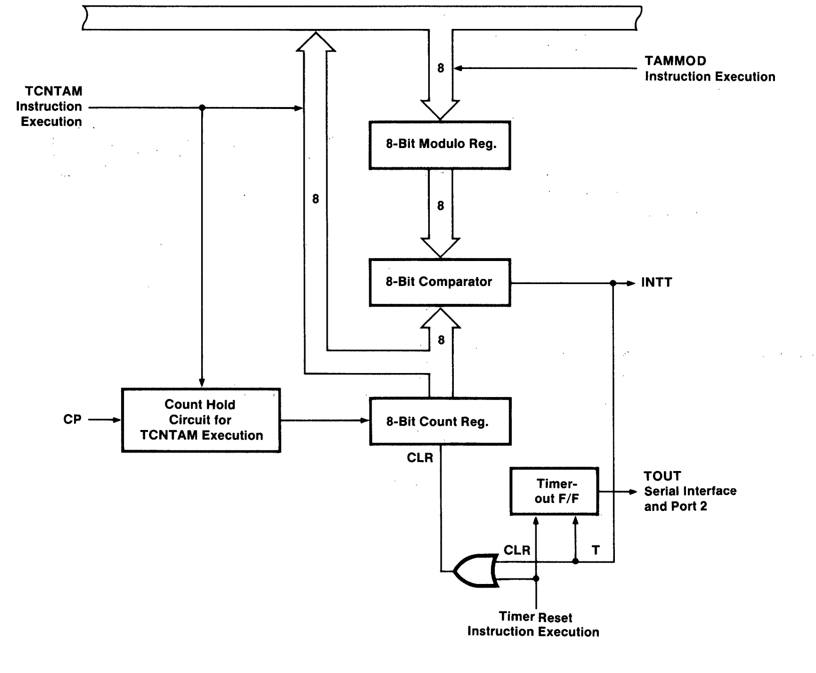
Event Counter Operation

To use the timer/event counter as an event counter, input the external event pulse into the $P0_0$ pin, and select $P0_0'$ as the count pulse (CP) for the clock controller. The count register counts the external event pulses input at the $P0_0$ pin, either as they are, or frequency divided.

As a result, the timer/event counter operates as an event counter that generates interrupts after observing the number of counts (events) specified by the modulo register. The TCNTAM instruction can read the current count at any time.

Set the modulo register with the number of count pulses minus one. If set to 0, no counting will occur because the counter register is held at 0 (both the detection of coincidence and zero-clearing are simultaneously made).

Figure 6. Block Diagram of Timer/Event Counter



83-002647B

Serial Interface

As figure 7 shows, the serial interface includes an 8-bit shift register, a 4-bit shift mode register, and a 3-bit counter.

The serial clock controls serial data I/O. At the falling edge of the serial clock ($\overline{SCK}$), the SO line outputs the most significant bit (7) of the shift register. The contents of the shift register are shifted by one bit at the rising edge of the next serial clock (n ←0 n+1). At the same time, the data on the SI line is loaded into the least significant bit (0) of the shift register.

The 3-bit counter (octal counter) counts up the serial clocks and generates an internal interrupt signal INTS at every count of 8 clocks (at the end of a 1-byte serial data transfer). It then sets the interrupt request flag (INT0/S RQF). The TAMSIO instruction sets data in the shift register during the transmission of serial data, then starts transmission. At the end of the transmission of each byte (8 bits) an internal interrupt (INTS) is generated.

The SIO instruction also starts the reception of serial data. The received data is taken from the shift register by executing the TSIOAM instruction after an interrupt (INTS) is generated by the reception of one byte of data.

The end of a 1-byte transfer can be confirmed by testing the INTS RQF with the SKI instruction instead of interrupt processing.

The following three types of serial clock sources are available: system clock ϕ, external clock ($\overline{SCK}$ input), and timer-out F/F output signal (TOUT). Bits SM_2-SM_0 of the shift mode register select the clock source.

If the system clock ϕ is chosen, execute the SIO instruction to supply the clock to the serial interface, controlling the input/output of serial data while ϕ is output from the $\overline{SCK}$ pin.

After eight ϕ pulses, the clock is automatically discontinued by holding the $\overline{SCK}$ output at a high level. Therefore, the input/output of serial data automatically stops after each byte has been transferred. Consequently, the software does not need to control the serial clock and the transfer rate is determined by the system clock frequency.

In this mode, after six machine cycles from the execution of the SIO, the TSIOAM instruction can read out the received data from the shift register or can write in the next transmit data.

Figure 8 shows the shift mode register format.

Figure 7. Serial Interface Block Diagram

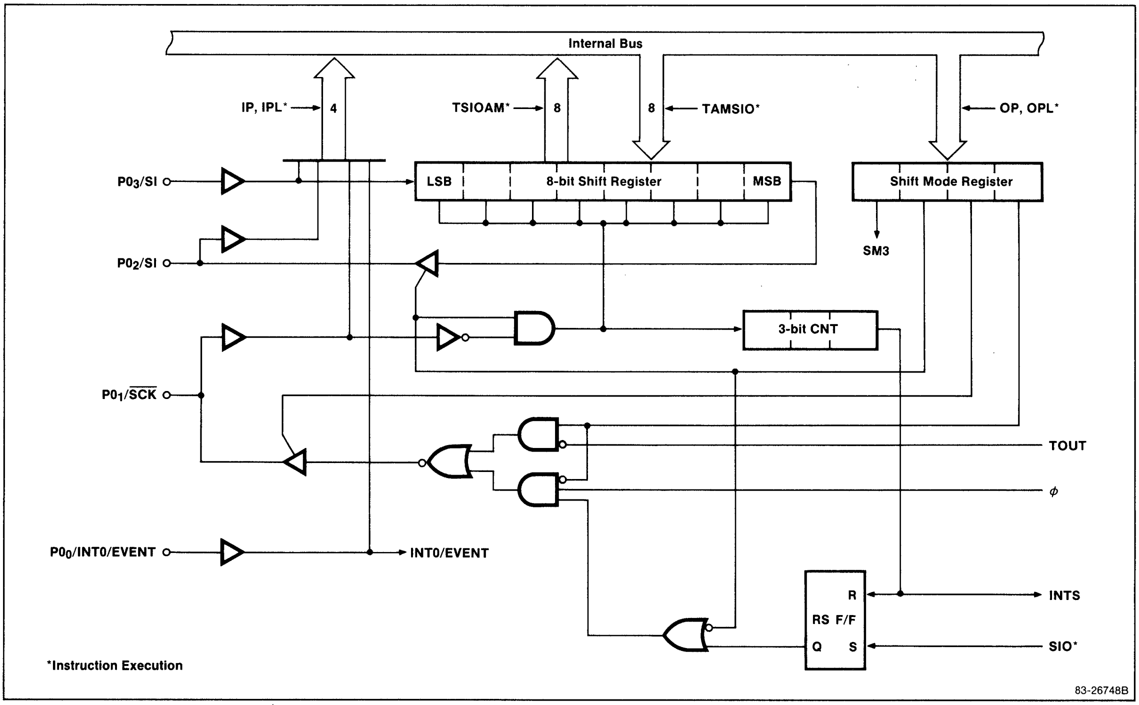


Figure 8. Format of Shift Mode Register

SM_2	SM_1	SM_0	PO_3/SI	PO_2/SO	$PO_1/\overline{SCK}$	Serial Operation
0	0	0	Port input	Port input	Port input	Stops
0	1	0			Outputs ϕ continuously	
0	1	1			Outputs TOUT continuously	
1	0	0	SI input	SO output	$\overline{SCK}$ input	Operates with external clock
1	1	0			$\overline{SCK}$ output (ϕ x 8)	Operates with ϕ
1	1	1			$\overline{SCK}$ output (TOUT)	Operates with TOUT

Bit SM_3 selects the interrupt source in the following manner:

SM_3	Interrupt Source
0	INTS
1	INT0

If the external clock ($\overline{SCK}$ input) is selected, the serial clocks are input from $\overline{SCK}$. When the eighth external serial clock is input, an internal interrupt (INTS) is generated, signalling the end of a 1-byte data transfer.

Since the serial clocks are not internally inhibited, the external clock must hold the signal high after eight clocks. The external serial clock determines the transfer rate. The serial interface can be operated from DC to the maximum rate in the electrical specifications.

If TOUT is selected, the half-frequency divided coincidence signal of the timer/event counter is the serial clock. This serial clock controls the input/output of the serial data and is output from the $\overline{SCK}$ pin.

The count pulse supplied to the timer/event counter and the value set in the modulo register determine the transfer rate. The end of a 1-byte data transfer is signalled by INTS. TOUT is not inhibited automatically, therefore the program should stop TOUT at intervals of 16.

To use the external clock or the TOUT signal, execute the SIO, TAMSIO or TSIOAM instructions while the serial clock ($\overline{SCK}$) is held high. Operation cannot be guaranteed if these intructions are executed over the rising or falling edge of $\overline{SCK}$, or at the low level.

In a system that does not require serial data transfer, the 8-bit shift register can be ued as a register with the serial operation stopped. The TSIOAM or TAMSIO instruction can read or write data.

Analog to Digital Converter

The μPD7533 integrates a 4-channel 8-bit A/D converter with separate positive reference and ground from the device power supply. Figure 9 shows that the A/D converter includes an A/D converter mode register, successive approximation (SA) register, and end of conversion (EOC) control circuitry.

A/D Converter Mode Register

The A/D converter mode register is a 4-bit internal port that controls the A/D circuitry. The lower two bits, ANI0 and ANI1, select which analog signal (AN0-AN3) is input to the A/D converter. The most significant bit, ADS, initiates the A/D conversion. If ADS is set to a logic 1, the analog signal selected by ANI1 and ANI0 is converted to 8-bit digital data. Upon completion of the data conversion, ADS is cleared to 0.

Figure 10 shows the format for the A/D conversion mode register.

Figure 10. A/D Conversion Mode Register Format

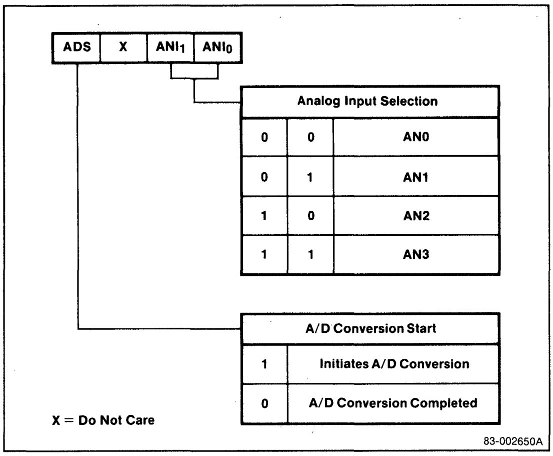
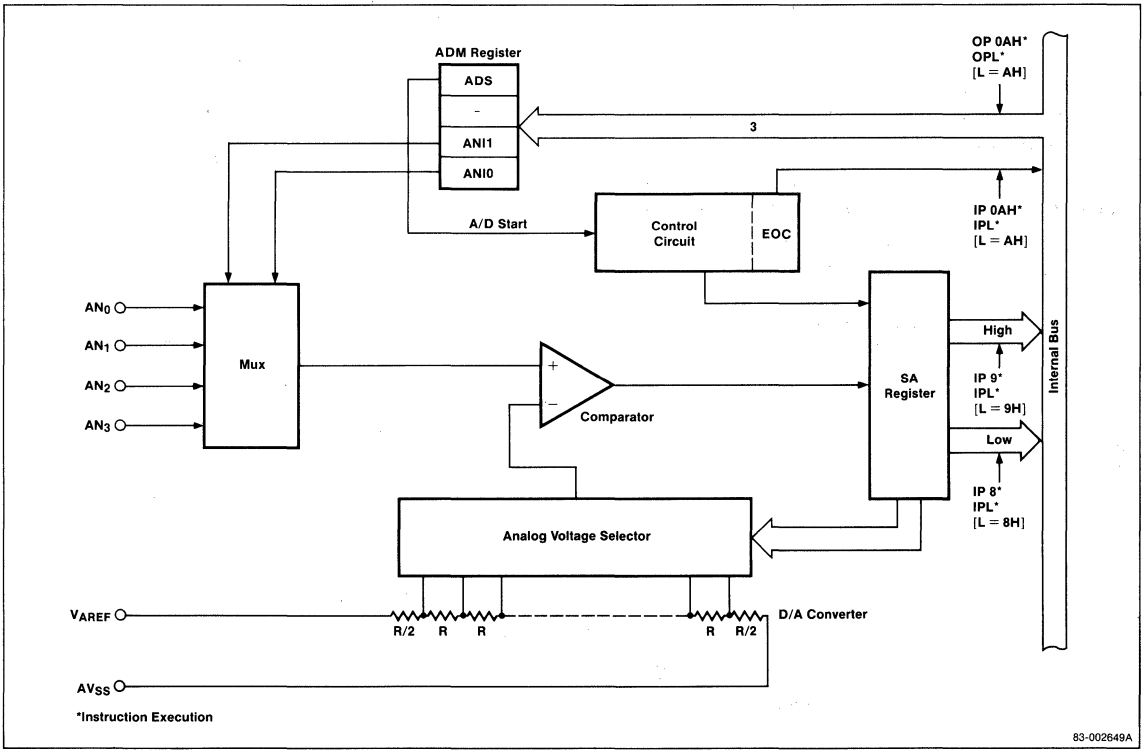


Figure 9. A/D Converter Block Diagram



Successive Approximation [SA]

The 8-bit data converted from the analog signal using the successive approximation method is stored in the SA register. When ADS is set to a logic 1, the contents of the SA register are undetermined. The SA register is set to 7FH after a reset.

End of Conversion [EOC] Flag

The EOC flag specifies the completion of an A/D conversion. When ADS is set to 1, the EOC flag is set to a logic 0 and an A/D conversion starts. When the 8-bit A/D conversion is complete, the EOC flag is set to a logic 1. The EOC flag resides in bit 2 of internal Port A. The IP 0AH or IPL instruction can read the contents of Port A when the L register is set to 0AH. The contents of Port A (other than bit 2) will be read as a logic 0. The EOC flag is set to 1 after a reset.

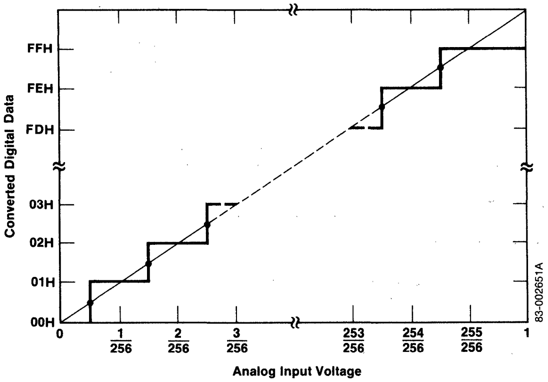
A/D Converter Operation

An OP 0AH or OPL instruction selects one of four analog signals and starts a conversion when the L register is set to 0AH. The lower two bits of the accumulator specify which analog signal will be converted. Bit 3 of the accumulator sets to 1 to initiate the A/D conversion. The A/D conversion requires 9 machine cycles for completion. When the conversion is complete, the EOC flag is set.

In order to assure an accurate data conversion, do not execute an output instruction when EOC is a logic 0.

Figure 11 shows how the analog input voltage corresponds to the converted digital data.

Figure 11. A/D Conversion Graph



Reading Converted Data

Internal port 9 specifies the upper four bits of the SA register. Therefore, execute an IP 9 or IPL (L = 9) instruction to read the data in the accumulator.

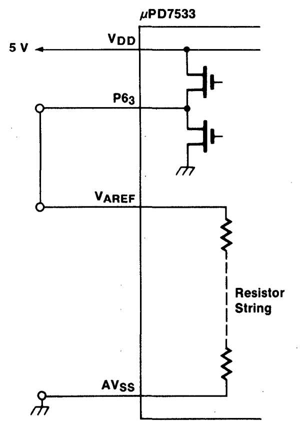
Internal port 8 specifies the lower four bits of the SA register. Therefore, execute an IP 8 or IPL (L = 8) instruction to read the data in the accumulator. Do not read the SA register until EOC is set to 1.

Figure 12 shows the configuration for the A/D converter reference voltage during standby mode.

Interrupt Function

The μPD7533 provides one external interrupt and two types of internal interrupts. The $P0_0$ pin is used as the input pin for external interrupt INT0. INT0 shares priority and vectored addresses with internal interrupt INTS. Figure 13 shows the interrupt controller block diagram.

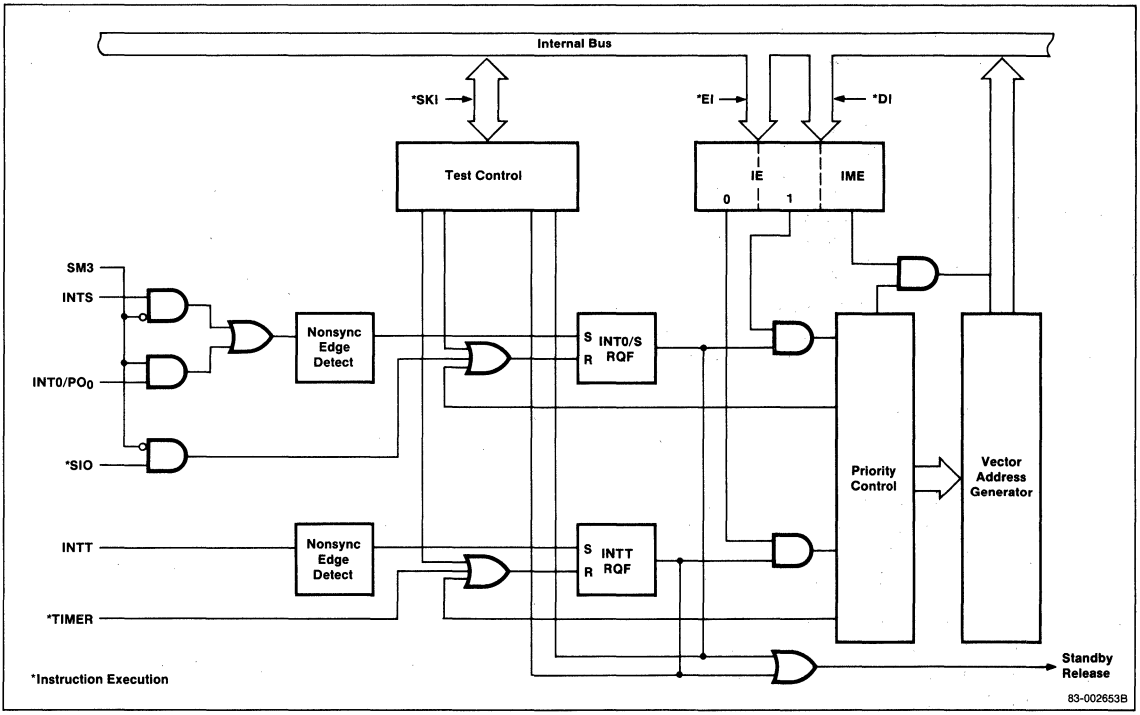
Figure 12. Configuration of V_{AREF} for Standby Mode Operation



$P6_3$ Output	V_{AREF}
High Level	V_{AREF} approximately = V_{DD}
Low Level or High Impedance	V_{AREF} approximately = 0 V

83-002652A

Figure 13. Interrupt Controller Block Diagram



Standby Function

The μPD7533 has two types of standby modes (STOP and HALT) to minimize power consumption during a program standby state. STOP mode is set by the STOP instruction and HALT mode by the HALT instruction.

When standby mode is set, program execution is stopped, and the contents of all internal registers and data memory are held. However, it is possible to operate the shift register and the timer/event counter. An interrupt or reset releases standby mode. Since an interrupt releases standby mode, neither STOP nor HALT modes can be set if an interrupt request flag is set. Therefore, when setting standby mode when there is a possibility of a request flag being set, first reset the interrupt request flag by processing the interrupt in advance or by executing the SKI instruction.

The major difference in the two modes is that crystal oscillation (CL) stops in STOP mode but does not stop in HALT mode.

In STOP mode, it is possible to go into data retention mode by lowering the power supply voltage. During data retention mode, all operation stops and only the data RAM stays intact.

Table 2 shows the differences between STOP and HALT modes.

Table 2. Differences Between STOP and HALT Modes

	Mode	
Operation	**STOP Mode**	**HALT Mode**
Ceramic Oscillation	X (1)	0 (2)
1/2 Ceramic Oscillation	X (1)	X (1)
CPU	X (1)	X (1)
Serial I/O	(3)	(2)
Timer/Event Counter	X (1)	0 (2)
A/D Converter	X (1)	0 (2)
Release of Standby Mode	RESET	INT0/S RQF NTT RQF RESET Input

Note:

(1) Not possible
(2) Possible
(3) Possible depending on clock source selected

STOP Mode

In STOP mode, ceramic oscillation and the half-frequency divider stop. The CPU stops and the operations requiring the system clock (CL, 0) stop.

Release from STOP mode is with the RESET input only. All other functions cease to operate.

In order to minimize power consumption, the current flowing through the resistor ladder of the A/D converter must be minimized. To minimize power consumption, turn off the power to the V_{AREF} pin.

Note that ceramic oscillation stops and disables the system clock during STOP mode by bringing CL2 to ground. Therefore, if the external clock is connected to CL1 and a STOP instruction is executed, the CPU will enter HALT mode instead.

HALT Mode

In HALT mode, only the half-frequency divider circuit stops in the clock generator circuit (CL operates, ϕ stops). Therefore, the CPU and the operation of the serial interface (when using ϕ as a serial clock) stop.

However, since the clock control circuit is still in operation, it can select the CL signal from the clock generator or the EVENT input and supply the count pulse (CP) to the timer/event counter.

Consequently, the timer/event counter can be operated in HALT mode. The serial interface operates if a serial clock other than ϕ (such as the external clock, TOUT signal) is selected. The HALT mode is released by the RESET input or an interrupt, even if the interrupt is disabled.

Release from Standby Mode by Interrupt

The standby mode is released when the interrupt request flag is set by an interrupt source, whether interrupts are disabled or enabled. However, the operations after release differ in each case.

If the interrupt master enable F/F is enabled, and if the interrupt is enabled, the corresponding interrupt routine is initiated after execution of one instruction after the STOP/HALT instruction. Then, the result flag is reset. If the corresponding bit of the interrupt enable register has been reset, execution of instructions starts after the STOP/HALT instruction, and the interrupt routine is not initiated. In this case, the request flag for release remains set. If necessary, reset the request flag with the SKI instruction.

If the interrupt master enable F/F is disabled, the instruction following the STOP/HALT instruction is executed regardless of the state of the interrupt enable register (interrupt routine is not initiated). In this case, the interrupt request flag is left set. If necessary, it can be reset by the SKI instruction.

After any release, operation resumes with the same register contents as before standby mode.

Release From Standby Mode with RESET

Both STOP and HALT modes are released unconditionally by the RESET input. Figure 14 shows the release timing.

If the device is reset during STOP mode, the low to high transition of the RESET pin will take the processor from STOP mode to HALT mode. When RESET goes high to low, the HALT mode is abandoned, and after a normal reset operation, the PC is initialized to 0. Only the data memory will stay intact during the HALT mode, but all registers become undefined.

If the device is reset during HALT mode, the high to low transition of RESET will release the device from standby mode. After a normal reset operation, the PC is initialized to 0. Only the data memory will stay intact during the HALT mode, but all registers become undefined.

Figure 15 shows the release from HALT mode by RESET.

Figure 14. Release from STOP mode by RESET

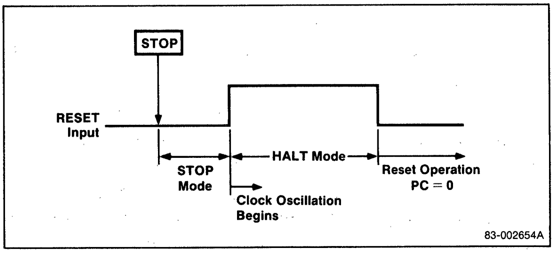
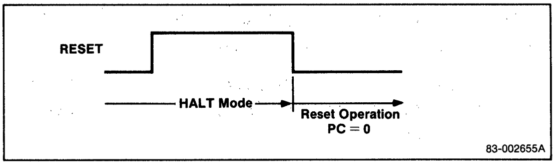


Figure 15. Release from HALT Mode by RESET



Reset Function

The μPD7533 is reset and initialized by the input of the RESET signal (active high).

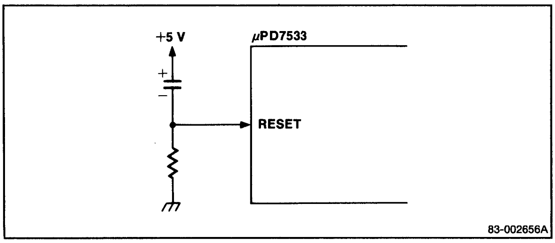
A RESET causes the CPU to initialize in the following manner:

- Program counter (PC) is cleared to 0
- Skip flags (SK1, SK0) and program status word (PSW) are reset to 0
- Timer/event counter:
  - — Count register = 00H
  - — Modulo register = FFH
  - — Timer-out F/F = 0
- Clock control circuitry:
  - — Clock mode register (CM_3-CM_0) = 0
  - — CP $= \frac{CL}{256}$
  - — Timer-out FF signal not output to PTOUT
  - — Prescalers 1-3 = 0
- Shift Mode Register (SM_3-SM_0) is cleared to 0.
  - — Shift operation stops
  - — Port 0 is in input mode (high impedance)
  - — INTS is selected interrupt source of INT0/S
- A/D converter circuit:
  - — ADM register is set to 0
  - — AN0 is selected
  - — SA register is set to 7FH
  - — EOC flag is set to logic 1
- Interrupt control circuit:
  - — Interrupt request flags = 0
  - — Interrupt master enable F/F = 0
  - — Interrupt enable register = 0
  - — All pending interrupts are cancelled
  - — All interrupts are disabled
- All Port 2-7 output buffers are turned off
- Contents of data memory and the following registers are undefined:
  - — Stack pointer (SP)
  - — Accumulator (A)
  - — Carry flag (C)
  - — General purpose registers (H,L)
  - — All port output latches
  - — Shift register

Power-on Reset Circuit

Figure 16 shows an example of the simplest power-on reset circuit using a resistor and a capacitor.

Figure 16. ***Power-on Reset Circuit***



NEC
NEC Electronics Inc.

μPD7537A/38A/75CG38E 4-Bit, Single-Chip CMOS Microcomputers With FIP® Driver

Description

The μPD7537A, μPD7538A, and μPD75CG38E are 4-bit, single-chip CMOS microcomputers with the μPD7500 architecture and FIP direct-drive capability.

The μPD7537A contains a 2048 × 8-bit ROM and a 128 × 4-bit RAM. The μPD7538A contains a 4096 × 8-bit ROM and a 160 × 4-bit RAM.

The μPD7537A/38A contains two 4-bit general purpose registers located outside RAM. The subroutine stack is implemented in RAM for greater depth and flexibility. The μPD7537A/38A typically executes 67 instructions with a 5μs instruction cycle time.

The μPD7537A/38A has one external and two internal edge-triggered hardware-vectored interrupts. An 8-bit timer/event counter and an 8-bit serial interface help to reduce software requirements.

Thirty-one high-voltage lines are organized into the 3-bit output port 2, the 4-bit output ports 3, 8, and 9, and the 4-bit I/O ports 4, 5, 10, and 11.

The low power consumption CMOS process allows the use of a power supply between 2.7 V and 6.0 V. Current consumption is less than 3.0 mA maximum, and can be further reduced in the halt and stop power-down modes.

The μPD75CG38E is a piggyback EPROM version of the μPD7537A/38A. Pin-compatible and function-compatible with the final, masked versions of the μPD7537A/38A, the μPD75CG38E is used for prototyping and for aiding in program development.

Features

□ 67 instructions
□ Instruction cycle:
 — Internal clock: 3.3μs/600 kHz, 5 V
 — External clock: 3.3μs/600 kHz, 5 V
□ Upwardly compatible with the μPD7500 series product family
□ 4,096 × 8-bit ROM (μPD7538A/75CG38E)
 2,048 × 8-bit ROM (μPD7537A)
□ 160 × 4-bit RAM (μPD7538A/75CG38E)
 128 × 4-bit RAM (μPD7537A)
□ 35 I/O lines
□ 31 high-voltage output lines that can directly drive a vacuum fluorescent diplay (FIP)
□ Can select either a pull-down resistor or open-drain output per 31 high-voltage outputs (mask optional)
□ Vectored interrupts: one external, two internal
□ 8-bit timer/event counter
□ 8-bit serial interface
□ Standby function (HALT, STOP)
□ Data retention mode
□ Zero-cross detector on $P0_0$/INT0 input (mask optional)
□ System clock (μPD7537A/7538A/75CG38E): on-chip ceramic oscillator
□ CMOS technology
□ Low power consumption
□ Single power supply
 — μPD7537A/7538A: 2.7 V to 6.0 V
 — μPD75CG38E: 5.0 V ± 10%

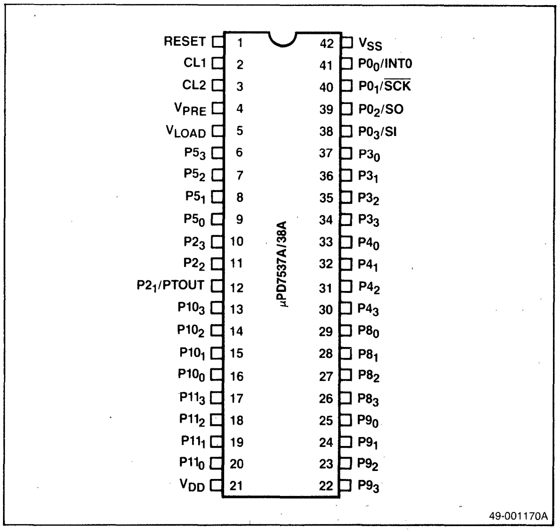
FIP is the registered trademark for NEC's fluorescent indicator panel (vacuum fluorescent display).

Ordering Information

Part Number	Package Type	Max Frequency of Operation
μPD7537AC / 38AC	42-pin plastic DIP	610 kHz
μPD7537ACU / 38ACU	42-pin plastic shrink DIP	610 kHz
μPD75CG38E	42-pin ceramic piggyback DIP	500 kHz

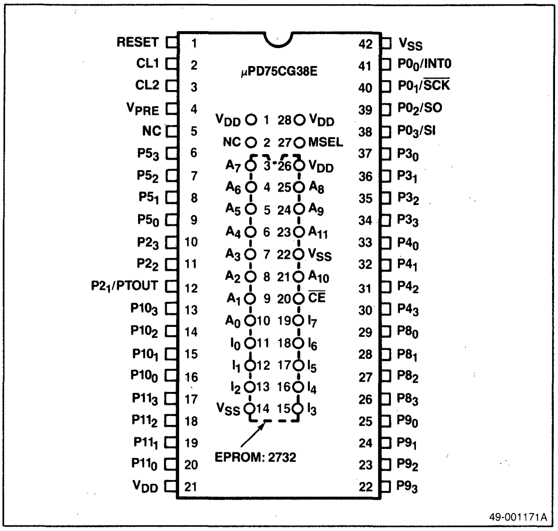
Pin Configurations

μPD7537A/38A 42-Pin Plastic DIP or Shrink DIP



Pin Configurations (cont)

μPD75CG38E 42-Pin Ceramic Piggyback DIP



Pin Identification

μPD7537A/38A and μPD75CG38E

No.	Symbol	Function
1	RESET	Reset input
2, 3	CL1, CL2	Clock pins
4	V_{PRE}	High-voltage output predriver supply
5	V_{LOAD}	High-voltage output option resistor supply 7537A / 38A only
6–9	$P5_0$–$P5_3$	High-voltage I / O port 5
10, 12	$P2_3$, $P2_2$ $P2_1$ / PTOUT	High-voltage output port 2, and output port from timer / event counter (PTOUT)
13–16	$P10_0$–$P10_3$	High-current, high-voltage I / O port 10
17–20	$P11_0$–$P11_3$	High-voltage, high-current I / O port 11
21	V_{DD}	Positive power supply
22–25	$P9_0$–$P9_3$	High-voltage, high-current output port 9
26–29	$P8_0$–$P8_3$	High-voltage, high-current output port 8
30–33	$P4_0$–$P4_3$	High-voltage I / O port 4
34–37	$P3_0$–$P3_3$	High-voltage output port 3
38 39 40 41	$P0_3$ / SI $P0_2$ / $\overline{SO}$ $P0_1$ / $\overline{SCK}$ $P0_0$ / INT0	4-bit input of port 0; or serial data input (SI), serial data output (SO), serial clock I / O ($\overline{SCK}$), and external interrupt input (INT0) or zero-cross detect input ($P0_0$).
42	V_{SS}	Ground

μPD75CG38E EPROM

No.	Symbol	Function
1	V_{DD}	Connection to pin 21 of μPD75CG38E
2	NC	No connection
3–10, 21, 24, 25	A_0–A_{10}	EPROM address output
11–13, 15–19	I_0–I_7	Data read input from the EPROM
14	V_{SS}	Connection to EPROM GND pin
20	CE	Chip enable output
22	V_{SS}	Supplies EPROM OE signal
23	A_{11}	Program counter MSB output
26	V_{DD}	Supplies V_{CC} to the EPROM
27	MSEL	Mode select input
28	V_{DD}	Supplies high-level signal to MSEL

Note:

(1) Output drivers on ports 2-5 and 8-11 are mask-optional. Accordingly, either an open-drain output or a pull-down resistor can be selected. V_{LOAD} is suitable for an output driver with a pull-down resistor.

(2) Ports 2-5 are suitable as FIP segment signal outputs, and ports 8-11 are suitable for FIP digit signal outputs.

(3) Ports 8-11 have high-current drive capability and can drive an LED directly.

Pin Functions, μPD7537A/38A and μPD75CG38E

RESET

System reset (input).

CL1, CL2

Connection to the ceramic oscillator. CL1 is the external clock input.

V_{PRE}

Negative power supply for high-voltage output predrivers (for ports 2-5, 8-11).

V_{LOAD}

Negative power supply for optional load resistors (pull-down resistors) of high-voltage output drivers (for ports 2-5, 8-11). This pin is only on the μPD7537A/38A.

$P5_3$-$P5_0$

4-bit, high-voltage I/O port 5.

$P2_1$-$P2_3$

3-bit, high-voltage output port 2.

PTOUT

Output port for the timer/event counter.

$P10_3$-$P10_0$

4-bit, high-voltage, high-current I/O port 10. Capable of bit set/reset by SPBL/RPBL instructions.

$P11_3$-$P11_0$

4-bit, high-voltage, high-current I/O port 11. Capable of bit set/reset by SPBL/RPBL instructions.

V_{DD}

Positive power supply.

$P9_3$-$P9_0$

4-bit, high-voltage, high-current output port 9. Capable of bit set/reset by SPBL/RPBL instructions.

$P8_3$-$P8_0$

4-bit, high-voltage, high-current output port 8. Capable of bit set/reset by SPBL/RPBL instructions.

$P4_3$-$P4_0$

4-bit, high-voltage I/O port 4.

$P3_3$-$P3_0$

4-bit, high-voltage output port 3.

$P0_0$-$P0_3$

4-bit input port 0. $P0_0$ is also used as the zero-cross detection input.

SI

Serial data input.

SO

Serial data output.

$\overline{SCK}$

Serial I/O clock.

INT0

External interrupt input.

V_{SS}

Ground.

Pin Functions, μPD75CG38E EPROM

MSEL

Changes the addressing area of the external EPROM and the on-chip RAM (with a pull-down resistor). Connecting a jumper between socket pins 27 (MSEL) and 28 (V_{DD}) selects μPD7537A mode (2-Kbyte EPROM, 128 × 4-bit RAM). Leaving MSEL open selects μPD7538A mode (4-Kbyte EPROM, 160 × 4-bit RAM).

A_0-A_{10}

Output the low-order 11 bits of the program counter (PC_0-PC_{10}). Used as EPROM address signals.

A_{11}

When MSEL is high level, A_{11} outputs high-level signals. When MSEL is open, A_{11} outputs the MSB of the PC, which is used as the most significant address signal of the 4-Kbyte EPROM 2732.

I_0-I_7

Input data read from the EPROM.

$\overline{CE}$

Outputs the chip enable signal to the EPROM.

V_{DD}

Pin 26 is electrically equivalent to the bottom V_{DD} pin and is used to supply V_{CC} to the EPROM. Pin 28 is electrically equivalent to the bottom V_{DD} pin and is used to supply the high level signal to MSEL. Pin 1 connects to pin 21 of μPD75CG38E.

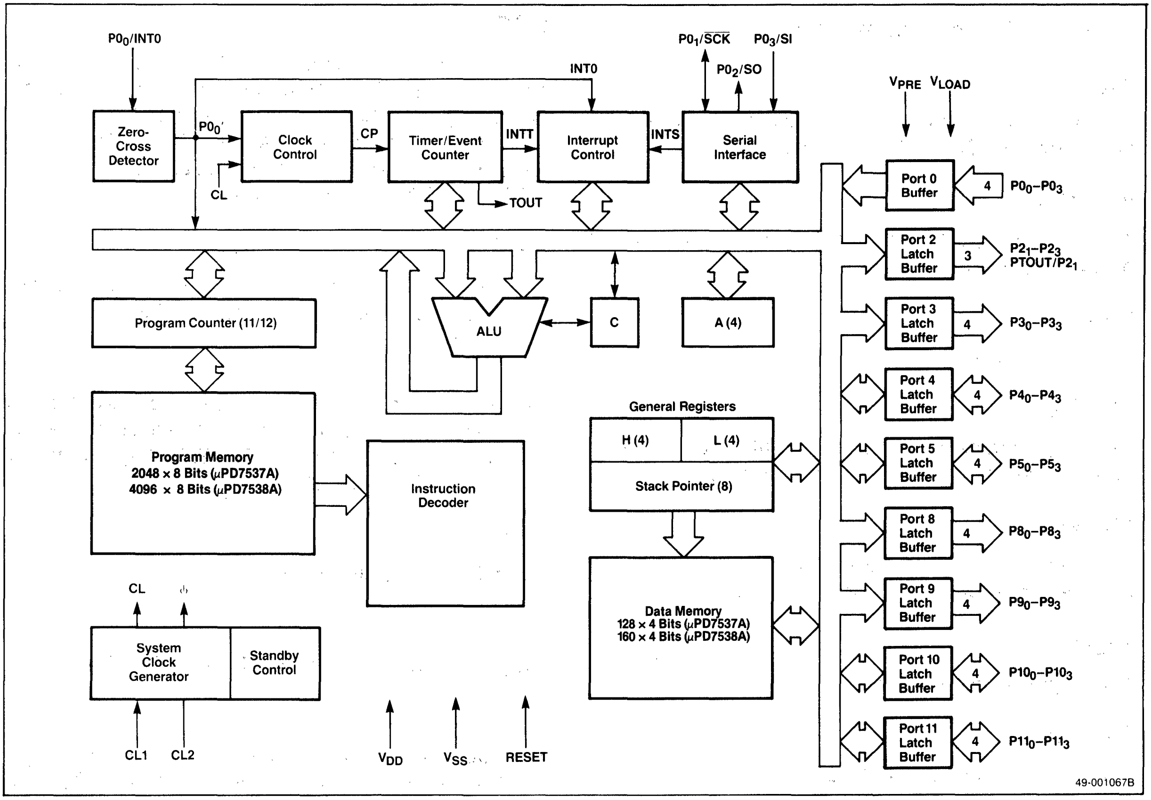
V_{SS}

Pin 14 is electrically equivalent to the bottom V_{SS} pin in voltage, and is connected to the EPROM GND pin. Pin 22 is electrically equivalent to the bottom V_{SS} pin and is used to supply the OE signal to the EPROM.

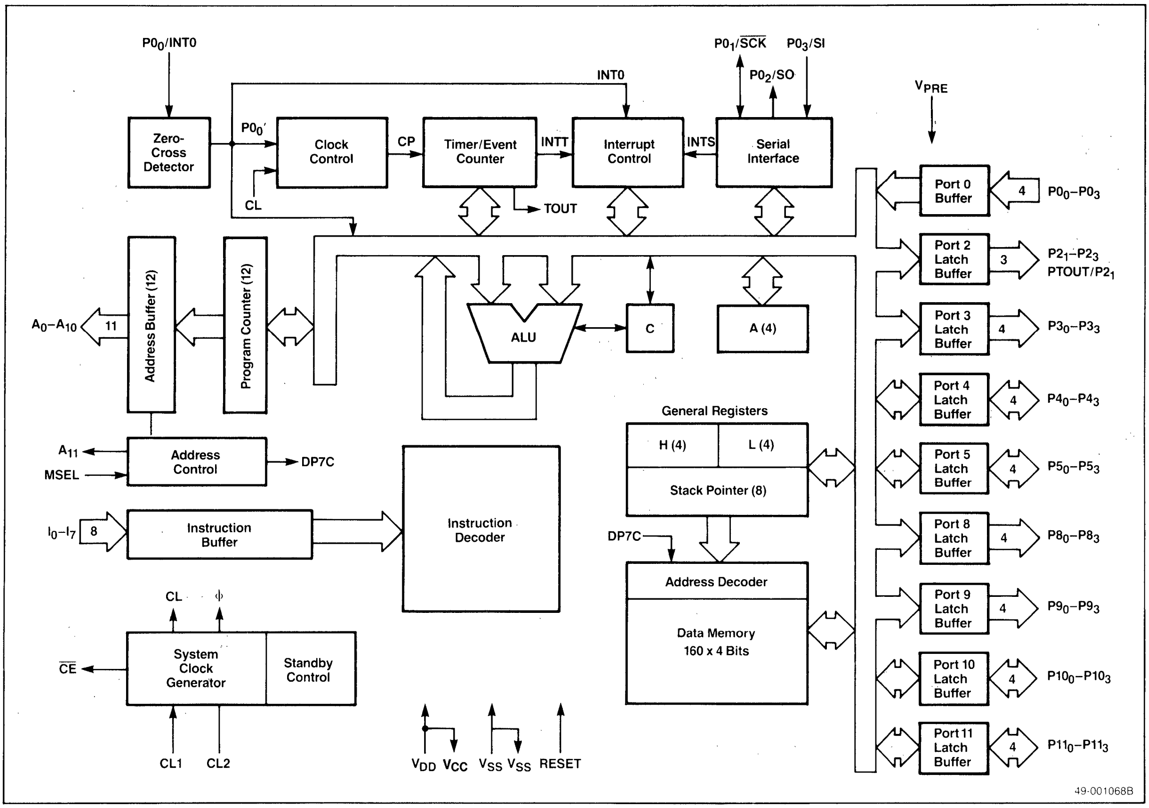
Instruction Set

Refer to the User's Manual. The instruction set appears also as subset A4 in the data sheet for the μPD7500 series of single-chip microcomputers.

Block Diagram, µPD7537A/38A



Block Diagram, μPD75CG38E



Absolute Maximum Ratings

$T_A = 25°C$

Power supply voltage, V_{DD}	−0.3 V to +7 V
Power supply voltage, V_{LOAD} (μPD7537A / 38A)	V_{DD} − 40 V to V_{DD} +0.3 V
Power supply voltage, V_{PRE}	V_{DD} − 12 V to V_{DD} +0.3 V
Input voltage, except ports 4, 5, 10, 11, V_{IN}	−0.3 V to V_{DD} +0.3 V
Input voltage, ports 4, 5, 10, 11, V_{IN}	V_{DD} − 40 V to V_{DD} +0.3 V
Output voltage, except ports 2–5, 8–11, V_O	−0.3 V to V_{DD} +0.3 V
Output voltage, ports 2–5, 8–11, V_O	V_{DD} − 40 V to V_{DD} +0.3 V
Output current high, per pin: P0₁, P0₂; I_{OH}	−15 mA
Output current high, per pin: ports 2–5, 8–11; I_{OH}	−30 mA
Output current high, ports 3, 4, 8, 9 total, I_{OH}	−55 mA
Output current high, ports 2, 5, 10, 11 total, I_{OH}	−55 mA
Output current low, per pin, I_{OL}	15 mA
Output current low, all ports total, I_{OL}	15 mA
Operating temperature, T_{OPT}	−10°C to +70°C
Storage temperature, T_{STG}	−65°C to +150°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of the specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

$T_A = 25°C$, $V_{DD} = 0$ V, f = 1.0 MHz, Unmeasured pins returned to GND

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input capacitance	C_I			15	pF	P0₀–P0₃
Output capacitance	C_O			15	pF	Port 2
				35	pF	Ports 3, 8, 9
I / O capacitance	C_{IO}			15	pF	P0₁, P0₂
				35	pF	Ports 4, 5, 10, 11

DC Characteristics

µPD7537A/38A

$T_A = -10°C$ to $+70°C$, $V_{DD} = +2.7 V$ to $6.0 V$

		Limits				
Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions
Input voltage, low	V_{IL1}	0		$0.3 V_{DD}$	V	Port 0, RESET
	V_{IL2}	0		0.5	V	CL1
	V_{IL3}	$V_{DD}-35$		$0.3 V_{DD}$	V	Ports 4, 5, 10, 11
Input voltage, high	V_{IH1}	$0.7 V_{DD}$		V_{DD}	V	Port 0, RESET
	V_{IH2}	$V_{DD}-0.5$		V_{DD}	V	CL1
	V_{IH3}	$0.7 V_{DD}$		V_{DD}	V	Ports 4, 5, 10, 11; $4.5 V \leq V_{DD} \leq 6.0 V$
		$V_{DD}-0.5$		V_{DD}	V	Ports 4, 5, 10, 11; $2.7 V \leq V_{DD} < 4.5 V$
Output voltage, low	V_{OL}			0.4	V	PO_1, PO_2; $4.5 V \leq V_{DD} \leq 6.0 V$; $I_{OL}=1.6 mA$
				0.5	V	PO_1, PO_2; $I_{OL}=400 \mu A$
Output voltage, high	V_{OH}	$V_{DD}-2.0$			V	Ports 2–5, $I_{OH}=-4 mA$ (Note 1)
		$V_{DD}-2.0$			V	Ports 8–11, $I_{OH}=-10 mA$ (Note 1)
		$V_{DD}-2.0$			V	Ports 2–5, $I_{OH}=-2 mA$ (Note 2)
		$V_{DD}-2.0$			V	Ports 8–11, $I_{OH}=-5 mA$ (Note 2)
		$V_{DD}-1.0$			V	PO_1, PO_2; $I_{OH}=-1 mA$ (Note 3)
		$V_{DD}-0.5$			V	PO_1, PO_2; $I_{OH}=-100 \mu A$
Input leakage current, low	I_{LIL1}			−3	µA	$V_{IN}=0 V$; PO_0–PO_3 (Note 4)
	I_{LIL2}			−40	µA	$V_{IN}=0 V$; PO_0 (Note 5)
	I_{LIL3}			−20	µA	$V_{IN}=0 V$; CL1
	I_{LIL4}			−10	µA	$V_{IN}=V_{DD}-35 V$; ports 4, 5, 10, 11
Input leakage current, high	I_{LIH1}			3	µA	$V_{IN}=V_{DD}$; PO_0 (Note 4)–PO_3
	I_{LIH2}			40	µA	$V_{IN}=V_{DD}$; PO_0 (Note 5)
	I_{LIH3}			20	µA	$V_{IN}=V_{DD}$; CL1
	I_{LIH4}			80	µA	$V_{IN}=V_{DD}$; ports 4, 5, 10, 11
Output leakage current, low	I_{LOL1}			−3	µA	$V_O=0 V$; PO_1, PO_2
	I_{LOL2}			−10	µA	$V_O=V_{DD}-35 V$; ports 2–5, 8–11

		Limits				
Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions
Output leakage current, high	I_{LOH1}			3	µA	$V_O=V_{DD}$; except ports 4, 5, 10, 11
	I_{LOH2}			80	µA	$V_O=V_{DD}$; ports 4, 5, 10, 11
Supply current, normal operation	I_{DD1}		1.5	4.0	mA	$V_{DD}=5 V \pm 10\%$, $f_{CC}=600 kHz$
Supply current, HALT mode (Note 6)	I_{DD2}		700	1800	µA	$V_{DD}=5 V \pm 10\%$, $f_{CC}=600 kHz$ (Note 4)
			230	700	µA	$V_{DD}=3 V \pm 10\%$, $f_{CC}=600 kHz$ (Note 4)
			710	1840	µA	$V_{DD}=5 V \pm 10\%$, $f_{CC}=600 kHz$ (Note 5)
			237	730	µA	$V_{DD}=3 V \pm 10\%$, $f_{CC}=600 kHz$ (Note 5)
Supply current, STOP mode (Note 6)	I_{DD3}		0.1	10	µA	(Notes 4, 6)
			10	40	µA	$V_{DD}=5 V \pm 10\%$ (Note 5)
			7	30	µA	$V_{DD}=3 V$ (Note 5)

Note:

(1) $V_{PRE}=V_{DD}-9 V \pm 1 V$. The circuit in figure 5 is recommended.

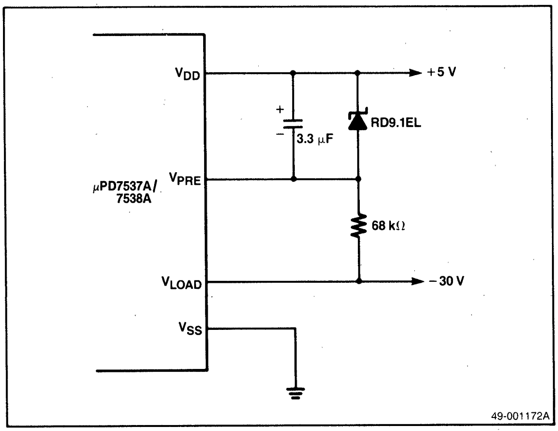
(2) $V_{PRE}=0 V$. $V_{DD}=4.5 V$ to $6.0 V$.

(3) $V_{DD}=4.5 V$ to $6.0 V$.

(4) Without zero-cross detector.

(5) With zero-cross detector.

Figure 1. Recommended Circuit, µPD7537A/7538A



DC Characteristics (cont)

μPD75CG38E

$T_A = -10°C$ to $+70°C$, $V_{DD} = +5\,V \pm 10\%$

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage, low	V_{IL1}	0		0.3 V_{DD}	V	Port 0, RESET
	V_{IL2}	0		0.5	V	CL1
	V_{IL3}	$V_{DD}-35$		0.3 V_{DD}	V	Ports 4, 5, 10, 11
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Port 0, RESET
	V_{IH2}	$V_{DD}-0.5$		V_{DD}	V	CL1
	V_{IH3}	0.7 V_{DD}		V_{DD}	V	Ports 4, 5, 10, 11
Output voltage, low	V_{OL}			0.4	V	$P0_1$, $P0_2$; $I_{OL} = 1.6\,mA$
				0.5	V	$P0_1$, $P0_2$; $I_{OL} = 400\,\mu A$
Output voltage, high	V_{OH}	$V_{DD}-2.0$			V	Ports 2–5, $I_{OH} = -4\,mA$ (Note 1)
		$V_{DD}-2.0$			V	Ports 8–11, $I_{OH} = -10\,mA$ (Note 1)
		$V_{DD}-2.0$			V	Ports 2–5, $I_{OH} = -2\,mA$ (Note 2)
		$V_{DD}-2.0$			V	Ports 8–11, $I_{OH} = -5\,mA$ (Note 2)
		$V_{DD}-1.0$			V	$P0_1$, $P0_2$; $I_{OH} = -1\,mA$ (Note 2)
Input current, low (I_0–I_7)	I_{IL}			−200	μA	$V_{IN} = 0\,V$
Input current, high (MSEL)	I_{IH}			300	μA	$V_{IN} = V_{DD}$
Input leakage current, low	I_{LIL1}			−3	μA	$V_{IN} = 0\,V$; $P0_1$–$P0_3$
	I_{LIL2}			−40	μA	$V_{IN} = 0\,V$; $P0_0$
	I_{LIL3}			−20	μA	$V_{IN} = 0\,V$; CL1
	I_{LIL4}			−10	μA	$V_{IN} = V_{DD}-35\,V$; ports 4, 5, 10, 11
Input leakage current, high	I_{LIH1}			3	μA	$V_{IN} = V_{DD}$; $P0_1$–$P0_3$
	I_{LIH2}			40	μA	$V_{IN} = V_{DD}$; $P0_0$
	I_{LIH3}			20	μA	$V_{IN} = V_{DD}$; CL1
	I_{LIH4}			80	μA	$V_{IN} = V_{DD}$; ports 4, 5, 10, 11
Output leakage current, low	I_{LOL1}			−3	μA	$V_O = 0\,V$; $P0_1$, $P0_2$
	I_{LOL2}			−10	μA	$V_O = V_{DD}-35\,V$; ports 2–5, 8–11

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Output leakage current, high	I_{LOH1}			3	μA	$V_O = V_{DD}$; except ports 4, 5, 10, 11
	I_{LOH2}			80	μA	$V_O = V_{DD}$; ports 4, 5, 10, 11
Supply current, normal operation	I_{DD1}		1.0	3.0	mA	$f_{CC} = 400\,kHz$
Supply current, HALT mode (Note 3)	I_{DD2}		460	1230	μA	$f_{CC} = 400\,kHz$
Supply current, STOP mode (Note 3)	I_{DD3}		10	40	μA	

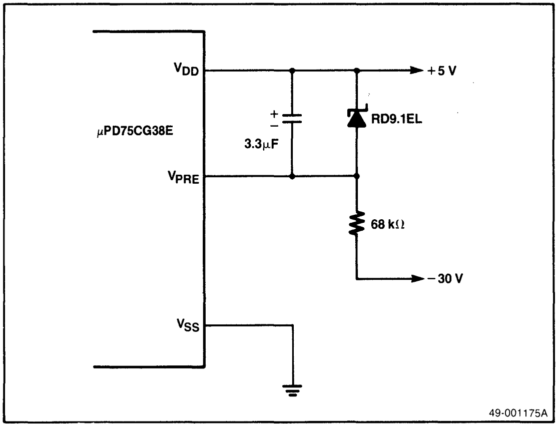
Note:

(1) $V_{PRE} = V_{DD} - 9\,V \pm 1\,V$. The circuit in figure 6 is recommended.

(2) $V_{PRE} = 0\,V$

(3) Ports 4, 5, 10, 11 are output off or low input.

Figure 2. Recommended Circuit, μPD75CG38E

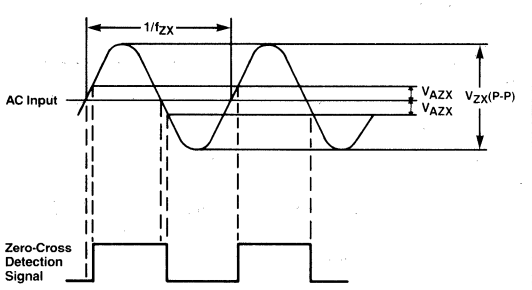


Zero-Cross Detection Characteristics

μPD7537A/38A: $T_A = -10°C$ to $+70°C$, $V_{DD} = 4.5\,V$ to $6.0\,V$
μPD75CG38E: $T_A = -10°C$ to $+70°C$, $V_{DD} = +5\,V \pm 10\%$

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Zero-cross detection input voltage	V_{ZX}(P-P)	1.0		3.0	V_{P-P}	AC coupled, C = 0.1 μF
Zero-cross accuracy	V_{AZX}			±100	mV	50 Hz to 60 Hz sine wave
Zero-cross detection input frequency	f_{ZX}	45		1000	Hz	

Zero-Cross Detection Waveform



Note: In the above waveforms, both 0-to-1 and 1-to-0 transitions of the zero-cross detection signal delay from the low-to-high and high-to-low transitions of the AC input signal, respectively. However, it is possible that the zero-cross detection leads low-to-high and/or high-to-low transition(s) of the AC input signal.

49-001055A

AC Characteristics

μPD7537A/38A

$T_A = -10°C$ to $+70°C$, $V_{DD} = +2.7\,V$ to $6.0\,V$

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Cycle time (Note 1)	t_{CY}	3.3		200	μs	V_{DD} = 4.5 V to 6.0 V
		9.5		200	μs	
$P0_0$ event input frequency	f_{PO}	0		610	kHz	V_{DD} = 4.5 V to 6.0 V
		0		210	kHz	
$P0_0$ input rise time	t_{POR}			0.1	μs	
$P0_0$ input fall time	t_{POF}			0.1	μs	
$P0_0$ input pulse width, low, high	t_{POL}, t_{POH}	2.3			μs	
		0.62			μs	V_{DD} = 4.5 V to 6.0 V
$\overline{SCK}$ cycle time	t_{KCY}	3.0			μs	Input; V_{DD} = 4.5 V to 6.0 V
		3.3			μs	Output; V_{DD} = 4.5 V to 6.0 V
		8.0			μs	Input
		9.5			μs	Output
$\overline{SCK}$ pulse width, low	t_{KL}	4.0			μs	Input
		4.7			μs	Output
$\overline{SCK}$ pulse width, high	t_{KH}	1.3			μs	Input; V_{DD} = 4.5 V to 6.0 V
		1.45			μs	Output; V_{DD} = 4.5 V to 6.0 V
SI set-up time (to rising-edge of $\overline{SCK}$)	t_{SIK}	300			ns	
SI hold time (after rising-edge of $\overline{SCK}$)	t_{KSI}	450			ns	
SO output delay time (after falling-edge of $\overline{SCK}$)	t_{KSO}			850	ns	V_{DD} = 4.5 V to 6.0 V
				1200	ns	
INT0 pulse width, high, low	t_{IOH}, t_{IOL}	10			μs	
RESET pulse width, high, low	t_{RSH}, t_{RSL}	10			μs	

Note:

(1) $t_{CY} = 2/f_{CC}$ or $2/f_C$

AC Characteristics (cont)

μPD75CG38E

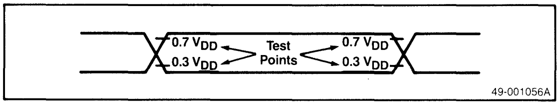
$T_A = -10°C$ to $+70°C$, $V_{DD} = +5\,V \pm 10\%$

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Cycle time (Note 1)	t_{CY}	4.0		200	μs	
$P0_0$ event input frequency	f_{P0}	0		500	kHz	
$P0_0$ input rise time	t_{POR}			0.2	μs	
$P0_0$ input fall time	t_{POF}			0.2	μs	
$P0_0$ input pulse width, high, low	t_{POH}, t_{POL}	0.8			μs	
$\overline{\text{SCK}}$ cycle time	t_{KCY}	3.0			μs	Input
		4.0			μs	Output
$\overline{\text{SCK}}$ pulse width, low	t_{KL}	1.8			μs	Output
$\overline{\text{SCK}}$ pulse width, high	t_{KH}	1.3			μs	Input
SI set-up time (to rising-edge of $\overline{\text{SCK}}$)	t_{SIK}	300			ns	
SI hold time (after rising-edge of $\overline{\text{SCK}}$)	t_{KSI}	450			ns	
SO output delay time (after falling-edge of $\overline{\text{SCK}}$)	t_{KSO}			850	ns	
INT0 pulse width, high, low	t_{IOH}, t_{IOL}	10			μs	
RESET pulse width, high, low	t_{RSH}, t_{RSL}	10			μs	
Data input delay time from address	t_{ACC}			700	ns	
Data input delay time from $\overline{\text{CE}}$	t_{CE}			700	ns	
Input hold time after address	t_{IH}	0			ns	

Note:

(1) $t_{CY} = 2/f_{CC}$ or $2/f_C$

AC Waveform Measurement Points (Except CL1)



Oscillation Characteristics

μPD7537A/38A

$T_A = -10°C$ to $+70°C$, $V_{DD} = 2.7\,V$ to $6.0\,V$

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
System clock oscillation frequency (Note 1)	f_{CC} 7537A	390	400	410	kHz	(Note 2) $V_{DD} = 4.0$ to $6.0\,V$
	f_{CC} 7538A	390	600	610	kHz	$V_{DD} = 4.5$ to $6.0\,V$
Oscillation stable time (Note 1)	t_{OS}	20			ms	(Note 3)
System clock CL1 input frequency (Note 4)	f_C	10		610	kHz	$V_{DD} = 4.5\,V$ to $6.0\,V$
		10		210	kHz	
CL1 input rise time	t_{CR}			0.1	μs	
CL1 input fall time	t_{CF}			0.1	μs	
CL1 input pulse width, low	t_{CL}	2.0		50	μs	
CL1 input pulse width, high	t_{CH}	0.7		50	μs	$V_{DD} = 4.5\,V$ to $6.0\,V$

Note:

(1) Ceramic resonator: CSB400P (MURATA) or KBR-400B (KYO-CERA) is recommended (see figure 3).

(2) Oscillation is only guaranteed at $3\,V \leqslant V_{DD} \leqslant 4.5\,V$.

(3) After V_{DD} reaches 4.5 V.

(4) External clock (see figure 4).

μPD75CG38E

$T_A = -10°C$ to $+70°C$, $V_{DD} = 5\,V \pm 10\%$

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
System clock oscillation frequency (Note 1)	f_{CC}	390	400	410	kHz	
Oscillation stable time (Note 1)	t_{OS}	20			ms	After V_{DD} reaches 4.5 V
System clock CL1 input frequency (Note 2)	f_C	10		500	kHz	
CL1 input rise time	t_{CR}			0.1	μs	
CL1 input fall time	t_{CF}			0.1	μs	
CL1 input pulse width high, low	t_{CH}, t_{CL}	0.8		50	μs	

Note:

(1) Ceramic resonator: CSB400P (MURATA) is recommended; C = 300 pF (see figure 3).

(2) External clock (see figure 4).

Figure 3. ***Recommended Circuit, μPD7537A/7538A***

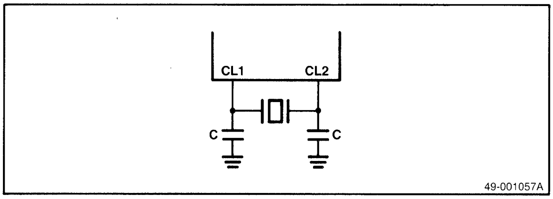
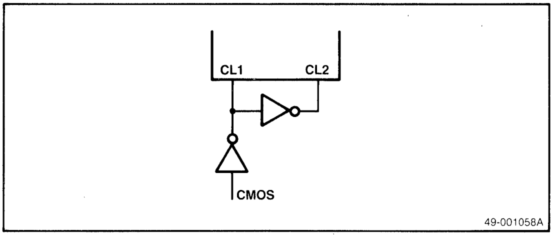


Figure 4. ***Recommended Circuit, μPD75CG38E***



Stop Mode Low Voltage Data Retention Characteristics

μPD7537A/38A

$T_A = -10°C$ to $+70°C$, $V_{DD} = 2.7\,V$ to $6.0\,V$

		Limits				Test
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
Data retention supply voltage	V_{DDDR}	2.0		6.0	V	
Data retention supply current	I_{DDDR}		0.1	10	μA	$V_{DDDR} = 2\,V$ (Note 1)
			7	30	μA	$V_{DDDR} = 2\,V$ (Note 2)
RESET set-up time	t_{SRS}	0			μs	
Oscillation stable time	t_{OS}	20			ms	After V_{DD} reaches 4.5 V

μPD75CG38E

$T_A = -10°C$ to $+70°C$, $V_{DD} = 5\,V \pm 10\%$

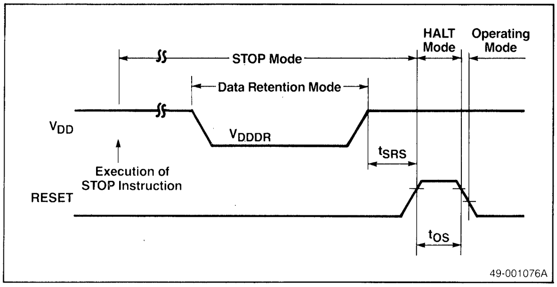
		Limits				Test
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
Data retention supply voltage	V_{DDDR}	2.0		5.5	V	
Data retention supply current	I_{DDDR}		7	30	μA	$V_{DDDR} = 2\,V$
RESET set-up time	t_{SRS}	0			μs	
Oscillation stable time	t_{OS}	20			ms	After V_{DD} reaches 4.5 V

Note:

(1) Without zero-cross detector.

(2) With zero-cross detector.

Data Retention Mode Timing



μPD75CG38E EPROM Interface

A 4-Kbyte EPROM (2732) plugs into socket pins on top of the μPD75CG38E. A high input to MSEL selects the μPD7537A mode and fixes the A_{11} output high level in order to access the upper 2-Kbytes of the 4-Kbyte EPROM. When MSEL is open, μPD7538A mode is selected. All EPROM addresses can be accessed because A_{11} functions as the MSB of the address. Figure 5 shows the address control unit. Figures 6 and 7 show the μPD75CG38E connected with the 2732.

Figure 8 shows the EPROM read timing. Data is read into the instruction buffer at the end of the T4 state. The chip enable ($\overline{CE}$) signal is made active during 2 states (T3, T4) in order to decrease the power consumption of the EPROM.

Figure 5. ***Address Control Unit***

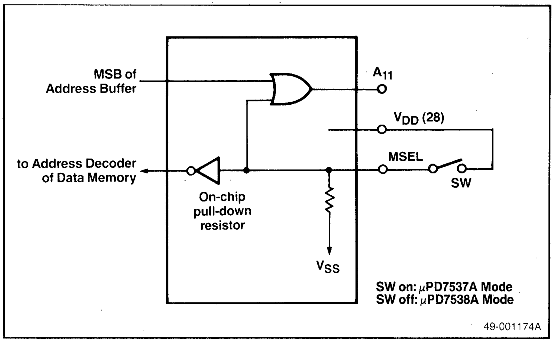


Figure 6. Connection with the 2732 (μPD7537A Mode)

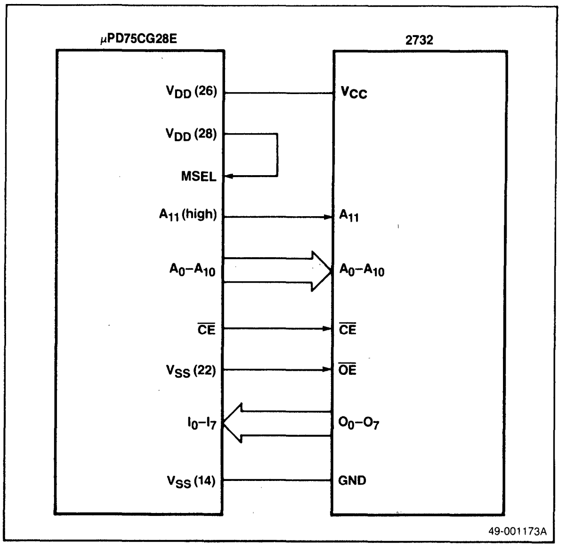


Figure 7. Connection with the 2732 (μPD7538A Mode)

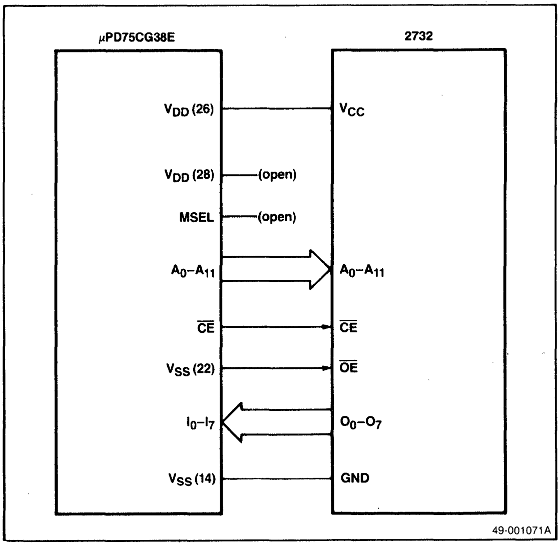
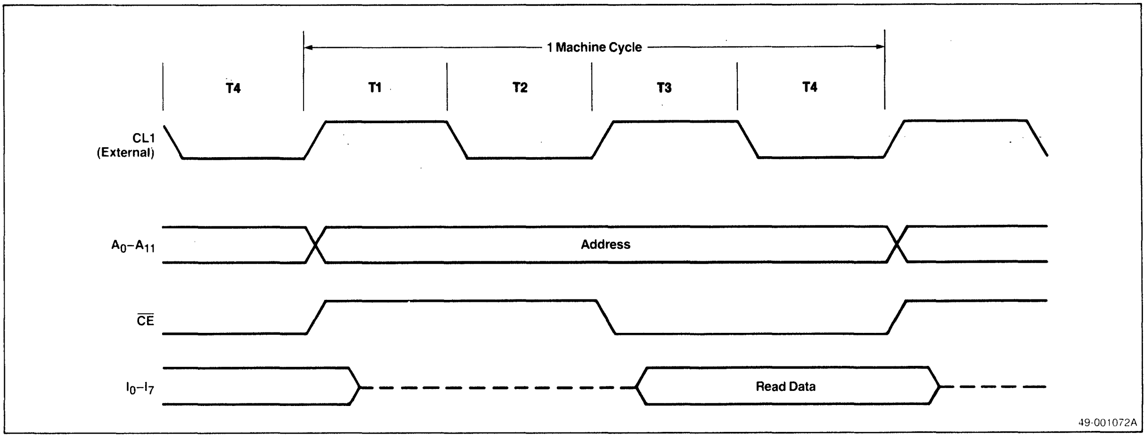
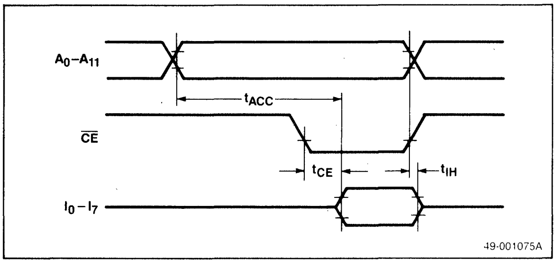


Figure 8. EPROM Read Timing

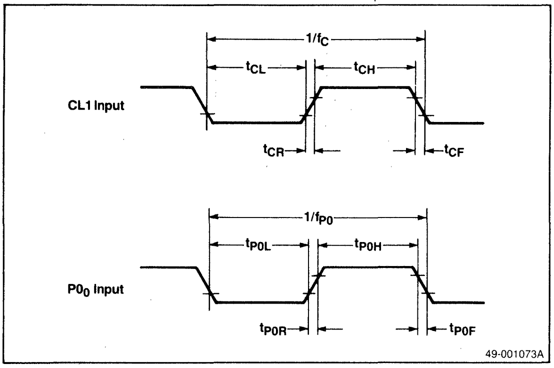


Timing Waveforms

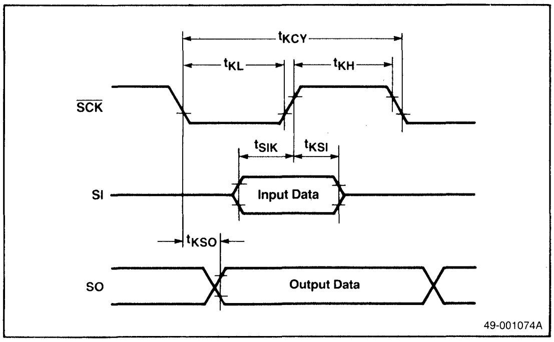
EPROM (μPD75CG38E only)



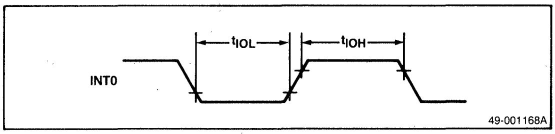
Clock



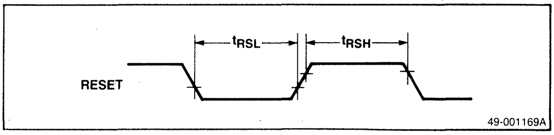
Serial Interface



Interrupt Input



Reset Input



Differences Among the μPD7537A/38A/CG38E

	μPD75CG38E	μPD7537A	μPD7538A
Program memory	4 Kbyte EPROM (2732) connectable on top	On-chip 2 Kbyte ROM	On-chip 4 Kbyte ROM
Data memory (RAM)	160 × 4	128 × 4	160 × 4
High-voltage output lines	All open-drain outputs	On-chip load resistor or open drain output (bit by bit, mask optional)	
V_{LOAD} pin	No	Yes	
Zero-cross detection	Yes	Mask optional	
Package	42-pin ceramic piggyback DIP bottom pin compatible with μPD7537A/38A	42-pin plastic DIP 42-pin plastic shrink DIP	
Power supply	5 V	2.7 V to 6.0 V	

NEC
NEC Electronics Inc.

µPD7554/54A/64/64A 4-Bit, Single-Chip CMOS Microcomputers With Serial I/O

Description

The µPD7554/54A and µPD7564/64A are low-end versions of µPD7500 series products. These microcomputers incorporate a serial interface and are useful as slave CPUs to high-end µPD7500 series or 8-bit µCOM-87 series products.

The µPD7554/54A/64/64A has output ports that can directly drive triacs and LEDs. Also, various mask-optional I/O circuits can be configured for a wide selection of outputs allowing a reduction of external circuitry in your design.

The µPD7554/54A and µPD7564/64A differ only in their clock circuitry. The µPD7554/54A uses an external resistor with an internal capacitor for an RC oscillator clock, where the µPD7554/54A uses an external ceramic oscillator as a clock. These microcomputers are ideally suited to control devices such as plain paper copiers (PPCs), printers, VCRs, and audio equipment.

Features

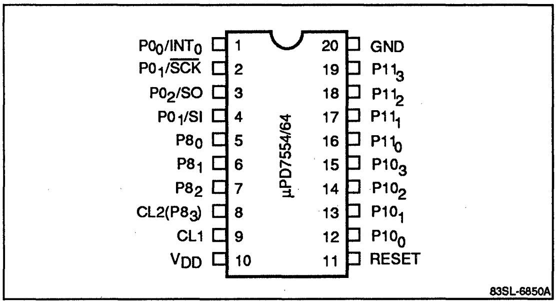
- 47 instructions (subset of µPD7500 set B)
- Instruction cycle:
  — External clock: 2.86 µs/700 kHz, 5 V
  — RC oscillator (µPD7554/54A): 4 µs/500 kHz, 5 V
  — Ceramic oscillator (µPD7564/64A): 3 µs/660 kHz, 5 V
- Program memory (ROM) of 1024 x 8-bits
- Data memory (RAM) of 64 x 4-bits
- 8-bit timer/event counter
- 8-bit serial interface
- I/O lines: 16-µPD7554/54A; 15-µPD7564/64A
- Data memory retention at low supply voltage
- CMOS technology
- Low-power consumption
- Single power supply
  — 2.5 to 6.0 V (µPD7554/54A)
  — 2.7 to 6.0 V (µPD7564/64A)
  — 2.0 to 6.0 V (µPD7554A)

Ordering Information

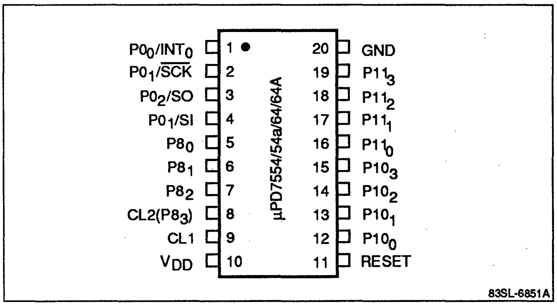
Part Number	Package Type
µPD7554CS	20-pin plastic shrink DIP
µPD7554ACS	
µPD7564CS	
µPD7564ACS	
µPD7554G	20-pin plastic SOP
µPD7554AG	
µPD7564G	
µPD7564AG	

Pin Configurations

20-Pin Plastic Shrink DIP



20-Pin Plastic SOP



Pin Identification

Symbol	Function
$P0_0$/INT0	4-bit input port 0/count clock input/serial interface
$P0_1/\overline{SCK}$	
$P0_2$/SO	
$P0_3$/SI	
$P8_0$-$P8_2$ $P8_3/CL_2$	4-bit output port 8 Connection for ceramic resonator or RC
CL1	Connection for ceramic resonator or RC
V_{DD}	+5 V power supply
RESET	Reset input pin
$P10_1$-$P10_3$	4-bit I/O port 10
$P11_0$-$P11_3$	4-bit I/O port 11
V_{SS}	Ground

PIN FUNCTIONS

$P0_0$/INT0, $P0_1/\overline{SCK}$ $P0_2$/SO, $P0_3$/SI (Port 0/Count clock input/Serial interface)

4-bit input port 0/count clock input/serial I/O interface. This port can be configured as a 4-bit parallel input port or as the 8-bit serial I/O interface, under control of the serial mode select register. The serial input SI (active high), serial output SO (active low), as the serial clock $\overline{SCK}$ (active low—synchronizes data transfer) comprise the 8-bit serial I/O interface. If $P0_0$/INT0 is unused, connect it to ground. If any of $P0_1$-$P0_3$ are unused, connect them to ground or V_{DD}. The port is in the input state at reset.

$P8_0$-$P8_2$, $P8_3$-CL2 (Port 8/Clock input 2)

4-bit output port 8. This port can sink 15 mA and interface 12 V. On the μPD7554/54A, the port function of $P8_3$/CL2 is specified by mask option. $P8_3$ is a normal output port on the μPD7564/64A. On the μPD7554/54A, CL2 is one of the pins to which a resistor for RC oscillation is connected. On the μPD7564/64A, CL2 is one of the pins to which a ceramic resonator is connected. If any of $P8_0$-$P8_2$ pins are unused, leave them open. The port is in the high impedance state at reset.

CL1 (Clock input 1)

On the μPD7554/54A, CL1 is one of the two pins to which a resistor for RC oscillation is connected. On the μPD7564/64A, CL1 is one of the two pins to which a ceramic resonator is connected.

V_{DD} (Power supply)

Positive power supply.

RESET (Reset)

System reset input pin (active high). This pin can be internally connected to a pull-down resistor if specified by mask option.

$P10_0$-$P10_3$ (Port 10)

4-bit I/O port. This port can sink 10 mA and interface 12 V. If any of these pins are unused, connect them to ground or V_{DD} in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset.

$P11_0$-$P11_3$ (Port 11)

4-bit I/O port. This port can sink 10 mA and interface 12 V. If any of these pins are unused, connect them to ground or V_{DD} in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset.

V_{SS} (Ground)

Ground.

Pin Mask Options

Table 1 shows the mask options for all the port pins and the RESET pin. You may select these options in bit units.

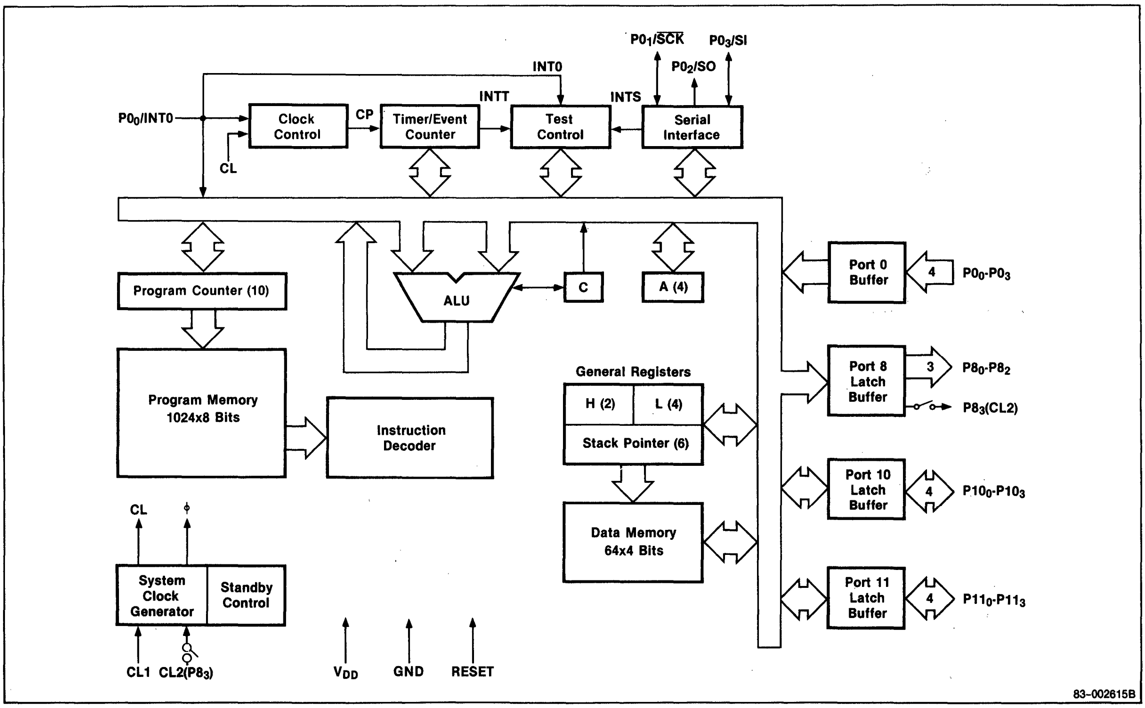
Table 1. Pin Mask Options

Pin	Options
$P0_0$-$P0_3$	1 No connection to internal resistor 2 Connected to internal pull-up resistor 3 Connected to internal pull-down resistor
$P8_0$-$P8_2$	1 CMOS (push-pull) output 2 N-channel, open-drain output
$P8_3$/CL2 (1)	1 Use as $P8_3$ 2 Use as CL2
Used as $P8_3$	1 CMOS (push-pull) output 2 N-channel, open-drain output
$P10_0$-$P10_3$ $P11_0$-$P11_3$	1 N-channel, open drain input/output 2 CMOS (push-pull) input/output 3 N-channel, open-drain input/output with internal pull-up resistor
RESET	1 Connected to internal pull-down resistor 2 Not connected to internal pull-down resistor

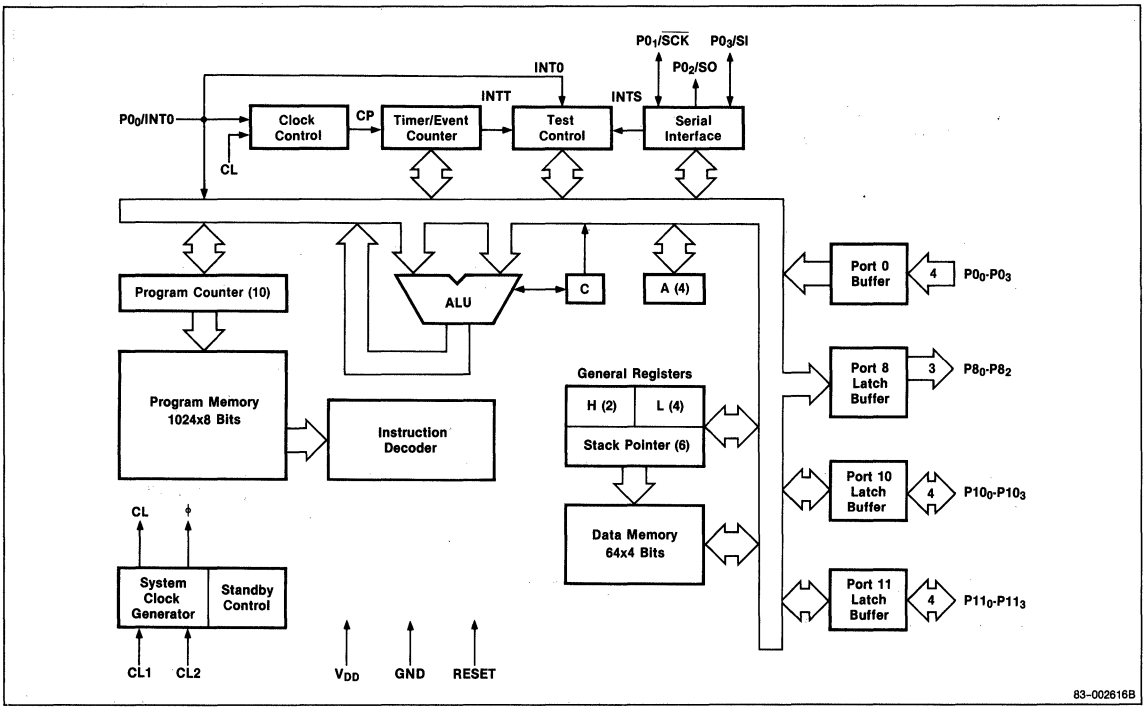
Notes:

(1) μPD7554/54A only.

μPD7554/54A Block Diagram



μPD7564/64A Block Diagram

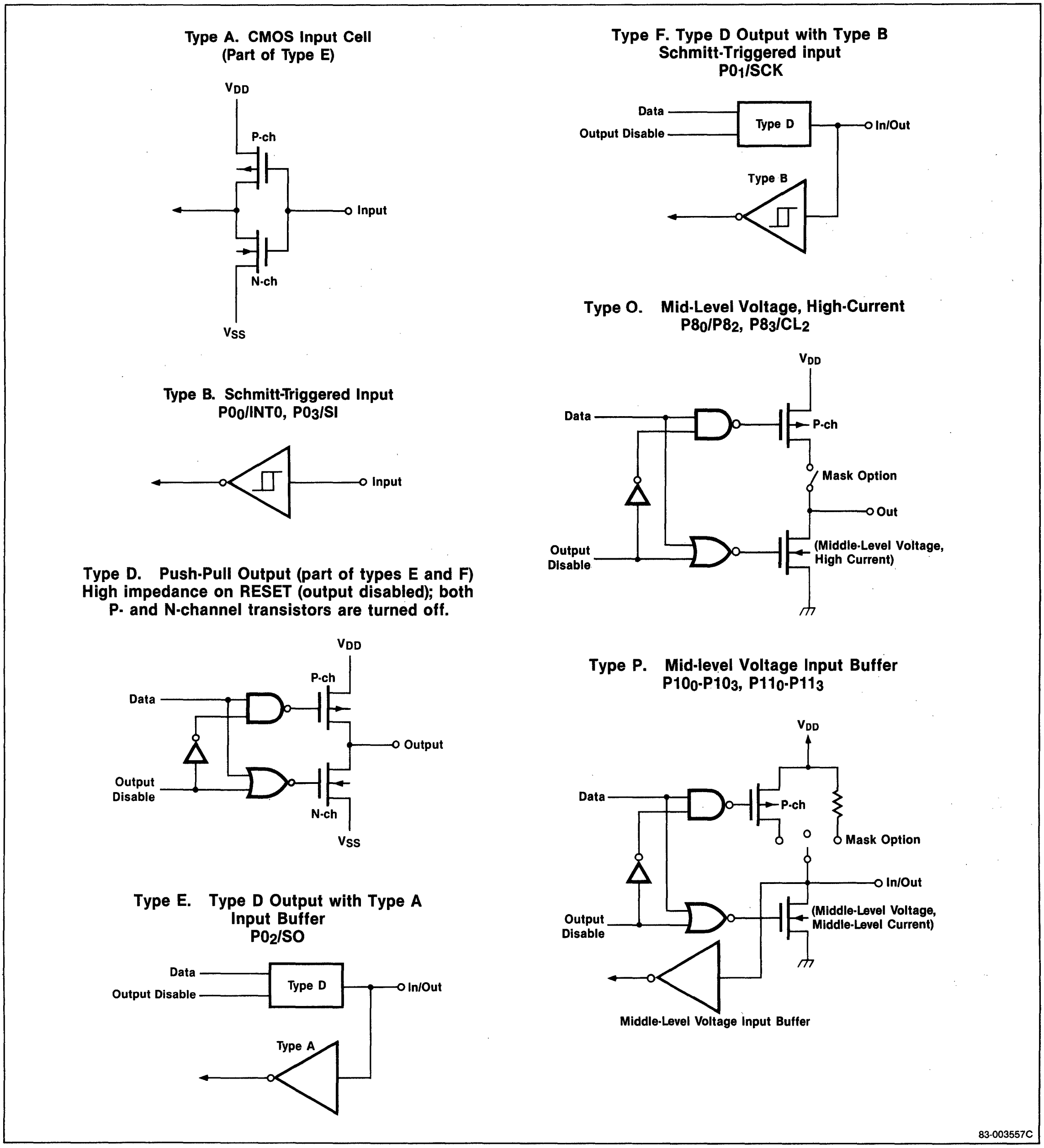


FUNCTIONAL DESCRIPTION

I/O Ports

Figure 1 shows the internal circuits at I/O ports P0, P8, P10, and P11.

Figure 1. Interface at I/O Ports



Type A. CMOS Input Cell (Part of Type E)

Type B. Schmitt-Triggered Input $P0_0$/INT0, $P0_3$/SI

Type D. Push-Pull Output (part of types E and F) High impedance on RESET (output disabled); both P- and N-channel transistors are turned off.

Type E. Type D Output with Type A Input Buffer $P0_2$/SO

Type F. Type D Output with Type B Schmitt-Triggered input $P0_1$/SCK

Type O. Mid-Level Voltage, High-Current $P8_0$/$P8_2$, $P8_3$/CL_2

Type P. Mid-level Voltage Input Buffer $P10_0$-$P10_3$, $P11_0$-$P11_3$

VDD
P-ch
Input
N-ch
VSS
Data
Output Disable
Output
Type D
In/Out
Type A
Type B
Mask Option
Out
(Middle-Level Voltage, High Current)
(Middle-Level Voltage, Middle-Level Current)
Middle-Level Voltage Input Buffer

83-003557C

Program Memory

The μPD7554/54A/64/64A has a mask-programmable ROM with a capacity of 1024 words by 8 bits for program storage. It is addressed by the program counter. The reset start address is 000H. Figure 2 shows the program memory map.

General-Purpose Registers

Two registers, H(2-bit) and L(4-bit), are provided as general-purpose registers. Each register can be individually manipulated. The two registers also form pair register HL; H being the high register and L being the low one. The HL register is a data pointer to address data memory. Figure 3 shows the configuration of the general purpose registers.

The L register also specifies an I/O port or mode register when an I/O instruction (IPL or OPL) is executed. It also specifies the bits of a port when the SPBL or RPBL instruction is executed.

Data Memory

The data memory is static RAM with a capacity of 64 words by 4 bits. Part of this memory is used as the stack area. The data memory is also used in 8-bit data processing when paired with the accumulator. Figure 4 shows the data memory map.

Data memory can be addressed directly, with the immediate data from an instruction; indirectly, with the contents of HL (including auto-increment and auto-decrement); and indirectly by the contents of the stack pointer.

You may use any area of the data memory as the stack. The boundary of the stack is determined by how the TAMSP instruction initializes the stack pointer. Once the boundary is set, a call or return instruction automatically accesses the stack.

When a call instruction is executed, the contents of the program counter and the program status word (PSW) are stored to the stack in the sequence shown in figure 5.

When a return instruction is executed, the contents of the program counter are automatically restored, but the PSW is not. The contents of data memory can be retained with a low supply voltage during STOP mode.

Accumulator

The accumulator is a 4-bit register used in arithmetic operations. The accumulator can process 8-bit data with paired data addressed by HL. Figure 6 shows the configuration of the accumulator.

Arithmetic Logic Unit

The arithmetic logic unit (ALU) is a 4-bit arithmetic circuit that performs operations such as binary addition, logical operation, increment, decrement, comparison, and bit processing.

Program Status Word

The program status word (PSW) consists of two skip flags (SK0 and SK1), a carry flag (C), and bit 1, which is always zero. Figure 7 shows the configuration of the PSW.

The contents of the PSW are stored to the stack when a call instruction is executed, but are not restored from the stack by the return instruction.

The skip flags retain the following skip conditions: string effect by LAI or LHLI instruction, and skip condition satisfied by an instruction other than a string-effect instruction. The skip flag is set or reset according to the instruction executed.

The carry flag is set to 1 if an addition instruction (ACSC) generates a carry from bit 3 of the ALU. If no carry is generated, the flag is reset to zero. The SC instruction sets the carry flag and the RC instruction resets it.

When a RESET is input, the SK1 and SK0 flags are cleared to zero and the contents of the carry flag are undefined.

Figure 2. Program Memory Map

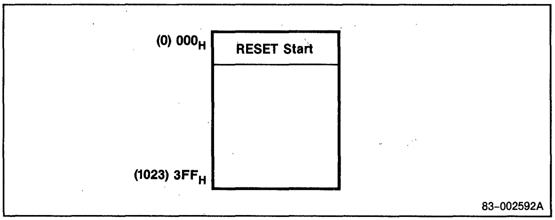


Figure 3. Configuration of General Purpose Registers

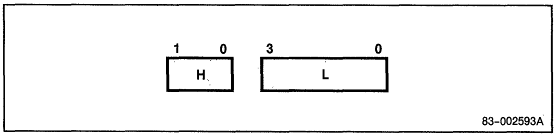


Figure 4. Data Memory Map

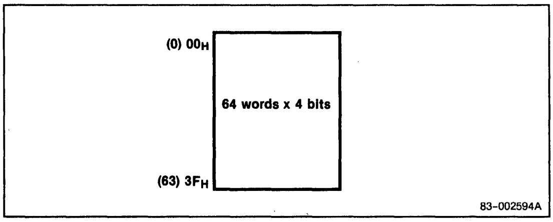


Figure 5. Call Instruction Storage to Stack

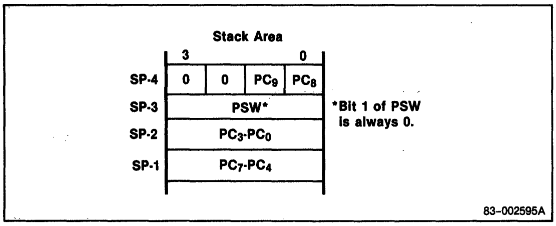


Figure 6. Configuration of the Accumulator

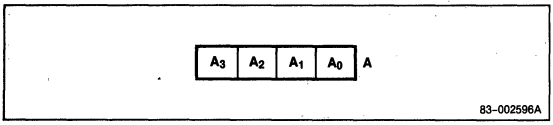
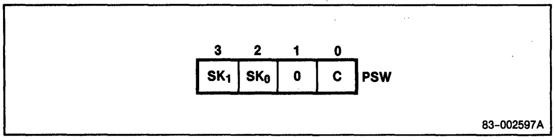


Figure 7. Configuration of the Program Status Word



System Clock Generator

The system clock generator consists of a ceramic oscillator, a 1/2 frequency divider, standby modes (STOP/HALT), and control circuit. Figure 8 is a circuit diagram of the system clock generator.

In the μPD7554/54A, the RC oscillator operates with a single external resistor connected across CL1 and CL2 (the capacitor C is incorporated). When the RC oscillator is not used, external clock pulses can be input by the CL1 pin. In this case, the RC oscillator functions as an inverting buffer. The output from the RC oscillator serves as the system clock (CL) which is then divided by two and used as the CPU clock (ϕ).

The standby mode control circuit is made up of a STOP flip-flop and a HALT flip-flop. The STOP instruction sets the STOP flip-flop and stops the system clock supply. This flip-flop also stops the RC oscillator. The STOP flip-flop is reset by the standby release signal that becomes active when one of the test requests flags is set or at the falling edge of the RESET signal. When the STOP flip-flop is reset, the RC oscillator resumes operation and supplies the system clock.

The HALT and STOP instructions and RESET HIGH set the HALT flip-flop which disables signals from going to the 1/2 frequency divider that generates the CPU clock. Only the CPU clock stops in HALT mode. The HALT flip-flop is reset by the same conditions as the STOP flip-flop.

Figure 9 shows the system clock generator circuit for the μPD7564/64A.

On the μPD7564/64A, the ceramic oscillator operates with a ceramic resonator connected across CL1 and CL2. The output from the ceramic oscillator is used as the system clock (CL); it is divided by two to produce the CPU clock (ϕ).

The standby mode control circuit is made up of a STOP flip-flop and a HALT flip-flop. The STOP instruction sets the STOP flip-flop and stops the ceramic oscillation, thus stopping the supply for all clocks. The STOP flip-flop is reset by the RESET signal (high level) and restarts ceramic oscillation. The supply of each clock resumes when RESET goes low.

The HALT instruction sets the HALT flip-flop which disables signals from going to the 1/2 frequency divider that generates the CPU clock. Only the CPU clock stops in HALT mode. The HALT flip-flop is reset by the HALT RELEASE signal (activated by setting at least one test request flag) or the falling edge of RESET, resuming supply of the CPU clock.

The HALT flip-flop is also set when RESET is active (high level). At power on reset operation, the rising edge of RESET starts ceramic oscillation; however, some time is required to achieve stable oscillation. To prevent the unstable clock from operating the CPU, the HALT flip-flop is set and the CPU clock is stopped while RESET is high. Accordingly, the high-level width of RESET must be more than the required stable time for the ceramic resonator.

Figure 8. System Clock Generator for µPD7554/54A

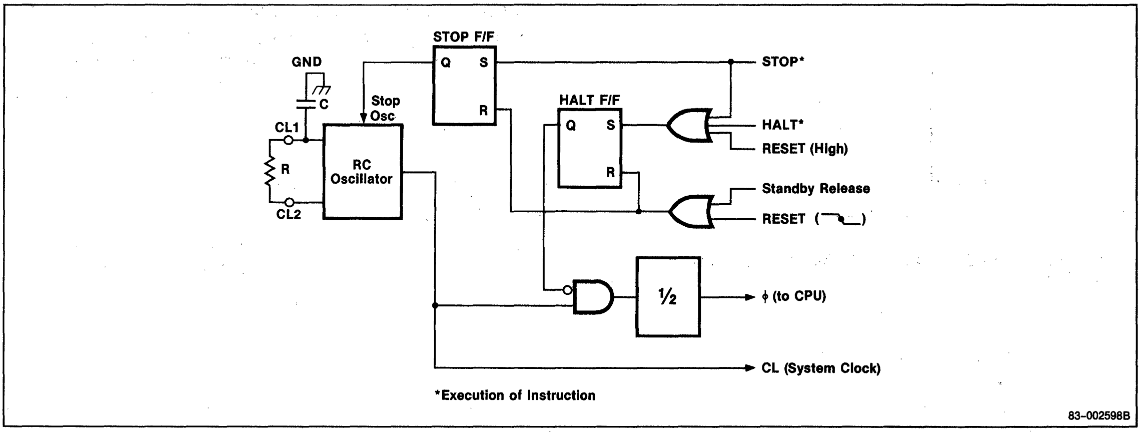
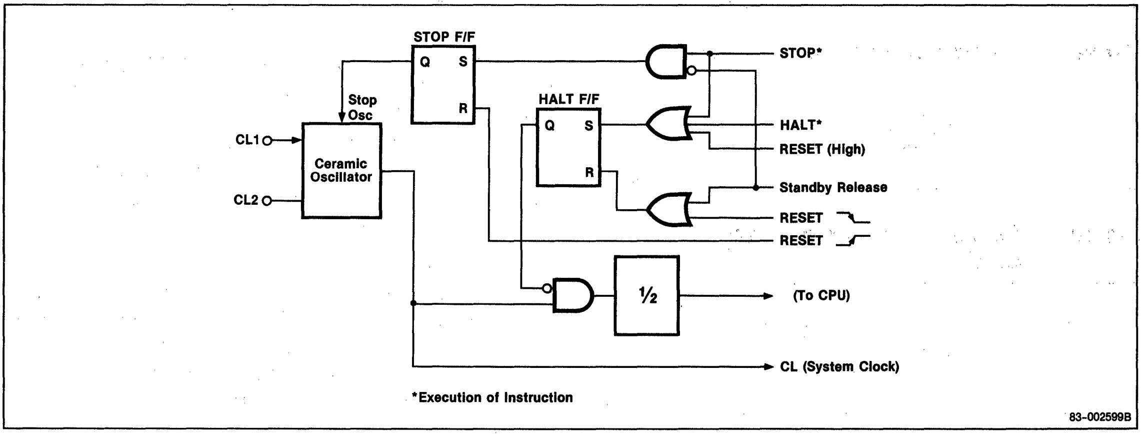


Figure 9. System Clock Generator for µPD7564/64A



Clock Control Circuit

The clock control circuit consists of a 2-bit clock mode register (bits CM1 and CM2), prescalers 1, 2, and 3, and a multiplexer. It takes the output of the system clock generator (CL) and event pulses ($P0_0$). It also selects the clock source and prescaler according to the setting in the clock mode register and supplies the timer/event counter with count pulses. Figure 10 shows the clock control circuit.

Table 2 lists the codes set in the clock mode register by the OPL instruction to specify the count pulse frequency.

When you set the clock mode register with the OPL instruction, clear bit 0 of the accumulator (corresponding to bit CM0 of the EVAKIT-7500 or µPD7500H during emulation).

Figure 10. Clock Control Circuit

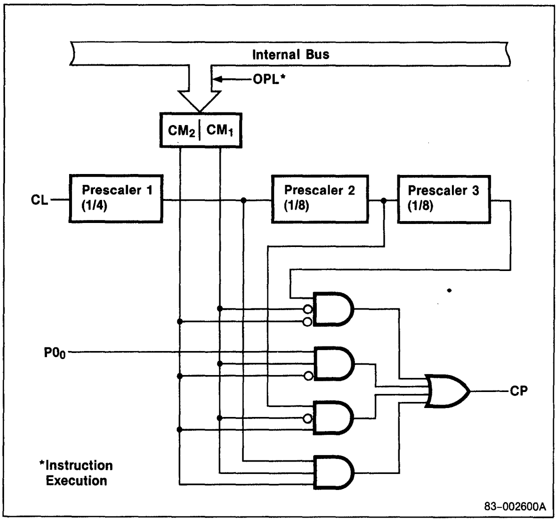


Table 2. Selecting the Count Pulse Frequency

CM2	CM1	Frequency Selected
0	0	CL/256
0	1	$P0_0$
1	0	CL/32
1	1	CL/4

Timer/Event Counter

The timer/event counter is a binary 8-bit up-counter which is incremented each time a count pulse is input. The TIMER instruction or a RESET signal clears it to 00H. When an overflow occurs, the counter is reset from FFH to 00H. Figure 11 shows the inputs and outputs of the counter.

Serial Interface

The serial interface consist of an 8-bit shift register, a 3-bit shift mode register, and a 3-bit counter. This interface inputs and outputs serial data. Figure 12 is a block diagram of the interface.

Test Control Circuit

The µPD7564/64A has three test sources, as shown in table 3.

The test control circuit consists of two test request flags (INTT RQF and INT0/S RQF) set by the three test sources, and a test request flag control circuit that checks the contents of each test request flag by executing an SKI instruction and resetting the flags.

Test sources INT0 and INTS share the request flag INT0/S RQF. Bit 3 of the shift mode register (SM_3) determines which source is selected. A zero in SM_3 selects INTS and a one selects INT0.

Figure 11. Timer/Event Counter

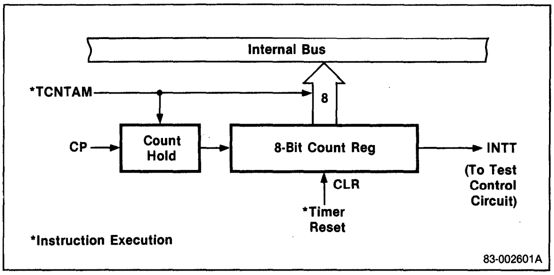


Table 3. μPD7564/64A Test Sources

Source	Function	Location	Request Flag
INTT	Overflow in timer/ event counter	Internal	INTT RQF
INTO	Test request signal from $P0_0$ pin	External	INT0/S RQF
INTS	Transfer complete signal from serial interface	Internal	INT0/S RQF

The request flag INTT RQF is set when a timer overflow occurs in the timer event counter. The SKI or TIMER instruction resets it.

When SM_3 is zero, request flag INT0/S RQF is set when the INTS signals is generated, indicating the end of an 8-bit serial data transfer. The SKI or SIO instruction resets the flag.

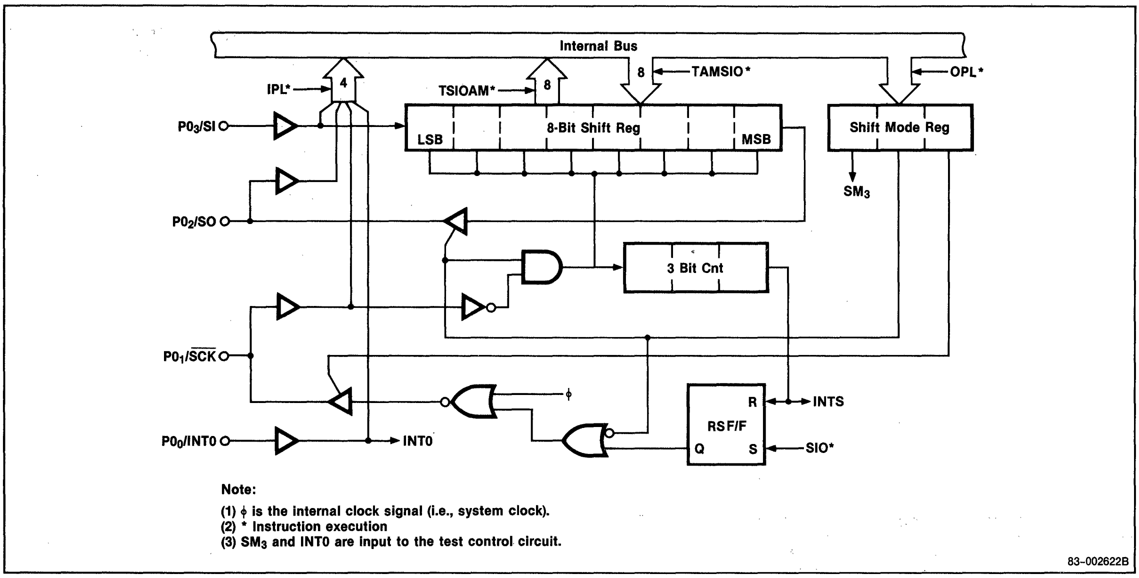
When SM_3 is one, request flag INT0/S RQF is set at the rising edge of the signal input to the $P0_0$/INT0 pin. The SKI instruction resets the flag.

The logical sum of the outputs from the test request flags releases standby mode (STOP[1] or HALT mode). The mode is released when one or both flags are set. Both flags and SM_3 are reset when the RESET signal is input. After reset, source INTS is selected and signal input to the INTO pin is inhibited as the initial condition.

Figure 13 is a block diagram of the test control circuit.

Note: (1) Only μPD7554/54A.

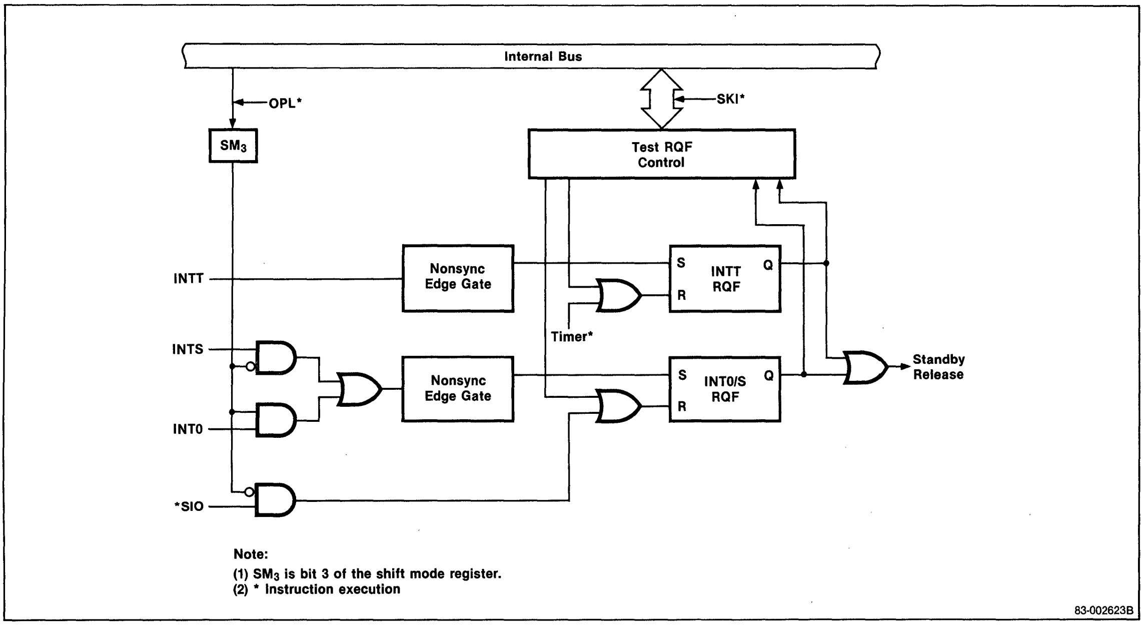
Figure 12. Serial Interface Block Diagram



Note:

(1) ϕ is the internal clock signal (i.e., system clock).
(2) * Instruction execution
(3) SM_3 and INTO are input to the test control circuit.

Figure 13. Test Control Circuit Block Diagram



Note:
(1) SM3 is bit 3 of the shift mode register.
(2) * Instruction execution

Standby Modes

The μPD7554/54A/64/64A has two standby modes to reduce power consumption while the program is in the wait state. The STOP and HALT instructions set these modes.

When the program enters a standby mode, program execution stops and the contents of all registers and data memory immediately before the program entered standby mode are retained. The timer and serial interface can operate.

The RESET signal and STANDBY Release signal(1) release STOP mode. HALT mode is released when one or both of the test request flags are set, or when the RESET signal is input. The program cannot enter a standby mode when a test request is being set, even if the STOP or HALT command is executed.

If there is some uncertainty about the state of the test request flags, execute the SKI instruction to reset them so the program can enter standby mode.

Table 4 compares STOP and HALT modes. The main difference is that STOP mode stops the system clock and HALT does not. Ceramic oscillation stops during STOP mode. The power consumed by the ceramic oscillator is the difference between the two modes. In STOP mode, data memory can be retained with a lower supply voltage.

Note: (1) Standby release signal for μPD7554/54A only.

Table 4. STOP and HALT Modes

Mode	CL	φ	PO_0	CPU	Timer	Released by
STOP	x	x	o	x	Δ	RESET input
HALT	o	x	o	x	o	INTT RQF INT0/S RQF RESET input

Notes:

(1) o: operates. x: stops.
Δ: operates depending on clock source. μPD7554/54A; if external clock is used, STOP instruction will not stop CL. In this case STOP mode acts as HALT mode.

Power-on Reset Circuit

Figure 14 shows a circuit example of the power-on reset circuit using a resistor and a capacitor. This is the simplest reset control circuit. Figure 15 shows the circuit with a pull-down resistor internally connected to RESET as a mask option.

μPD7554/54A/64/64A Applications

Figures 16 and 17 show examples of application circuits for the μPD7554/54A/64/64A.

Table 5 compares the features of the low-end products of the 7500 series devices.

Figure 14. Power-on Reset Circuit

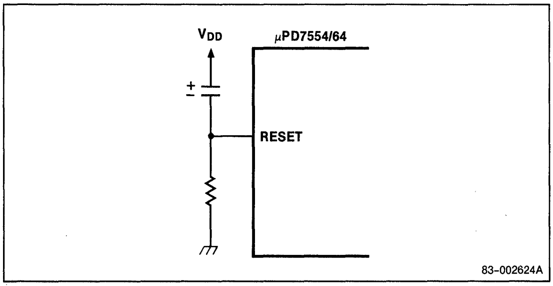


Figure 15. Power-on Reset Circuit with Pull-down Resistor

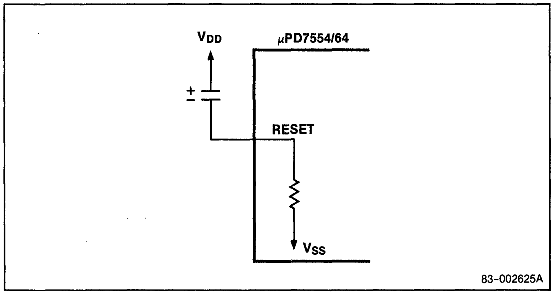


Table 5. Product Comparison

Item		μPD7554/54A	μPD7564/64A	μPD7556/56A	μPD7566/66A
Instruction cycle/system clock (5 V)	RC	4 μs/ 500 kHz		4 μs/ 500 kHz	
	External	2.86 μs/ 700 kHz		2.86 μs/ 700 kHz	
	Ceramic		3 μs/ 660 kHz		3 μs/ 660 kHz
Instruction set		47	47	45	45
ROM		1024 x 8	1024 x 8	1024 x 8	1024 x 8
RAM		64 x 4	64 x 4	64 x 4	64 x 4
I/O port total		16 (max)	15	20 (max)	19
Port 0		$P0_0$-$P0_3$	$P0_0$-$P0_3$	$P0_0$-$P0_1$	$P0_0$-$P0_1$
Port 1				$P1_0$-$P1_3$	$P0_1$-$P0_3$
Port 8		$P8_0$-$P8_2$ $P8_3$/CL2	$P8_0$-$P8_2$	$P8_0$-$P8_2$ $P8_3$/CL2	$P8_0$-$P8_2$
Port 9				$P9_0$-$P9_1$	$P9_0$-$P9_1$
Port 10		$P10_0$-$P10_3$	$P10_0$-$P10_3$	$P10_0$-$P10_3$	$P10_0$-$P10_3$
Port 11		$P11_0$-$P11_3$	$P11_0$-$P11_3$	$P11_0$-$P11_3$	$P11_0$-$P11_3$
Timer/Event counter		8-bit	8-bit	8-bit	8-bit
Serial interface		8-bit	8-bit		
Comparator				4-channel	4-channel
Process		CMOS	CMOS	CMOS	CMOS
Package		20-pin plastic SOP	20-pin plastic SOP	24-pin plastic SOP	24-pin plastic SOP
		20-pin shrink DIP	20-pin shrink DIP	24-pin shrink DIP	24-pin shrink DIP

Figure 16. Tape Counter Circuit

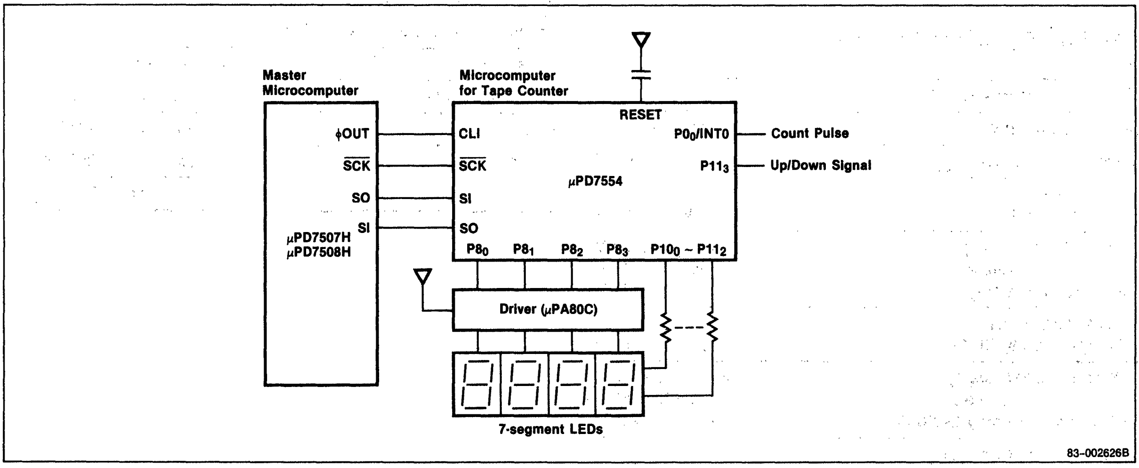
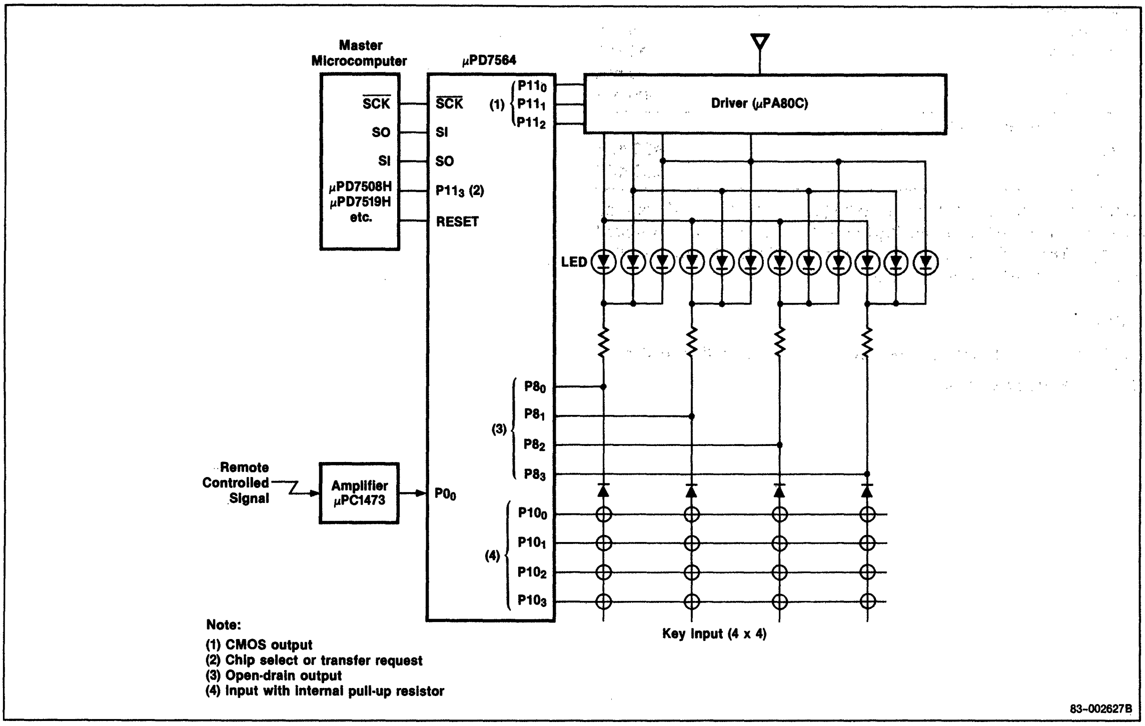


Figure 17. Remote-controlled Data Reception, Key Input and LED Display



Note:
(1) CMOS output
(2) Chip select or transfer request
(3) Open-drain output
(4) Input with internal pull-up resistor

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C
Power supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_I	
Except ports 10, 11	−0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 1)	−0.3 to V_{DD} +0.3 V
(Note 2)	−0.3 to +13 V
μPD7554A/64A only (Note 2)	−0.3 to +11 V
Output voltage, V_O	
Except ports 8, 10, 11	−0.3 to V_{DD} +0.3 V
Ports 8, 10, 11(Note 1)	−0.3 to V_{DD} +0.3 V
(Note 2)	−0.3 to +13 V
μPD7554A/64A only (Note 2)	−0.3 V to +11 V
Output current, high I_{OH}	
One port	−5 mA
All output ports, total	−15 mA
Output current, low I_{OL}	
PO_1, PO_2	5 mA
Ports 9-11	15 mA
Port 8	30 mA
All ports, total	100 mA
Power dissipation, P_D (T_A = +70°C)	
Shrink DIP	480 mW
SOP	250 mW

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Notes:

(1) CMOS I/O or N-channel open drain + internal pull up resistor.

(2) N-channel open drain I/O.

Capacitance

T_A = 25°C, V_{DD} = GND = 0 V; f = 1 MHz
Unmeasured pins returned to GND

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I			15	pF	PO_0, PO_3
Output capacitance	C_O			35	pF	Port 8
I/O capacitance	$C_{I/O}$			35	pF	Ports 10, 11
				15	pF	PO_1, PO_2

DC Characteristics 1; V_{DD} = 2.5 to 3.3 V; μPD7554/54A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.8 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.3		V_{DD}	V	
Input high voltage ports 10, 11	V_{IH3}	0.8 V_{DD}		12 (Note 1); 9 (Note 2)	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1	V_{IL1}	0		0.2 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.3	V	
Input leakage current except CL1	I_{LI1}	−3		3	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11	I_{LI3}			10 (Note 1)	μA	V_I = 12 V
				10 (Note 2)	μA	V_I = 9 V
Output voltage high P0$_1$, P0$_2$, ports 8-11	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −80 μA
Output voltage low P0$_1$, P0$_2$, ports 10, 11	V_{OL}			0.5	V	P0$_1$, P0$_2$: I_{OL} = 350 μA; Ports 10, 11: I_{OL} = 350 μA
Output voltage low port 8	V_{OL}			0.5	V	I_{OL} = 500 μA
Output leakage current	I_{LO1}	−3		3	μA	0 V ≤ V_O ≤ V_{DD}
Output leakage current ports 8-11	I_{LO2}			10 (Notes 1, 2)	μA	V_O = 12 V μPD7554; V_O = 9 V μPD7554A
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation; R oscillation (Note 3)	I_{DD1}		55	180	μA	V_{DD} = 3 V ±0.3 V; R = 150 kΩ ±2%
			40	150	μA	V_{DD} = 2.5 V; R = 150 kΩ ±2%
Supply current, HALT mode; R oscillation (Note 3)	I_{DD2}		25	80	μA	V_{DD} = 3 V ±0.3 V; R = 150 kΩ ±2%
			18	60	μA	V_{DD} = 2.5 V; R = 150 kΩ ±2%
Supply current, STOP mode (Note 3)	I_{DD3}		0.1	5	μA	
Supply current, data retention mode (Note 3)	I_{DDDR}		0.1	5	μA	V_{DDDR} = 2.0 V
Pull-up/down resistance, port 0, RESET	RP1	23.5	47	70.5	kΩ	
Pull-up resistance, ports 8-11	RP2	7.5	15	22.5	kΩ	

Notes:

(1) N-channel, open drain I/O ports, μPD7554.

(2) N-channel, open drain I/O ports, μPD7554A.

(3) Current in built-in pull-up/down resistors excluded.

DC Characteristics 2; V_{DD} = 2.7 to 6.0 V; μPD7554/54A/64/64A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.7 V_{DD}		V_{DD}	V	
Input high voltage CL1 (Note 2)	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	
Input high voltage ports 10, 11	V_{IH3}	0.7 V_{DD}		12 (Note 1); 9 (Note 2)	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1 (Note 3)	V_{IL1}	0		0.3 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.5	V	
Input leakage current except CL1 (Note 3)	I_{LI1}	−3		3	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11 (Note 4)	I_{LI3}			10	μA	V_I = 9 V (7554A); or 12 V
Output voltage high P0$_1$, P0$_2$, ports 8-11	I_{OH}	V_{DD} − 2.0			V	V_{DD} = 4.5 to 6.0 V; I_{OH} = −1 mA
	V_{OH}	V_{DD} − 1.0			V	V_{DD} = 2.7 V; I_{OH} = −100 μA
Output voltage low P0$_1$, P0$_2$	V_{OL}			0.4	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 1.6 mA
				0.5	V	I_{OL} = 400 μA
Output voltage low ports 10, 11	V_{OL}			0.4	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 1.6 mA
				2.0	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 10 mA
				0.5	V	I_{OL} = 400 μA
Output voltage low port 8	V_{OL}			2.0	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 15 mA
				0.5	V	I_{OL} = 600 μA
Output leakage current	I_{LO1}	−3		3	μA	0 V ≤ V_O ≤ V_{DD}
Output leakage current, port 8-11 (Note 4)	I_{LO2}			10	μA	V_O = 12 V μPD7554/64; V_O = 9 V μPD7554A/64A
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation; ceramic oscillation (Notes 3, 5)	I_{DD1}		650	2200	μA	V_{DD} = 5 V ±0.5 V; f_{CC} = 700 kHz
			120	360	μA	V_{DD} = 3 V ±10%; f_{CC} = 300 kHz
Supply current, normal operation; R oscillation (Note 3)	I_{DD1}		270	900	μA	V_{DD} = 5 V ±0.5 V; R = 56 kΩ ±2%
			80	240	μA	V_{DD} = 3 V ±10%; R = 100 kΩ ±2%
Supply current, HALT mode; ceramic oscillation (Notes 3, 5)	I_{DD2}		450	1500	μA	V_{DD} = 5 V ±0.5 V; f_{CC} = 700 kHz
			65	200	μA	V_{DD} = 3.0 V ±10%; f_{CC} = 300 kHz
Supply current, HALT mode; R oscillation (Note 3)	I_{DD2}		120	400	μA	V_{DD} = 5 V ±0.5 V; R = 56 kΩ ±2%
			35	110	μA	V_{DD} = 3 V ±10%; R = 100 kΩ ±2%
Supply current, STOP mode (Note 3)	I_{DD3}		0.1	10	μA	V_{DD} = 5 V ±0.5 V
			0.1	5	μA	V_{DD} = 3 V ±10%
Supply current, data retention mode (Note 3)	I_{DDDR}		0.1	5	μA	V_{DDDR} = 2.0 V
Pull-up/down resistance, port 0, RESET	RP1	23.5	47	70.5	kΩ	
Pull-up resistance, ports 8-11	RP2	7.5	15	22.5	kΩ	

Notes:

(1) μPD7554/64.

(2) μPD7554A/64A.

(3) Current in built-in pull-up/down resistors excluded.

(4) N-channel, open-drain I/O ports.

(5) μPD7564/64A.

DC Characteristics 3; V_{DD} = 2.0 to 3.3 V; µPD7554A only

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.8 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.2		V_{DD}	V	
Input high voltage ports 10, 11	V_{IH3}	0.85 V_{DD}		9	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1	V_{IL1}	0		0.15 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.2	V	
Input leakage current except CL1	I_{LI1}	−3		3	µA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	µA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11 (Note 1)	I_{LI3}			10	µA	V_I = 9 V
Output voltage high $P0_1$, $P0_2$, ports 8-11	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −70 µA
Output voltage low $P0_1$, $P0_2$, ports 10, 11	V_{OL}			0.5	V	$P0_1$, $P0_2$: I_{OL} = 270 µA Ports 10, 11: I_{OL} = 300 µA
Output voltage low port 8	V_{OL}			0.5	V	I_{OL} = 400 µA
Output leakage current	I_{LO1}	−3		3	µA	0 V ≤ V_O ≤ V_{DD}
Output leakage current ports 8-11 (Note 1)	I_{LO2}			10	µA	V_O = 9 V
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation; R oscillation (Note 2)	I_{DD1}		38	130	µA	V_{DD} = 3.0 V ±10%; R = 240 kΩ ±2%
			20	70	µA	V_{DD} = 2.0 V; R = 240 kΩ ±2%
Supply current, HALT mode; R oscillation (Note 2)	I_{DD2}		17	60	µA	V_{DD} = 3 V ±10%; R = 240 kΩ ±2%
			8	25	µA	V_{DD} = 2 V; R = 240 kΩ ±2%
Supply current, STOP mode (Note 2)	I_{DD3}		0.1	5	µA	
Supply current, data retention mode	I_{DDDR}		0.1	5	µA	V_{DDDR} = 2.0 V
Pull-up/down resistance, port 0, RESET	RP1	23.5	47	70.5	kΩ	
Pull-up resistance, ports 8-11	RP2	7.5	15	22.5	kΩ	

Notes:

(1) N-channel, open-drain I/O ports.

(2) Current in built-in pull-up/down resistors excluded.

AC Characteristics 1; V_{DD} = 2.5 to 3.3 V; μPD7554/54A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency	f_{CC}	140	180	220	kHz	R = 150 kΩ ±2%
System clock oscillation frequency, CL1, CL2	f_{CC}	140	175	210	kHz	V_{DD} = 2.5 V; R = 150 kΩ ±2%
External clock frequency, CL1	f_C	10		250	kHz	50% duty
System clock rise time, CL1	t_{CR}			200	ns	
System clock fall time, CL1	t_{CF}			200	ns	
System clock pulse width, high	t_{CH}	2		50	μs	
System clock pulse width, low	t_{CL}	2		50	μs	
External clock frequency ($P0_0$)	f_{P00}	0		250	kHz	50% duty
$P0_0$ rise time	t_{CRP00}			200	ns	
$P0_0$ fall time	t_{CFP0}			200	ns	
$P0_0$ pulse width, high	t_{P00H}	2			μs	
$P0_0$ pulse width, low	t_{P00L}	2			μs	
INT0 high time	t_{IOH}	30			μs	
INT0 low time	t_{IOL}	30			μs	
RESET high time	t_{RSH}	30			μs	
RESET low time	t_{RSL}	30			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	
$\overline{SCK}$ cycle time	t_{KCY}	8.0			μs	Input
		10.0			μs	Output
$\overline{SCK}$ pulse width, high	t_{KH}	4.0			μs	Input
$\overline{SCK}$ pulse width, low	t_{KL}	5.0			μs	Output
SI setup time to $\overline{SCK}$ ↑	t_{SIK}	0.3			μs	
SI hold time after $\overline{SCK}$ ↑	t_{KSI}	0.3			μs	
SO output delay time after $\overline{SCK}$ ↑	t_{KSO}			2.0	μs	C_{OUT} = 100 pF max

AC Characteristics 2; V_{DD} = 2.7 to 6.0 V; μPD7554/54A/64/64A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency (Note 1)	f_{CC}	400	500	600	kHz	V_{DD} = 4.5 to 6.0 V; R = 56 kΩ ±2%
		200	250	300	kHz	V_{DD} = 3 V ±10%; R = 100 kΩ ±2%
External clock frequency, CL1	f_C	10		710	kHz	V_{DD} = 4.5 to 6.0 V; 50% duty
		10		350	kHz	V_{DD} = 2.7 V; 50% duty
System clock oscillation frequency (Note 2)	f_{CC}	290	700	710	kHz	V_{DD} = 4.5 to 6.0V
		290	500	510	kHz	V_{DD} = 4.0 to 6.0 V
		290	400	410	kHz	V_{DD} = 3.5 to 6.0 V
		290	300	310	kHz	V_{DD} = 2.7 to 6.0 V
Oscillation stabilization time	t_{OS}	20			ms	V_{DD} = 2.7 to 6.0 V
System clock rise time, CL1	t_{CR}			200	ns	
System clock fall time, CL1	t_{CF}			200	ns	
System clock pulse width	t_{CH}	0.7		50	μs	V_{DD} 4.5 to 6.0 V
System clock pulse width, CL1	t_{CL}	1.45		50	μs	V_{DD} = 2.7 V
External clock frequency ($P0_0$)	f_{P00}	0		710	kHz	V_{DD} = 4.5 to 6.0 V; 50% duty
		0		350	kHz	V_{DD} = 2.7 V; 50% duty
$P0_0$ rise time	t_{CRP00}			200	ns	
$P0_0$ fall time	t_{CFP0}			200	ns	
$P0_0$ pulse width, high	t_{P00H}	0.7			μs	V_{DD} = 4.5 to 6.0 V
$P0_0$ pulse width, low	t_{P00L}	1.45			μs	V_{DD} = 2.7 V
INT0 high time	t_{IOH}	10			μs	
INT0 low time	t_{IOL}	10			μs	
RESET high time	t_{RSH}	10			μs	
RESET low time	t_{RSL}	10			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	
$\overline{SCK}$ cycle time	t_{KCY}	2.0			μs	Input; V_{DD} = 4.5 to 6.0 V
		2.5			μs	Output; ; V_{DD} = 4.5 to 6.0 V
		5.0			μs	Input; V_{DD} = 2.7 V
		5.7			μs	Output; ; V_{DD} = 2.7 V
$\overline{SCK}$ pulse width	t_{KH}	1.0			μs	Input; V_{DD} = 4.5 to 6.0 V
		1.25			μs	Output; V_{DD} = 4.5 to 6.0 V
$\overline{SCK}$ pulse width	t_{KL}	2.5			μs	Input; V_{DD} = 2.7 V
		2.85			μs	Output; V_{DD} = 2.7 V
SI setup time to $\overline{SCK}$ ↑	t_{SIK}	0.1			μs	
SI hold time after $\overline{SCK}$ ↑	t_{KSI}	0.1			μs	
SO output delay time after $\overline{SCK}$ ↑	t_{KSO}			0.85	μs	V_{DD} = 4.5 to 6.0 V; C_{OUT} = 100 pF max
				1.2	μs	V_{DD} = 2.7 V; C_{OUT} = 100 pF max

Notes:

(1) μPD7554/54A.

(2) μPD7564/64A.

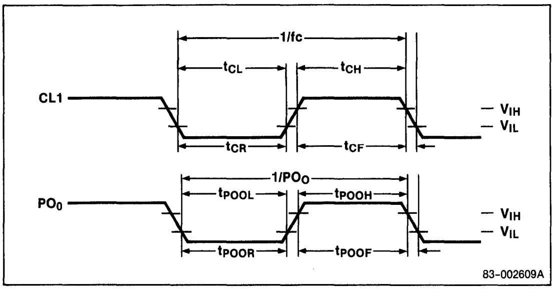
AC Characteristics 3; V_{DD} = 2.0 to 3.3 V; μPD7554A

T_A = −10 to +70°C; GND = 0 V

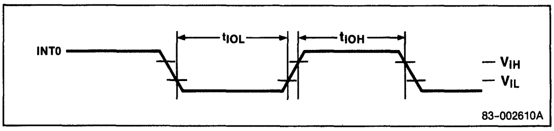
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency	f_{CC}	65	120	145	kHz	R = 240 kΩ ±2%
System clock oscillation frequency, CL1, CL2	f_{CC}	65	100	130	kHz	V_{DD} = 2.0 V; R = 240 kΩ ±2%
External clock frequency, CL1	f_C	10		150	kHz	
System clock rise time, CL1	t_{CR}			200	ns	
System clock fall time, CL1	t_{CF}			200	ns	
System clock pulse width, high	t_{CH}	3.3		50	μs	
System clock pulse width, low	t_{CL}	3.3		50	μs	
External clock frequency ($P0_0$)	f_{P00}	0		150	kHz	50% duty
$P0_0$ rise time	t_{CRP00}			200	ns	
$P0_0$ fall time	t_{CFP0}			200	ns	
$P0_0$ pulse width, high	t_{P00H}	3.3			μs	
$P0_0$ pulse width, low	t_{P00L}	3.3			μs	
INT0 high time	t_{IOH}	50			μs	
INT0 low time	t_{IOL}	50			μs	
RESET high time	t_{RSH}	50			μs	
RESET low time	t_{RSL}	50			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	
$\overline{\text{SCK}}$ cycle time	t_{KCY}	13.4			μs	Input
		16.6			μs	Output
$\overline{\text{SCK}}$ pulse width, high	t_{KH}	6.7			μs	Input
$\overline{\text{SCK}}$ pulse width, low	t_{KL}	8.3			μs	Output
SI setup time to $\overline{\text{SCK}}$ ↑	t_{SIK}	0.5			μs	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{KSI}	0.5			μs	
SO output delay time after $\overline{\text{SCK}}$ ↑	t_{KSO}			3.5	μs	C_{OUT} = 100 pF max

TIMING WAVEFORMS

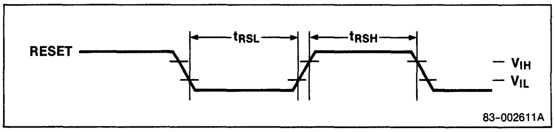
Clocks



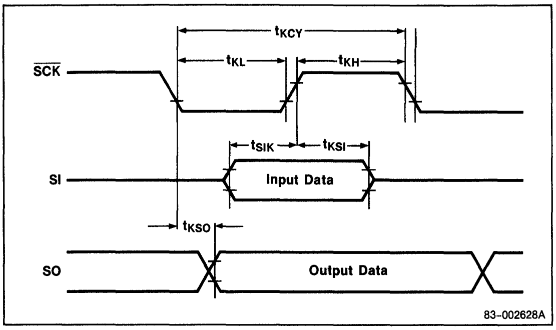
External Interrupt



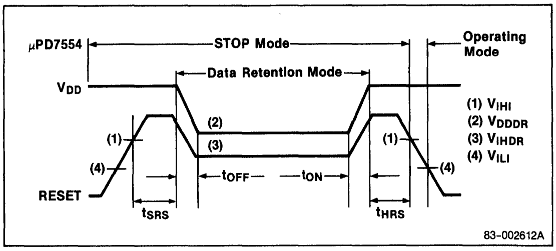
Reset



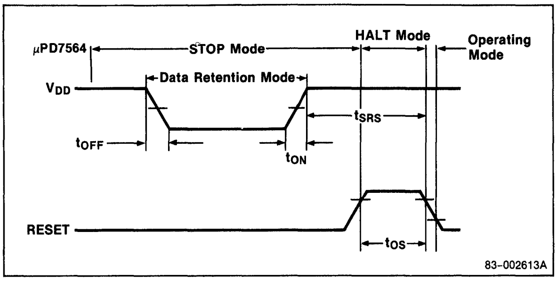
Serial Interface



Data Retention Mode, μPD7554/54A



Data Retention Mode, μPD7564/64A



NEC Electronics Inc.

μPD75P54/P64 4-Bit, Single-Chip, One-Time Programmable (OTP) CMOS Microcomputers With Serial I/O

Description

The μPD75P54 and μPD75P64 are 1024 x 8-bit on-chip, one-time programmable (OTP) ROM versions of the mask ROMs, μPD7554 and μPD7564. They have the same functions as, and are pin-compatible with, their mask ROMs. Because of their programming capabilities, the μPD75P54/P64 are suitable for evaluation and small lot production for system development. Their unique features will be described in this data sheet. For information about the base part μPD7554/64, please refer to its data sheet.

Features

□ 47 instructions (subset of μPD7500 set B)

□ Instruction cycle:
— External clock (μPD75P54): 2.86 μs/700 kHz, 5 V
— RC oscillator (μPD75P54): 4 μs/500 kHz, 5 V
— Ceramic oscillator (μPD75P64):
2.86 μs/700 kHz, 5 V

□ Program memory (ROM) of 1024 x 8 bits

□ Data memory (RAM) of 64 x 4 bits

□ 8-bit timer/event counter

□ 8-bit serial interface

□ I/O lines: 16-μPD75P54; 15-μPD75P64

□ Data memory retention at low supply voltage

□ CMOS technology

□ Low-power consumption

□ Single power supply:
— 4.5 to 6.0 V normal operation
— 6.0 V OTP

□ STOP, HALT standby functions

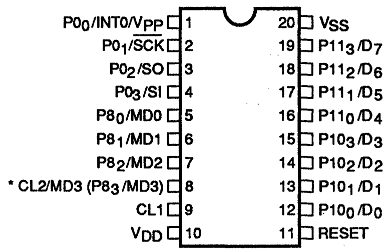
□ 20-pin plastic shrink DIP or SOP (OTP)

Ordering Information

Part Number	Package Type
μPD75P54CS	20-pin plastic shrink DIP (OTP)
μPD75P64CS	
μPD75P54G	20-pin plastic SOP (OTP)
μPD75P64G	

Pin Configuration

20-Pin Plastic Shrink DIP or SOP (OTP)



* $P8_3$ can be selected when using external clock on μPD75P54 only. CL2 can be selected when using RC oscillator.

49NR-650A

Pin Identification

Symbol	Function
P0$_0$/INT0/V$_{PP}$	4-bit input port 0/count clock input/serial interface. Programming voltage supply pin for program memory write/verify.
P0$_1$/$\overline{SCK}$	
P0$_2$/SO	
P0$_3$/SI	
P8$_0$-P8$_2$/MD0-MD2 CL2/MD3 (P8$_3$/MD3)	4-bit output port 8/OTP operation mode. Connection for ceramic resonator or RC (No P8$_3$ on μPD75P64) (Note 1)
CL1	Connection for ceramic resonator or RC
V$_{DD}$	4.5 to 6.0 V power supply, normal operation. 6.0 V for OTP.
RESET	Reset input pin
P10$_1$-P10$_3$/D$_0$-D$_3$	4-bit I/O port 10 and D$_0$-D$_3$ during programming write/verify.
P11$_0$-P11$_3$/D$_4$-D$_7$	4-bit I/O port 11 and D$_4$-D$_7$ during programming write/verify.
V$_{SS}$	Ground

Note:

(1) MD0-MD3 are used as mode select pins during programming.

PIN FUNCTIONS

P0$_0$/INT0/V$_{PP}$, P0$_1$/$\overline{SCK}$ P0$_2$/SO, P0$_3$/SI (Port 0/Count Clock Input/Programming/ Serial Interface)

4-bit input port 0/count clock input/serial I/O interface. This port can be configured as a 4-bit parallel input port or as the 8-bit serial I/O interface, under control of the serial mode select register. The serial input SI (active high), serial output SO (active low), as the serial clock $\overline{SCK}$ (active low—synchronizes data transfer) comprise the 8-bit serial I/O interface. If P0$_0$/INT0 is unused, connect it to ground. If any of P0$_1$-P0$_3$ are unused, connect them to ground. The port is in the input state at reset.

P8$_0$-P8$_2$/MD0-MD2, P8$_3$/MD3 (CL2/MD3) (Port 8/Clock Input/Mode Selection for OTP)

4-bit output port 8. This port can sink 15 mA and interface 12 V. P8$_3$ is a output port on the μPD75P64. On the μPD75P54, CL2 is one of the pins to which a resistor for RC oscillation is connected. On the μPD75P64, CL2 is one of the pins to which a ceramic resonator is connected. If any of P8$_0$-P8$_2$ pins are unused, leave them open. The port is in the high impedance state at reset. MD0-MD3 are used for OTP program memory write and read mode selection. There is no P8$_3$ on the μPD75P64.

CL1 (Clock Input 1)

On the μPD75P54, CL1 is one of the two pins to which a resistor for RC oscillation is connected. On the μPD75P64, CL1 is one of the two pins to which a ceramic resonator is connected.

V$_{DD}$ (Power Supply)

Positive power supply. 4.5 to 6.0 V for normal operation. 6.0 V for program memory write/verify.

RESET (Reset)

System reset input pin (active high). This pin is not internally connected to a pull-down resistor.

P10$_0$-P10$_3$/D$_0$-D$_3$ (Port 10/Data I/O)

4-bit I/O port. This port can sink 10 mA and interface 12 V. If any of these pins are unused, connect them to ground or V$_{DD}$ in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset. D$_0$-D$_3$ are 4-bit I/O pins for program memory write/verify.

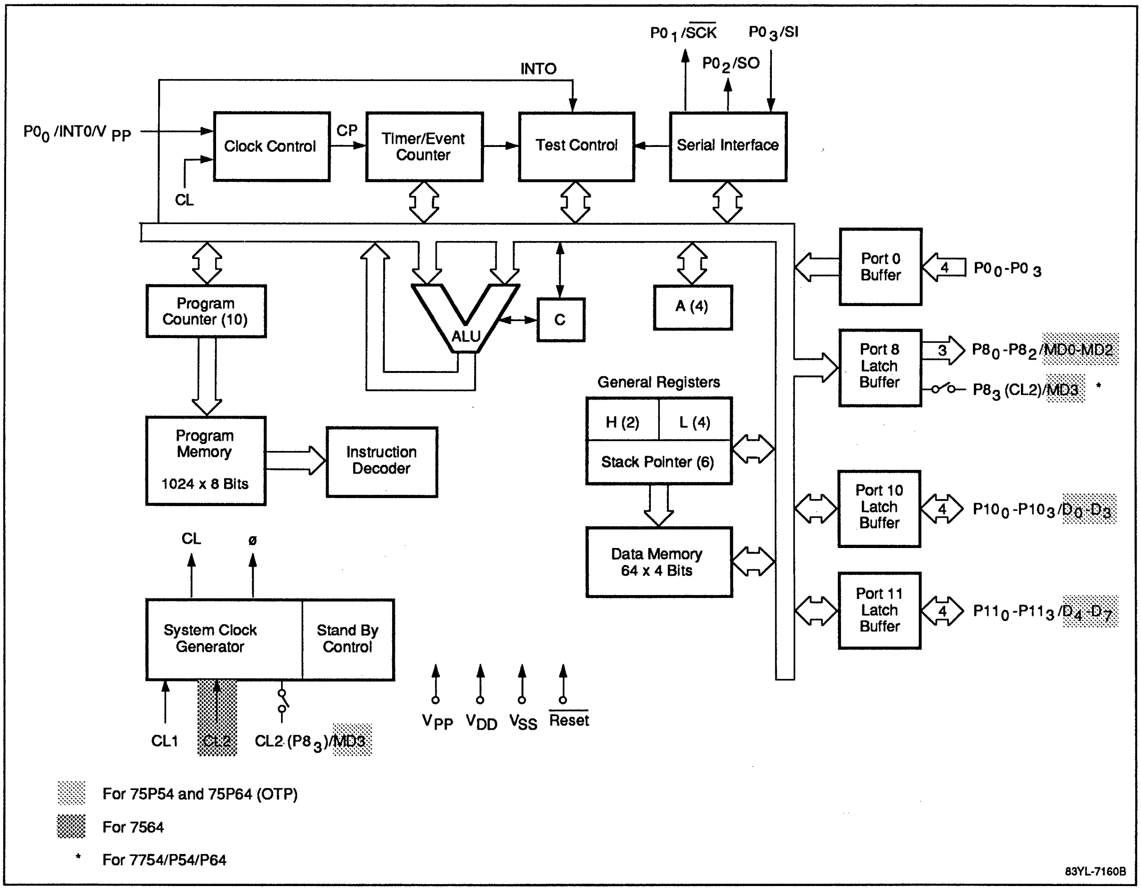
P11$_0$-P11$_3$/D$_4$-D$_7$ (Port 11/Data I/O)

4-bit I/O port. This port can sink 10 mA and interface 12 V. If any of these pins are unused, connect them to ground or V$_{DD}$ in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset. D$_4$-D$_7$ are 4-bit I/O pins for program memory write/verify.

V$_{SS}$ (Ground)

Ground.

Block Diagram

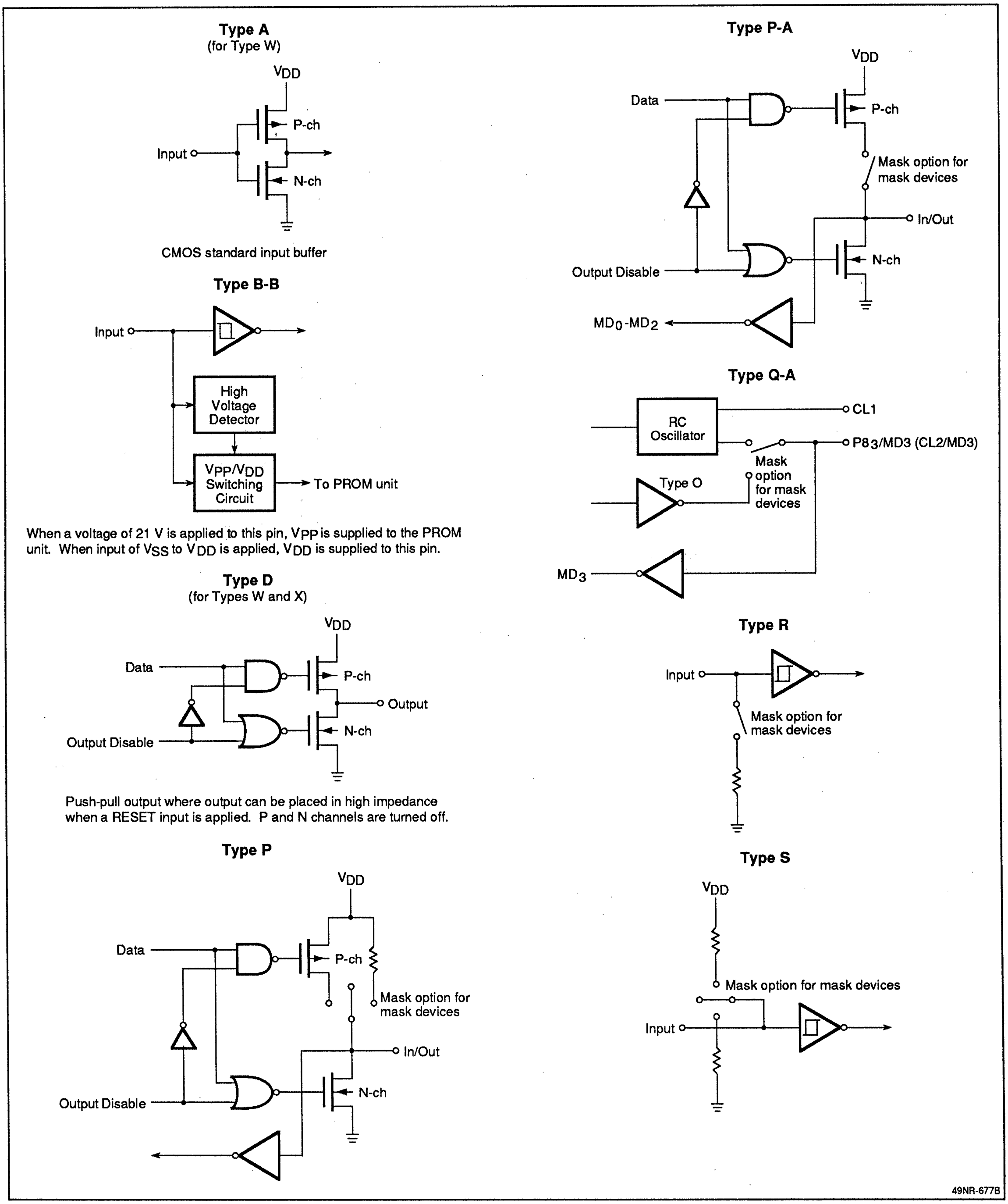


FUNCTIONAL DESCRIPTION

I/O Ports

Figure 1 shows the internal circuits at I/O ports 0, 8, 10, and 11.

Figure 1. Interface at I/O Ports



Type A
(for Type W)

CMOS standard input buffer

Type B-B

When a voltage of 21 V is applied to this pin, V_{PP} is supplied to the PROM unit. When input of V_{SS} to V_{DD} is applied, V_{DD} is supplied to this pin.

Type D
(for Types W and X)

Push-pull output where output can be placed in high impedance when a RESET input is applied. P and N channels are turned off.

Type P

Type P-A

Type Q-A

Type R

Type S

Figure 1. Interface at I/O Ports (cont)

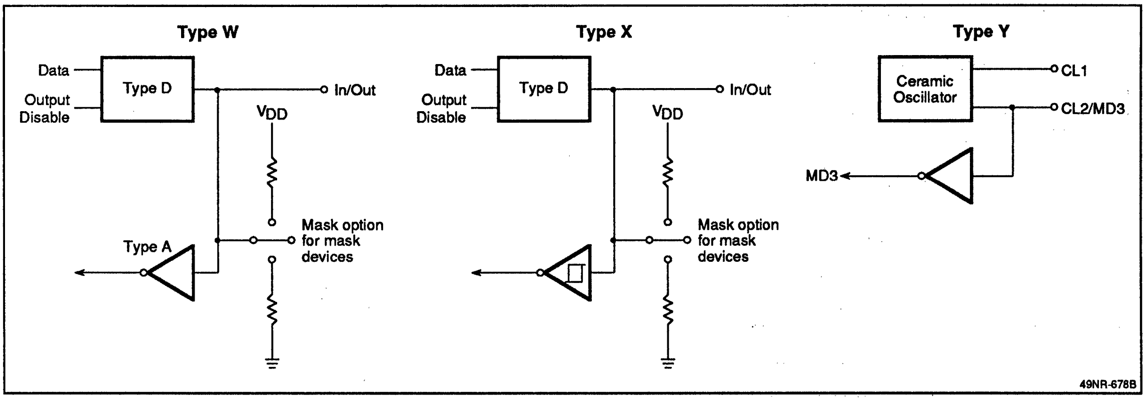


Table 1 compares the features of the μPD7554/64 and their OTP versions, μPD75P54/P64.

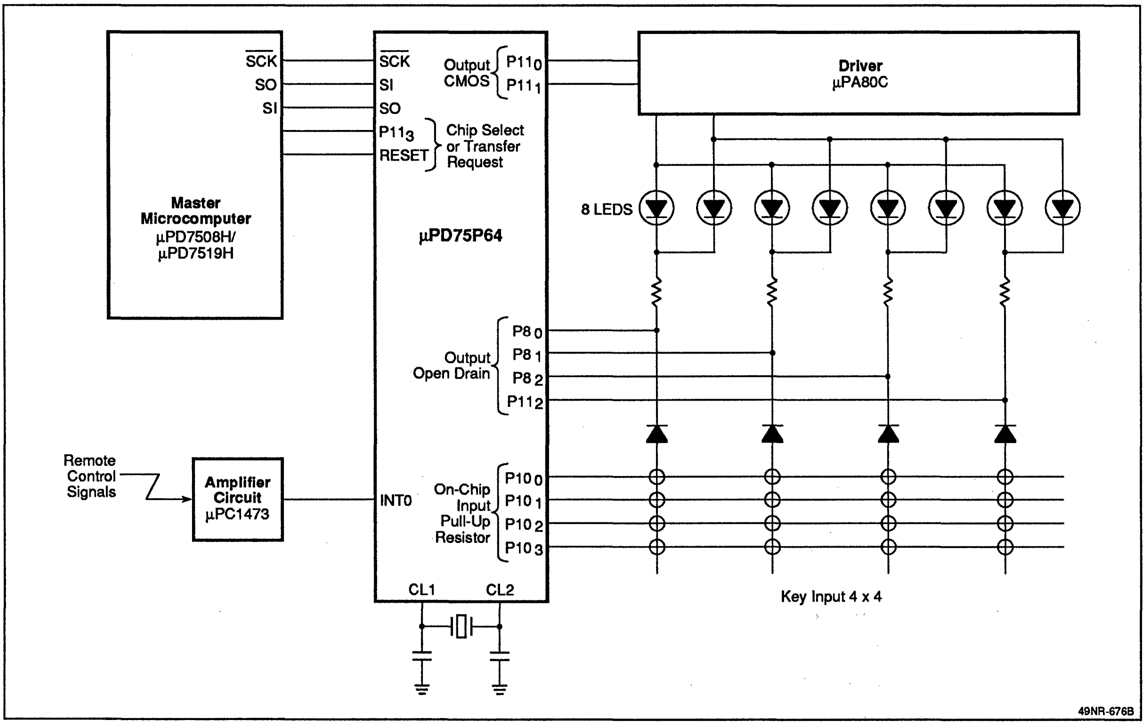
Table 1. Product Differences and Comparisons, μPD7554/64 and μPD75P54/P64

Item		μPD7554	μPD7564	μPD75P54 (OTP)	μPD75P64 (OTP)
Instruction cycle/system clock (5 V)	RC	4 μs/ 500 kHz	4 μs/ 500 kHz	4 μs/ 500 kHz	
	External	2.86 μs/ 700 kHz		2.86 μs/ 700 kHz	
	Ceramic		3 μs/ 660 kHz		2.86 μs/ 700 kHz
Instruction set		47 (set B)	47 (set B)	47 (set B)	47 (set B)
ROM or PROM		1024 x 8 mask ROM	1024 x 8 mask ROM	1024 x 8 one-time PROM	1024 x 8 one-time PROM
RAM		64 x 4	64 x 4	64 x 4	64 x 4
I/O port total		16 (max)	15	16 (max)	15
Port 0		P0$_0$-P0$_3$	P0$_0$-P0$_3$	P0$_0$-P0$_3$/MD0-MD3	P0$_0$-P0$_3$/MD0-MD3
P0$_0$ pin mask option		Available	Available	None	None
Port 8		P8$_0$-P8$_2$ P8$_3$/CL2	P8$_0$-P8$_2$	P8$_0$-P8$_2$/MD0-MD2 P8$_3$/MD3	P8$_0$-P8$_2$/MD0-MD2 CL2/MD3
Port 10		P10$_0$-P10$_3$	P10$_0$-P10$_3$	P10$_0$-P10$_3$/D$_0$-D$_3$	P10$_0$-P10$_3$/D$_0$-D$_3$
Port 11		P11$_0$-P11$_3$	P11$_0$-P11$_3$	P11$_0$-P11$_3$/D$_4$-D$_7$	P11$_0$-P11$_3$/D$_4$-D$_7$
Timer/event counter		8-bit	8-bit	8-bit	8-bit
Serial interface		8-bit	8-bit	8-bit	8-bit
Sense input		INT0, INTS, INTT	INT0, INTS, INTT	INT0, INTS, INTT	INT0, INTS, INTT
Supply voltage		2.5 to 6.0 V	2.7 to 6.0 V	4.5 to 6.0 V	4.5 to 6.0 V
Process		CMOS	CMOS	CMOS	CMOS
Package		20-pin plastic SOP	20-pin plastic SOP	20-pin plastic SOP	20-pin plastic SOP
		20-pin shrink DIP	20-pin shrink DIP	20-pin shrink DIP	20-pin shrink DIP
Output and I/O pins		N-channel open drain	N-channel open drain	N-channel open drain	N-channel open drain
Input pins		Mask options available	Mask options available	No on-chip resistor	No on-chip resistor
RESET		Mask options available	Mask options available	No pull-down resistor	No pull-down resistor

μPD75P64 Application

Figure 2 shows an example of an application circuit for remote-controlled data reception, key input, and LED display for the μPD75P64.

Figure 2. Remote-Controlled Data Reception, Key Input, and LED Display (μPD75P64)



OTP PROM (Program Memory Write and Verify)

The μPD75P54/P64 is a, one-time programmable (OTP) PROM version of the μPD7554/64. The OTP is programmed by the pins and their functions listed in table 2. During OTP programming, addresses are updated by inputting clocks, instead of addresses, from the CL1 pin.

Table 2. OTP Access

Pin	Function
V_{PP}	OTP programming voltage pin (normally V_{DD})
CL1	Address update clock input during programming
MD0-MD3	Mode selection during OTP programming
D_0-D_7	8-bit data I/O pins during OTP programming
V_{DD}	Supply voltage pin: 4.5 to 6.0 V during normal operation; 6 V during OTP programming

Notes:

The μPD75P54/P64 has no erasure window. The program memory data cannot be erased with ultraviolet light.

OTP Operation Mode

The μPD75P54/P64 operates in the program memory write/verify mode when +6 V is applied to V_{DD} and 21 V to V_{PP}. Mode pins MD0-MD3 select the operation modes shown in Table 3.

Table 3. OTP Operation Mode Selection
V_{PP} = +21 V; V_{DD} = +6 V

MD0	MD1	MD2	MD3	Operating Mode
H	L	H	L	Program memory address clear (Note 2)
L	H	H	H	Program memory write (Note 3)
L	L	H	H	Program memory verify (Note 4)
H	X	H	H	Program inhibit (Note 5)

Notes:

(1) X = L or H.

(2) While HLHL is being applied, the program counter continues to be cleared.

(3) While LHHH is being applied, data applied to D0-D7 continue to be written to the OTP.

(4) While LLHH is being applied, the OTP contents at the address that the program counter indicates continue to be output to P0-D7.

(5) While HXHH is being applied, the OTP continues to be nonaccessible, and D0-D7 remain at a high impedance level.

Program Memory Write Procedure. The program memory write procedure follows (high speed write is enabled):

(1) Connect unused pins to V_{SS} through a pull-down resistor. RESET is pulled up to V_{DD} through a resistor. Hold CL1 low.

(2) Supply 5 V to V_{DD} and V_{PP}.

(3) Select the program memory address clear mode.

(4) Change the voltage on V_{DD} to 6 V, and on V_{PP} to 21 V.

(5) Select the program inhibit mode.

(6) Write data in the 1 ms write mode.

(7) Select the program inhibit mode.

(8) Select the verify mode. If data is written correctly, proceed to step 9; if data is not written correctly, repeat steps 6-8.

(9) Perform an additional write of X (number of times a write was performed in steps 6-8) x 1 ms.

(10) Select the program inhibit mode.

(11) Increment the program memory address by one, by inputting four pulses to CL1.

(12) Repeat steps 6-11 until the end address occurs.

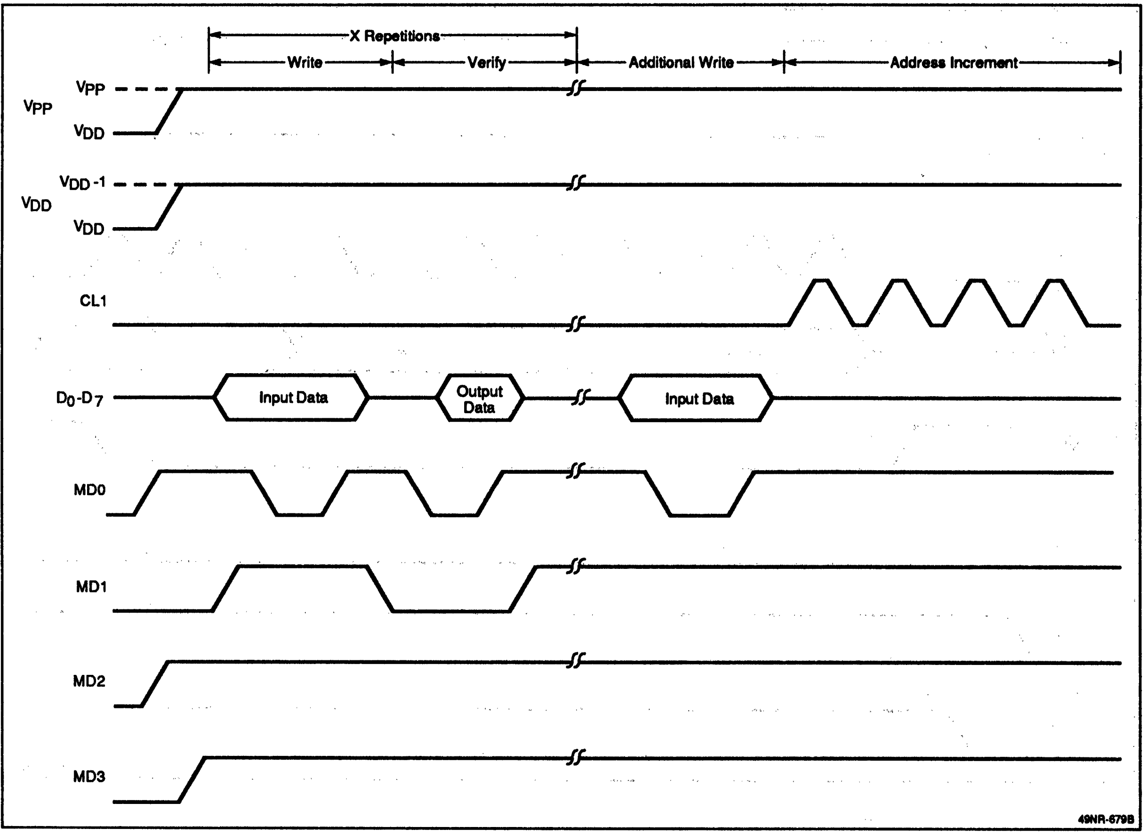
(13) Select the program memory address clear mode.

(14) Change the voltage on V_{DD} and V_{PP} to 5 V.

(15) Turn off power.

The timing for steps 2-11 is shown in figure 3.

Figure 3. Timing Diagram for OTP Program Memory Write



Program Memory Read Procedure. The program memory read procedure follows:

(1) Connect unused pins to V_{SS} through a pull-down resistor. RESET is pulled up to V_{DD} through a resistor. Hold CL1 low.

(2) Supply 5 V to V_{DD} and V_{PP}.

(3) Select the program memory address clear mode.

(4) Change the voltage on V_{DD} to 6 V, and on V_{PP} to 21 V.

(5) Select the program inhibit mode.

(6) Select the verify mode. Data is read from "000H." Upon entry of a clock pulse to CL1, data is sequentially output by one address in a cycle of four pulses.

(7) Pulse input to the CL1. Program memory address is updated at the rising edge of the third pulse. Address after updated one (+1) is updated every four pulses. Repeat update to last address.

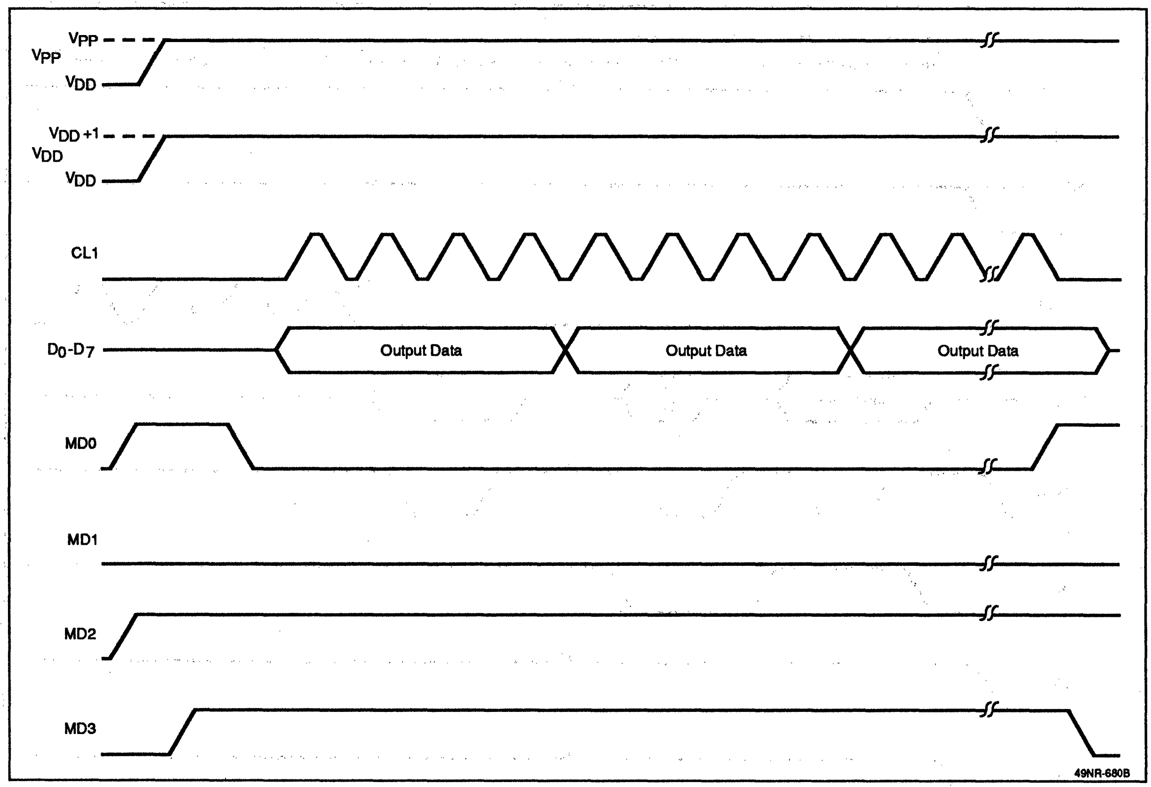
(8) Select the program memory address clear mode.

(9) Change the voltage on V_{DD} and V_{PP} to 5 V.

(10) Turn off power.

The timing for steps 2-9 is shown in figure 4.

Figure 4. Timing Diagram for Program Memory Read



ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C
Power supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_I	
Except ports 10, 11	−0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 1)	−0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 2)	−0.3 to +13 V
Output voltage, V_O	
Except ports 10, 11	−0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 1)	−0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 2)	−0.3 to +13 V
Output current, high I_{OH}	
One pin	−5 mA
All output pins, total	−15 mA
Output current, low I_{OL}	
PO_1, PO_2	5 mA
Ports 10, 11	15 mA
Port 8	30 mA
All ports, total	100 mA
Power dissipation, P_D (T_A = +70°C)	
Shrink DIP	480 mW
SOP	250 mW

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Notes:

(1) CMOS I/O or N-channel open drain + internal pull up resistor.

(2) N-channel open drain I/O.

Capacitance

T_A = 25°C, V_{DD} = V_{SS} = 0 V; f = 1 MHz. Unmeasured pins returned to V_{SS}.

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I			50	pF	PO_0
				15	pF	PO_3
Output capacitance	C_O			35	pF	Port 8
I/O capacitance	$C_{I/O}$			35	pF	Ports 10, 11 and PO_1, PO_2

DC Characteristics, Normal Operation; V_{DD} = 4.5 to 6.0 V; V_{SS} = 0 V

T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.7 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	
Input high voltage ports 10, 11 (Note 1)	V_{IH3}	0.7 V_{DD}		12	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1	V_{IL1}	0		0.3 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.5	V	
Input leakage current except CL1	I_{LI1}	−3		3	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11	I_{LI3}			10 (Note 1)	μA	V_I = 12 V
Output voltage high $P0_1$, $P0_2$, ports 8-11	V_{OH}	V_{DD} − 2.0			V	I_{OH} = −1 mA
Output voltage low $P0_1$, $P0_2$, ports 10, 11	V_{OL}			0.4	V	$P0_1$, $P0_2$: I_{OL} = 1.6 mA; Ports 10, 11: I_{OL} = 1.6 mA
Output voltage low ports 8, 10, 11	V_{OL}			2.0	V	Port 8: I_{OL} = 15 mA Ports 10, 11: I_{OL} = 10 mA
Output leakage current	I_{LO1}	−3		3	μA	0 V ≤ V_O ≤ V_{DD}
Output leakage current ports 8-11	I_{LO2}			10 (Note 1)	μA	V_O = 12 V
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation	I_{DD1}		400	1400	μA	μPD75P54: V_{DD} = 5 V ±10%; R = 56 kΩ ±2%
			700	2300	μA	μPD75P64: V_{DD} = 5 V ±10%; f_{CC} = 700 kHz
Supply current, HALT mode	I_{DD2}		120	400	μA	μPD75P54: V_{DD} = 5 V ±10%; R = 56 kΩ ±2%
			450	1500	μA	μPD 75P64: V_{DD} = 5 V ±10%; f_{CC} = 700 kHz
Supply current, STOP mode	I_{DD3}		0.1	10	μA	V_{DD} = 5.0 V ±10%
Supply current, data retention mode	I_{DDDR}		0.1	5	μA	V_{DDDR} = 2.0 V

Notes:

(1) N-channel, open drain I/O ports.

DC Characteristics, Programming Mode; $V_{DD} = 6.0 \pm 0.25$ V; $V_{PP} = 21 \pm 0.5$ V, $V_{SS} = 0$ V (Notes 1 and 2)

$T_A = 25°C$

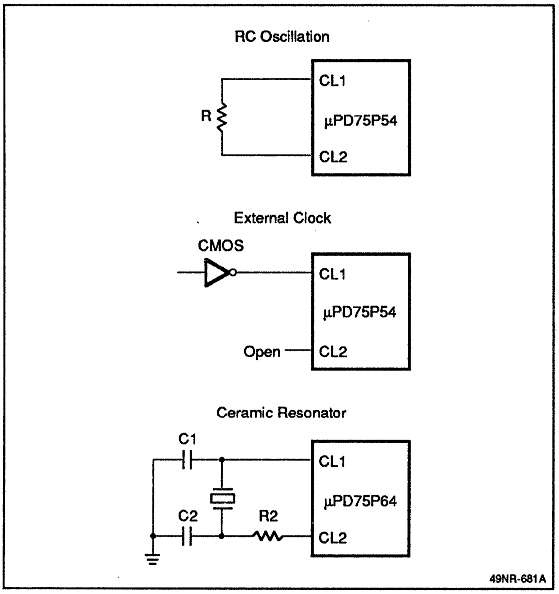
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.7 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} – 0.5		V_{DD}	V	
Input low voltage except CL1	V_{IL1}	0		0.3 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.5	V	
Input leakage current	I_{LI}			10	µA	$V_I = V_{IL}$ or V_{IH}
Output voltage high	I_{OH}	V_{DD} – 2.0			V	$I_{OH} = -1$ mA
Output voltage low	V_{OL}			0.4	V	$I_{OL} = 1.6$ mA
V_{DD} supply voltage	I_{DD}			30	mA	
V_{PP} supply current	I_{PP}			30	mA	MD0 = V_{IL}, MD1 = V_{IH}

Notes:

(1) V_{PP}, including an overshoot, should not exceed +22 V.

(2) Apply V_{DD} before V_{PP}, and cut off after V_{PP}.

Figure 5. Recommended Circuits



AC Characteristics, Normal Operation; V_{DD} = 4.5 to 6.0 V; V_{SS} = 0 V

T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency	f_{CC}	400	500	600	kHz	μPD75P54: R = 56 kΩ ±2%
		290		710	kHz	μPD75P64: R = 100 kΩ ±2%
External clock frequency, CL1	f_C	10		710	kHz	μPD75P54: 50% duty
Oscillation stabilization time	t_{OS}	20			ms	μPD75P64 (Note 1)
System clock rise time, CL1	t_{CR}			0.2	μs	
System clock fall time, CL1	t_{CF}			0.2	μs	
System clock pulse width	t_{CH}	0.7		50	μs	
System clock pulse width, CL1	t_{CL}	0.7		50	μs	
Event input frequency ($P0_0$)	f_{P0}	0		710	kHz	50% duty
$P0_0$ rise time	t_{POR}			200	ns	
$P0_0$ fall time	t_{POF}			200	ns	
$P0_0$ pulse width, high	t_{P0H}	0.7			μs	V_{DD} = 4.5 to 6.0 V
$P0_0$ pulse width, low	t_{P0L}	0.7			μs	V_{DD} = 2.7 V
INT0 high time	t_{IOH}	10			μs	
INT0 low time	t_{IOL}	10			μs	
RESET high time	t_{RSH}	10			μs	
RESET low time	t_{RSL}	10			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	
$\overline{SCK}$ cycle time	t_{KCY}	2.0			μs	Input
		2.5			μs	Output
$\overline{SCK}$ pulse width, high	t_{KH}	1.0			μs	Input
$\overline{SCK}$ pulse width, low	t_{KL}	1.25			μs	Output
SI setup time to $\overline{SCK}$ ↑	t_{SIK}	0.1			μs	
SI hold time after $\overline{SCK}$ ↑	t_{KSI}	0.1			μs	
SO output delay time after $\overline{SCK}$ ↑	t_{KSO}			0.85	μs	CL = 100 pF

Notes:

(1) Hold the RESET signal at a high level until oscillation becomes stable.

AC Characteristics, Programming Mode; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 21 ±0.5 V, V_{SS} = 0 V

T_A = 25°C

Parameter	Symbol	Note 1	Min	Typ	Max	Unit	Conditions
Address setup time for MD0 ↓ (Note 2)	t_{AS}	t_{AS}	2			µs	
MD1 setup time for MD0 ↓	t_{MIS}	t_{OES}	2			µs	
Data setup for MD0 ↓	t_{DS}	t_{DS}	2			µs	
Address hold time for MD0 ↑ (Note 2)	t_{AH}	t_{AH}	2			µs	
Data hold time for MD0↑	t_{DH}	t_{DH}	2			µs	
MD0 ↑ to data output float delay time	t_{DF}	t_{DF}	0		200	ns	
V_{PP} setup time for MD3 ↑	t_{VPS}	t_{VPS}	2			µs	
V_{DD} setup time for MD3 ↑	t_{VDS}	t_{VCS}	2			µs	
Initial program pulse width	t_{PW}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{OPW}	t_{OPW}	0.95		21.0	ms	
MD0 setup time for MD1 ↑	t_{MOS}	t_{CES}	2			µs	
MD0 ↓ to data output delay time	t_{DV}	t_{DV}			1 (Note 3)	µs	MD0 = MD1 = V_{IL}
MD1 hold time for MD0 ↑	t_{M1H}	t_{OEH}	2			µs	t_{M1H} + t_{MIR} ≥ 50 µs
MD1 recovery time for MD0 ↓	t_{M1R}	t_{OR}	2			µs	
Program counter reset time	t_{PCR}		10			µs	
CL1 input high- and low-level widths	t_{XH}, t_{XL}		0.7			µs	
CL1 input frequency	f_X				710	kHz	
Initial mode set time	t_I		2			µs	
MD3 setup time for MD1 ↑	t_{M3S}		2			µs	
MD3 hold time for MD1 ↓	t_{M3H}		2			µs	
MD3 setup time for MD0 ↓	t_{M3SR}		2			µs	During program memory read
Address to data output delay time (Note 2)	t_{DAD}	t_{ACC}	2			µs	
Address to data output hold time (Note 2)	t_{HAD}	t_{OH}	0		300	ns	
MD3 hold time for MD0 ↑	t_{M3HR}		2			µs	
MD3 ↓ to data output float delay time	t_{DFR}		2			µs	

Notes:

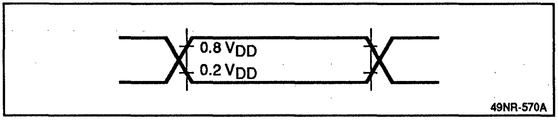
(1) Symbol of the corresponding µPD27C256.

(2) "1" is added to the internal address signal at the rising edge of the third CL1 input. The signal is not input to the pin.

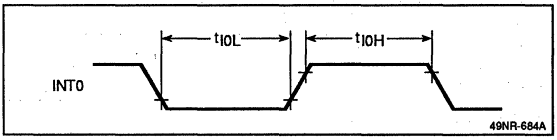
(3) During CMOS output.

Timing Waveforms

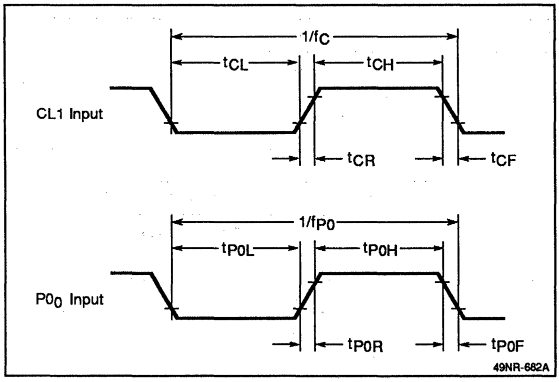
AC Test Points



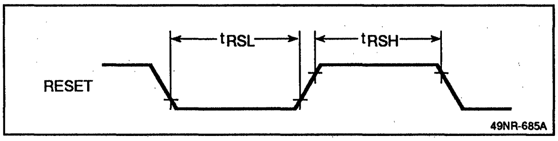
Test



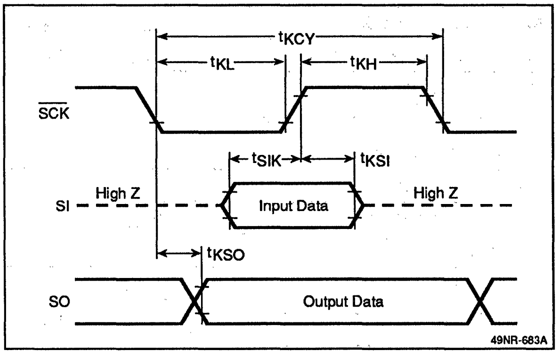
Clock



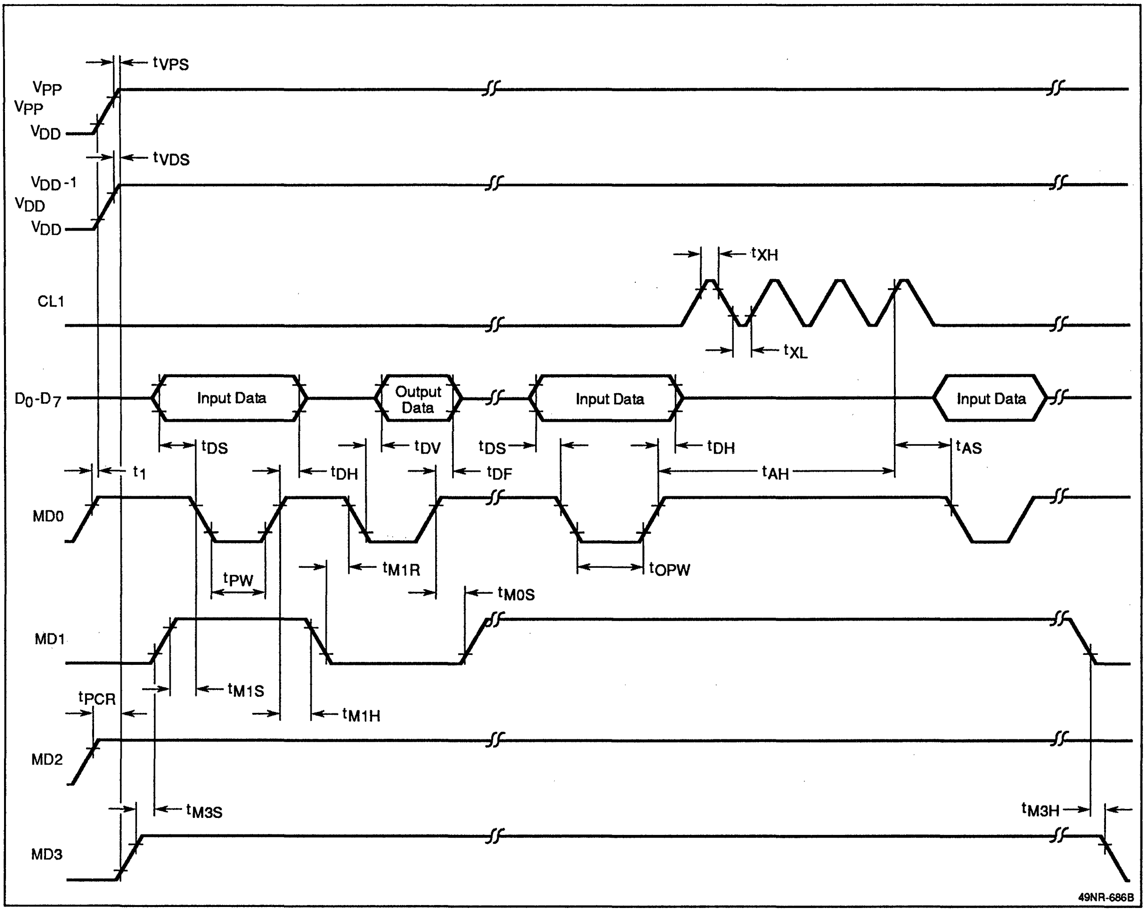
RESET



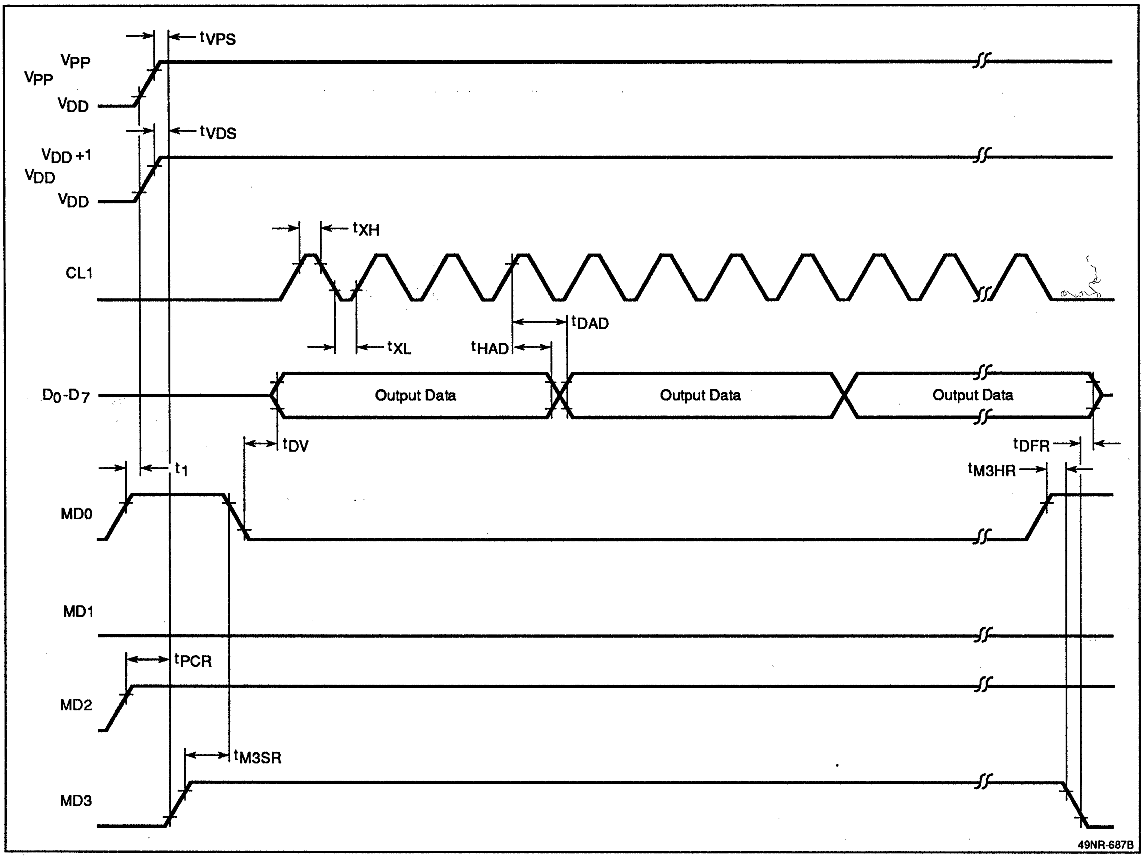
Serial Transfer



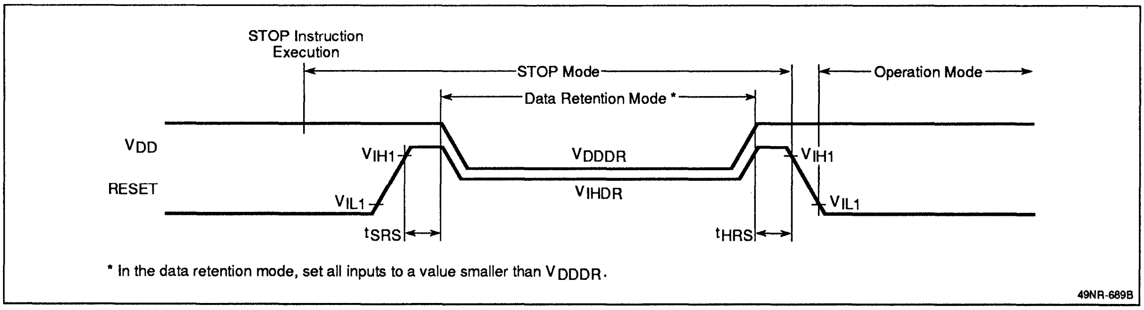
Program Memory Write



Program Memory Read



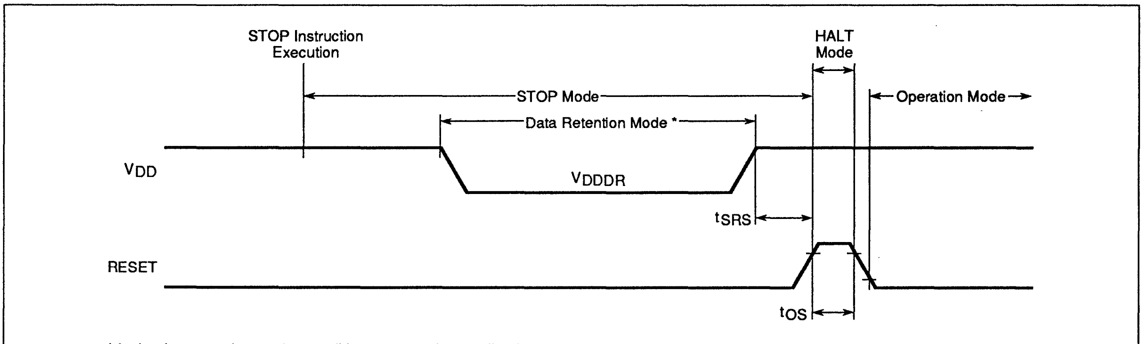
Data Retention Timing, μPD75P54



* In the data retention mode, set all inputs to a value smaller than V_{DDDR}.

49NR-689B

Data Retention Timing, μPD75P64



* In the data retention mode, set all inputs to a value smaller than V_{DDDR}.

49NR-688B

NEC
NEC Electronics Inc.

μPD7556/56A/66/66A 4-Bit, Single-Chip CMOS Microcomputers With Comparator

Description

The μPD7556/66A and μPD7566/66A are low-end versions of μPD7500 series products. These microcomputers incorporate a 4-bit comparator input and are useful as slave CPUs to high-end μPD7500 series or 8-bit μCOM-87 series products.

The μPD7556/56A/66/66A has output ports that can directly drive triacs and LEDs. Also, various mask-optional I/O circuits can be configured for a wide selection of outputs allowing a reduction of external circuitry in your design. There are two testable interrupts.

The μPD7556/56A and μPD7566/66A differ only in their clock circuitry. The μPD7556/56A uses an external resistor with an internal capacitor for an RC oscillator clock, where the μPD7566/66A uses a ceramic oscillator as a clock. These microcomputers are ideally suited to control devices such as air conditioners, microwave ovens, refrigerators, rice cookers, and audio equipment.

Features

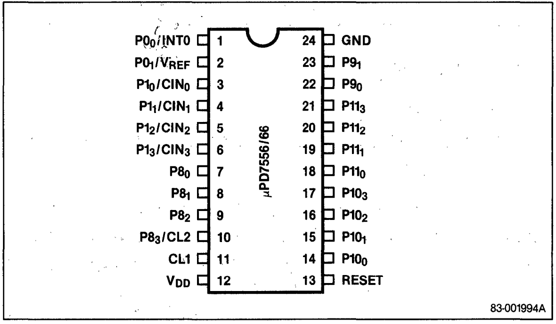
- □ 45 instructions (subset of μPD7500 set B)
- □ Instruction cycle:
  - — External clock: 2.86 μs/700 kHz, 5 V
  - — RC oscillator (μPD7556/56A); 4 μs/500 kHz, 5 V
  - — Ceramic resonator (μPD7566/66A): 2.86 μs/700 kHz, 5 V
- □ Program memory (ROM) of 1024 x 8-bits
- □ Data memory (RAM) of 64 x 4-bits
- □ 8-bit timer/event counter
- □ 4-bit comparator
- □ I/O lines: 20-μPD7556/56A; 19-μPD7566/66A
- □ Data memory retention at low supply voltage
- □ Standby (STOP/HALT) functions
- □ CMOS technology
- □ Low-power consumption
- □ Single power supply
  - — 2.5 to 6.0 V (μPD7556/56A)
  - — 2.7 to 6.0 V (μPD7566/66A)
  - — 2.0 to 6.0 V (μPD7556A)

Ordering Information

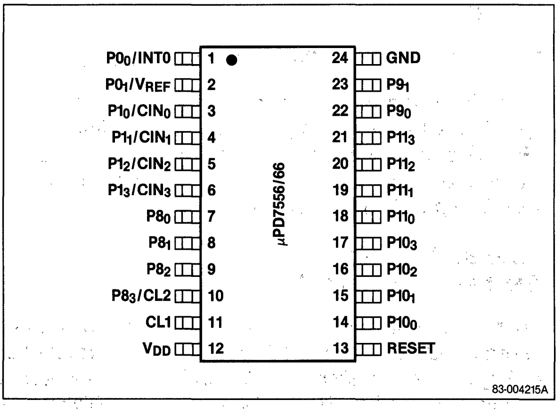
Part Number	Package Type
μPD7556CS	24-pin plastic shrink DIP
μPD7556ACS	
μPD7566CS	
μPD7566ACS	
μPD7556G	24-pin plastic SOP
μPD7556AG	
μPD7566G	
μPD7566AG	

Pin Configurations

20-Pin Plastic Shrink DIP



20-Pin Plastic SOP



Pin Identification

Symbol	Function
$P0_0$/INT0 $P0_1/V_{REF}$	2-bit input port 0/testable input pin/ comparator reference voltage input pin
$P1_0/CIN_0$ $P1_1/CIN_1$ $P1_2/CIN_2$ $P1_3/CIN_3$	4-bit input port 1/4-bit comparator inputs
$P8_0$-$P8_2$ $P8_3$/CL2	3-bit output port 8 (μPD7566/66A) 3- (4-) bit output port 8/connection for RC oscillator (μPD7556/56A)/ceramic resonator (μPD7566/66A)
CL1	Connection for ceramic resonator or RC oscillator
V_{DD}	+5 V power supply
RESET	Reset input pin
$P10_1$-$P10_3$	4-bit I/O port 10
$P11_0$-$P11_3$	4-bit I/O port 11
$P9_0$-$P9_1$	2-bit output port 9
V_{SS}	Ground

PIN FUNCTIONS

$P0_0$/INT0, $P0_1/V_{REF}$ (Port 0/Count Clock input/Comparator reference voltage input)

Two-bit input port 0/count clock input/comparator reference voltage input. INT0 is an edge-sensitive testable input pin that detects a signal at the rising edge. V_{REF} is the comparator reference voltage input pin. A mask option specifies whether this pin is used as $P0_1$ or V_{REF}. If $P0_0$/INT0 is unused; connect it to ground. If $P0_1/V_{REF}$ is unused, connect it to ground or V_{DD}. The port is in the input state at reset.

$P1_0/CIN_0$-$P1_3/CIN_3$ (Port 1/Comparator inputs)

Four-bit input port 1/comparator inputs. A mask option specifies whether these pins are used as digital input (Port 1) or as comparator inputs (CIN_0-CIN_3). If any of $P1_0$-$P1_3$ pins are unused, connect them to ground or V_{DD}. The port is the input state at reset.

$P8_0$-$P8_2$, $P8_3$-CL2 (Port 8/Clock input 2)

Four-bit output port 8. This port can sink 15 mA and interface 12 V(1). On the μPD7556/56A, the port function of $P8_3$/CL2 is specified by mask option. $P8_3$ is a normal output port on the μPD7566/66A. On the μPD7556/56A, CL2 is one of the pins to which a resistor for RC oscillation is connected. On the μPD7566/66A, CL2 is one of the pins to which a ceramic resonator is connected. If any of $P8_0$-$P8_2$ pins are unused, leave them open. The port is in the high impedance state at reset.

CL1 (Clock input 1)

On the μPD7566/66A, CL1 is one of the two pins to which a resistor for RC oscillation is connected. On the μPD7566/66A, CL1 is one of the two pins to which a ceramic resonator is connected.

V_{DD} (Power supply)

Positive power supply.

RESET (Reset)

System reset input pin (active high). This pin can be internally connected to a pull-down resistor if specified by mask option.

$P10_0$-$P10_3$ (Port 10)

Four-bit I/O port. This port can sink 10 mA and interface 12 V(1). If any of these pins are unused, connect them to ground or V_{DD} in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset.

$P11_0$-$P11_3$ (Port 11)

Four-bit I/O port. This port can sink 10 mA and interface 12 V(1). If any of these pins are unused, connect them to ground or V_{DD} in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset.

$P9_0$-$P9_1$ (Port 9)

Two-bit output port. This port sinks 15 mA and can interface to 12 V(1). If either of these pins are unused, leave it open. The port is in the high impedance state at reset.

V_{SS} (Ground)

Ground.

Note: (1) 9 V for the μPD7556A/66A.

Pin Mask Options

Table 1 shows the mask options for all the port pins and the RESET pin. You may select these options in bit units.

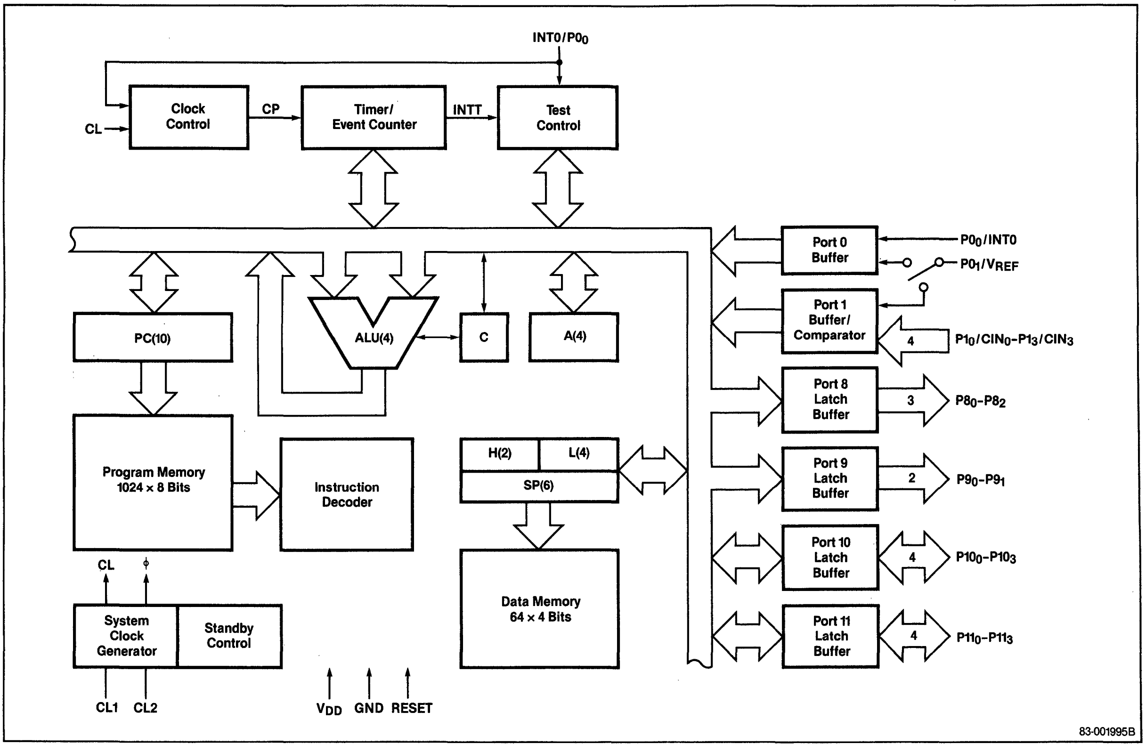
Table 1. Pin Mask Options

Pin	Options
$P0_0$	1 No connection to internal resistor 2 Connected to internal pull-up resistor 3 Connected to internal pull-down resistor
$P0_1/V_{REF}$	1 No connection to internal resistor 2 Connected to internal pull-up resistor 3 Connected to internal pull-down resistor 4 Used as V_{REF} pin 1 A bias of $V_{DD}/2$ internally applied to V_{REF} pin 2 Bias not applied
$P1_0/CIN_0$- $P1_3/CIN_3$	1 No connection to internal resistor 2 Connected to internal pull-up resistor 3 Connected to internal pull-down resistor 4 Used as comparator input pins
$P8_0$-$P8_2$/ $P9_0$-$P9_1$	1 CMOS (push-pull) output 2 N-channel, open-drain output
$P8_3/CL_2$ (1) option 1	1 Used as $P8_3$ 2 Used as CL2
$P8_3/CL_2$ (1) option 2	1 CMOS (push-pull) 2 N-channel open-drain
$P10_0$-$P10_3$ $P11_0$-$P11_3$	1 N-channel, open drain input/output 2 CMOS (push-pull) input/output 3 N-channel, open-drain input/output with internal pull-up resistor
RESET	1 Connected to internal pull-down resistor 2 Not connected to internal pull-down resistor

Notes:

(1) μPD7556/56A only.

μPD7556/56/66/66AA Block Diagram

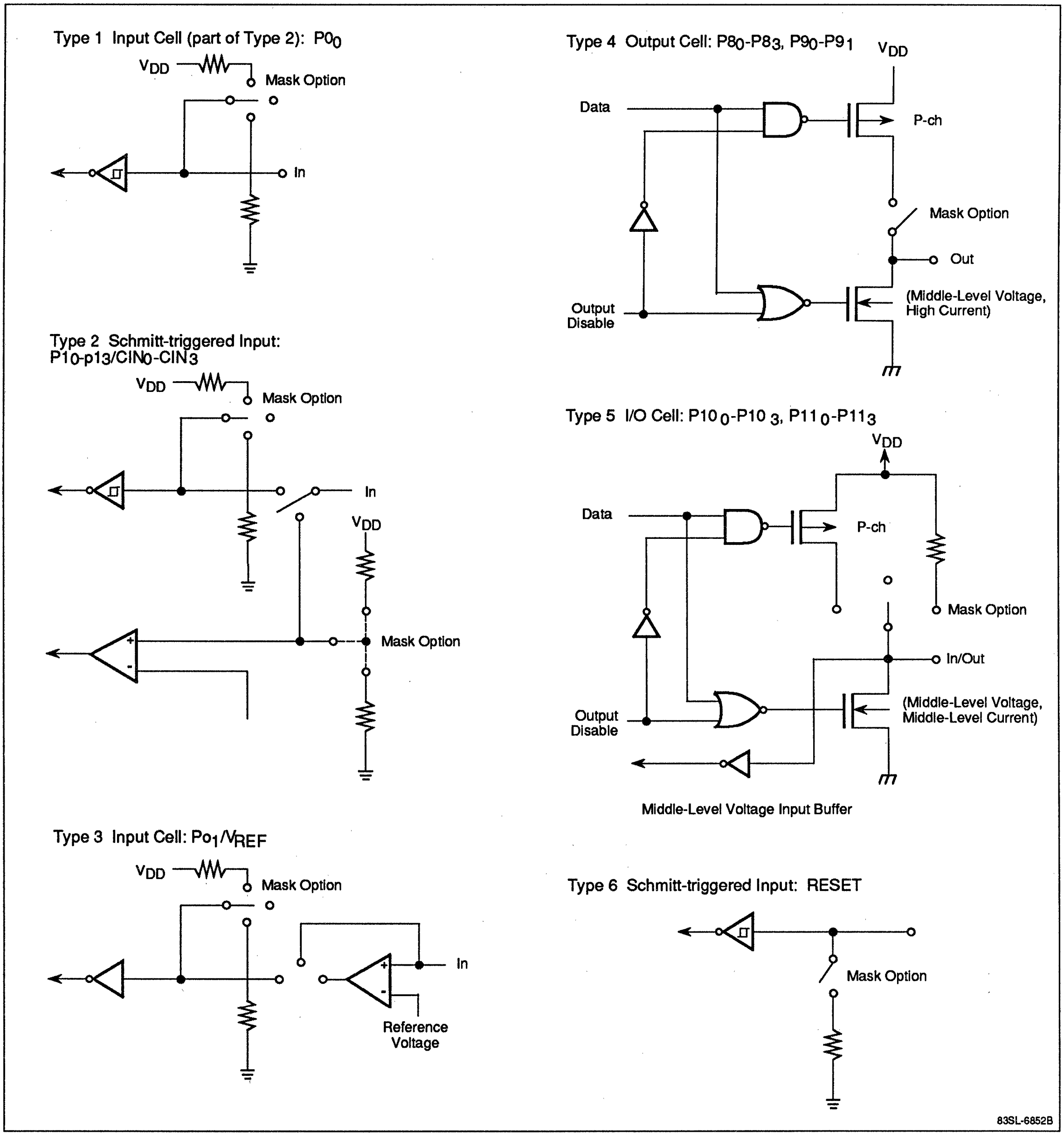


FUNCTIONAL DESCRIPTION

I/O Ports

Figure 1 shows the internal circuits at I/O ports P0, P8, P10, and P11.

Figure 1. I/O Ports


Type 1 Input Cell (part of Type 2): P0$_0$
VDD
Mask Option
In
Type 2 Schmitt-triggered Input: P1$_0$-p1$_3$/CIN$_0$-CIN$_3$
VDD
Mask Option
In
VDD
+
-
Mask Option
Type 3 Input Cell: Po$_1$/V$_{REF}$
VDD
Mask Option
+
-
In
Reference Voltage
Type 4 Output Cell: P8$_0$-P8$_3$, P9$_0$-P9$_1$
VDD
Data
P-ch
Mask Option
Out
Output Disable
(Middle-Level Voltage, High Current)
Type 5 I/O Cell: P10$_0$-P10$_3$, P11$_0$-P11$_3$
VDD
Data
P-ch
Mask Option
In/Out
Output Disable
(Middle-Level Voltage, Middle-Level Current)
Middle-Level Voltage Input Buffer
Type 6 Schmitt-triggered Input: RESET
Mask Option
83SL-6852B

Program Memory

The μPD7556/56A/66/66A has a mask-programmable ROM with a capacity of 1024 words by 8 bits for program storage. It is addressed by the program counter. The reset start address is 000H. Figure 2 shows the program memory map.

General-Purpose Registers

Two registers, H(2-bit) and L(4-bit) are provided as general-purpose registers. Each register can be individually manipulated. The two registers also form pair register HL; H being the high register and L being the low one. The HL register is a data pointer to address data memory. Figure 3 shows the configuration of the general-purpose registers.

The L register also specifies an I/O port or mode register when an I/O instruction (IPL or OPL) is executed. It also specifies the bits of a port when the SPBL or RPBL instruction is executed.

Data Memory

The data memory is static RAM with a capacity of 64 words by 4 bits. Part of this memory is used as the stack area. The data memory is also used in 8-bit data processing when paired with the accumulator. Figure 4 shows the data memory map.

Data memory can be addressed directly, with the immediate data from an instruction; indirectly, with the contents of HL (including auto-increment and auto-decrement); and indirectly by the contents of the stack pointer.

You may use any area of the data memory as the stack. The boundary of the stack is determined by how the TAMSP instruction initializes the stack pointer. Once the boundary is set, a call or return instruction automatically accesses the stack.

When a call instruction is executed, the contents of the program counter and the program status word (PSW) are stored to the stack in the sequence shown in figure 5.

When a return instruction is executed, the contents of the program counter are automatically restored, but the PSW is not. The contents of data memory can be retained with a low supply voltage during STOP mode.

Accumulator

The accumulator is a 4-bit register used in arithmetic operations. The accumulator can process 8-bit data with paired data addressed by HL. Figure 6 shows the configuration of the accumulator.

Arithmetic Logic Unit

The arithmetic logic unit (ALU) is a 4-bit arithmetic circuit that performs operations such as binary addition, logical operation, increment, decrement, comparison, and bit processing.

Program Status Word

The program status word (PSW) consists of two skip flags (SK0 and SK1), a carry flag (C), and bit 1, which is always zero. Figure 7 shows the configuration of the PSW.

The contents of the PSW are stored to the stack when a call instruction is executed, but are not restored from the stack by the return instruction.

The skip flags retain the following skip conditions: string effect by LAI or LHLI instruction, and skip condition satisfied by an instruction other than a string-effect instruction. The skip flag is set or reset according to the instruction executed.

The carry flag is set to 1 if an addition instruction (ACSC) generates a carry from bit 3 of the ALU. If no carry is generated, the flag is reset to zero. The SC instruction sets the carry flag and the RC instruction resets it.

When a RESET is input, the SK1 and SK0 flags are cleared to zero and the contents of the carry flag are undefined.

System Clock Generator

The system clock generator consists of a RC oscillator (μPD7556/56A), a ceramic resonator (μPD7566/66A), a 1/2 frequency divider, standby modes (STOP/HALT), and control circuit. Figure 8 is a circuit diagram of the system clock generator for the μPD7556/56A.

In the μPD7556/56A, the RC oscillator operates with a single external resistor connected across CL1 and CL2 (the capacitor C is incorporated). When the RC oscillator is not used, external clock pulses can be input by the CL1 pin. In this case, the RC oscillator functions as an inverting buffer. The output from the RC oscillator serves as the system clock (CL) which is then divided by two and used as the CPU clock (φ).

The standby mode control circuit is made up of a STOP flip-flop and a HALT flip-flop. The STOP instruction sets the STOP flip-flop and stops the system clock supply. This flip-flop also stops the RC oscillator. The STOP flip-flop is reset by the standby release signal that becomes active when one of the test requests flags is set or at the falling edge of the RESET signal. When the STOP flip-flop is reset, the RC oscillator resumes operation and supplies the system clock.

The HALT and STOP instructions and RESET HIGH set the HALT flip-flop which disables signals from going to the 1/2 frequency divider that generates the CPU clock. Only the CPU clock stops in HALT mode. The HALT flip-flop is reset by the same conditions as the STOP flip-flop.

Figure 9 shows the system clock generator circuit for the μPD7566/66A.

On the μPD7566/66A, the ceramic oscillator operates with a ceramic resonator connected across CL1 and CL2. The output from the ceramic oscillator is used as the system clock (CL); it is divided by two to produce the CPU clock (φ).

The standby mode control circuit is made up of a STOP flip-flop and a HALT flip-flop. The STOP instruction sets the STOP flip-flop and stops the ceramic oscillation, thus stopping the supply for all clocks. The STOP flip-flop is reset by the RESET signal (high level) and restarts ceramic oscillation. The supply of each clock resumes when RESET goes low.

The HALT instruction sets the HALT flip-flop which disables signals from going to the 1/2 frequency divider that generates the CPU clock. Only the CPU clock stops in HALT mode. The HALT flip-flop is reset by the HALT RELEASE signal (activated by setting at least one test request flag) or the falling edge of RESET, resuming supply of the CPU clock.

The HALT flip-flop is also set when RESET is active (high level). At power on reset operation, the rising edge of RESET starts ceramic oscillation; however, some time is required to achieve stable oscillation. To prevent the unstable clock from operating the CPU, the HALT flip-flop is set and the CPU clock is stopped while RESET is high. Accordingly, the high-level width of RESET must be more than the required stable time for the ceramic resonator.

Figure 2. Program Memory Map

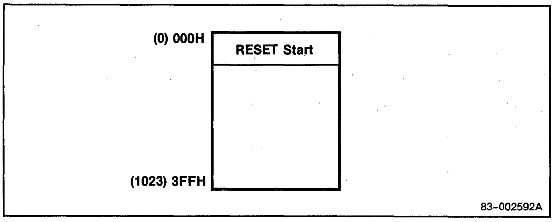


Figure 3. Configuration of General Purpose Registers

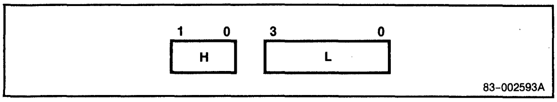


Figure 4. Data Memory Map

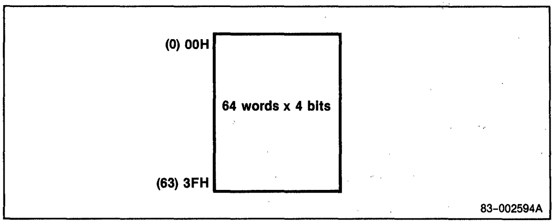


Figure 5. Call Instruction Storage to Stack

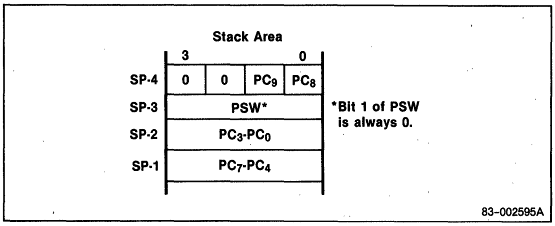


Figure 6. Configuration of the Accumulator

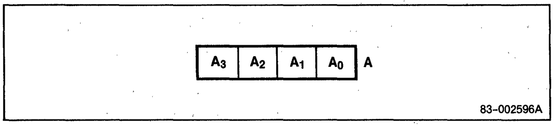


Figure 7. Configuration of the Program Status Word

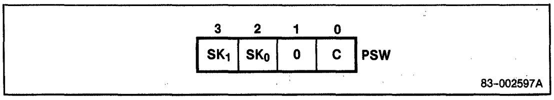


Figure 8. System Clock Generator for μPD7556/56A

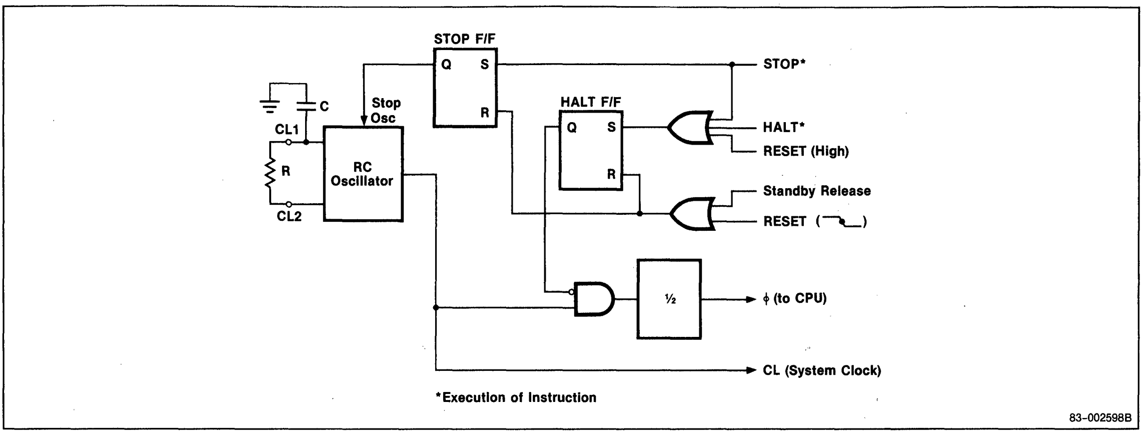
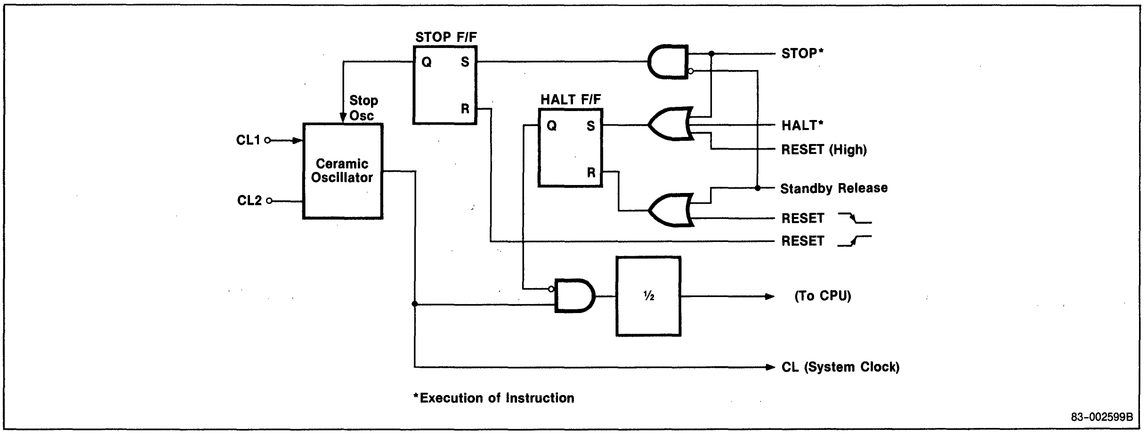


Figure 9. System Clock Generator for μPD7566/66A



Clock Control Circuit

The clock control circuit consists of a 2-bit clock mode register (bits CM1 and CM2), prescalers 1, 2, and 3, and a multiplexer. It takes the output of the system clock generator (CL) and event pulses ($P0_0$). It also selects the clock source and prescaler according to the setting in the clock mode register and supplies the timer/event counter with count pulses. Figure 10 shows the clock control circuit.

Table 2 lists the codes set in the clock mode register by the OPL instruction to specify the count pulse frequency.

When you set the clock mode register with the OPL instruction, clear bit 0 of the accumulator (corresponding to bit CM0 of the EVAKIT-7500 or μPD7500H during emulation).

Figure 10. ***Clock Control Circuit***

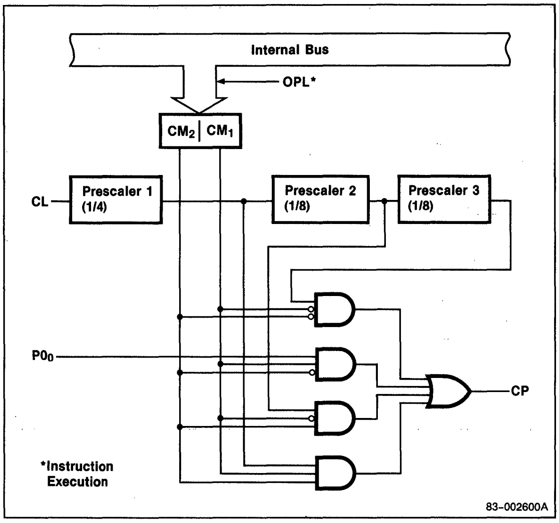


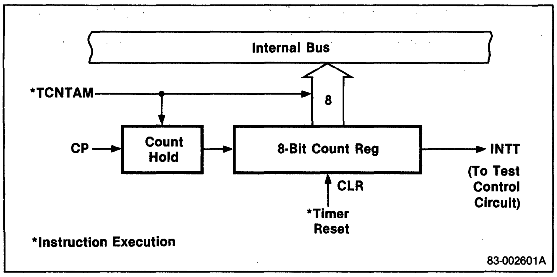
Table 2. Selecting the Count Pulse Frequency

CM2	CM1	Frequency Selected
0	0	CL/256
0	1	$P0_0$
1	0	CL/32
1	1	CL/4

Timer/Event Counter

The timer/event counter is a binary 8-bit up-counter which is incremented each time a count pulse is input. The TIMER instruction or a RESET signal clears it to 00H. When an overflow occurs, the counter is reset from FFH to 00H. Figure 11 shows the inputs and outputs of the counter.

Figure 11. Timer/Event Counter



Test Control Circuit

The μPD7566/66A has two test sources, as shown in table 3.

The test control circuit consists of two test request flags INTT RQF and INT0 RQF) set by the two test sources, the SM_3 flag which determines whether INT0 is enabled, and a test request flag control circuit that checks the contents of each test request flag by executing an SKI instruction and resetting the flags.

The OPL instruction (L = FH, corresponding to A_3) sets the SM_3 flag. INT0 is enabled when $SM_3 = 1$.

Table 3. μPD7556/56A/66/66A Test Sources

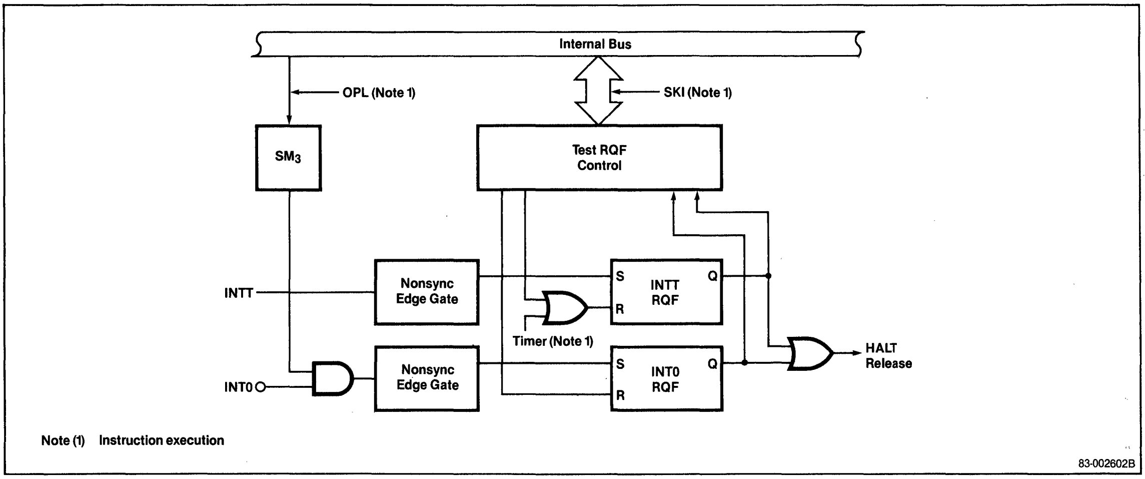
Source	Function	Location	Request Flag
INTT	Overflow in timer/ event counter	Internal	INTT RQF
INT0	Test request signal from $P0_0$ pin	External	INT0 RQF

The request flag INTT RQF is set when a timer overflow occurs in the timer event counter. The SKI or TIMER instruction resets it.

The logical sum of the outputs from the test request flags releases HALT mode. The mode is released when one or both flags are set. Both flags and SM_3 are reset when the RESET signal is input. After reset, signal input to the INT0 pin is inhibited as the initial condition.

Figure 12 is a block diagram of the test control circuit.

Figure 12. Test Control Circuit Block Diagram



Standby Modes

The μPD7556/56A/66/66A has two standby modes to reduce power consumption while the program is in the wait state. The STOP and HALT instructions set these modes.

When the program enters a standby mode, program execution stops and the contents of all registers and data memory immediately before the program entered standby mode are retained. The timer can operate even in HALT mode.

The RESET signal or STANDBY release signal[1] releases STOP mode. HALT mode is released when one or both of the test request flags are set, or when the RESET signal is input. The program cannot enter a standby mode when a test request is being set, even if the STOP or HALT command is executed.

If there is some uncertainty about the state of the test request flags, execute the SKI instruction to reset them so the program can enter standby mode.

Table 4 compares STOP and HALT modes. The main difference is that STOP mode stops the system clock and HALT does not. Oscillation stops during STOP mode. The power consumed by the oscillator is the difference between the two modes. In STOP mode, data memory can be retained with a lower supply voltage.

Note: (1) Standby release signal for μPD7556/56A only.

Table 4. STOP and HALT Modes

Mode	CL	ϕ	PO_0	CPU	Timer	Released by
STOP	x	x	o	x	Δ	RESET input INTT RQF, INT0 RQF (μPD7556/56A only)
HALT	o	x	o	x	o	INTT RQF INT0 RQF RESET input

Notes:

(1) o: operates. x: stops.
Δ: operates depending on clock source. μPD7556/56A; if external clock is used, STOP instruction will not stop CL. In this case STOP mode acts as HALT mode.

Power-on Reset Circuit

Figure 13 shows a circuit example of the power-on reset circuit using a resistor and a capacitor. This is the simplest reset control circuit. Figure 14 shows the circuit with a pull-down resistor internally connected to RESET as a mask option.

μPD7556/56A/66/66A Applications

Figures 15-18 show examples of application circuits for the μPD7556/56A/66/66A.

Table 5 compares the features of the low-end products of the 7500 series devices.

Figure 13. Power-on Reset Circuit

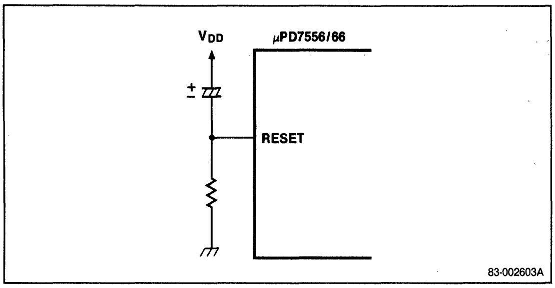


Figure 14. Power-on Reset Circuit with Pull-down Resistor

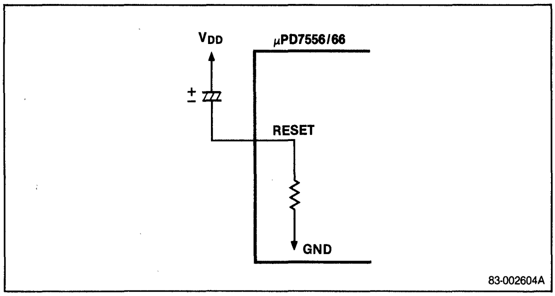
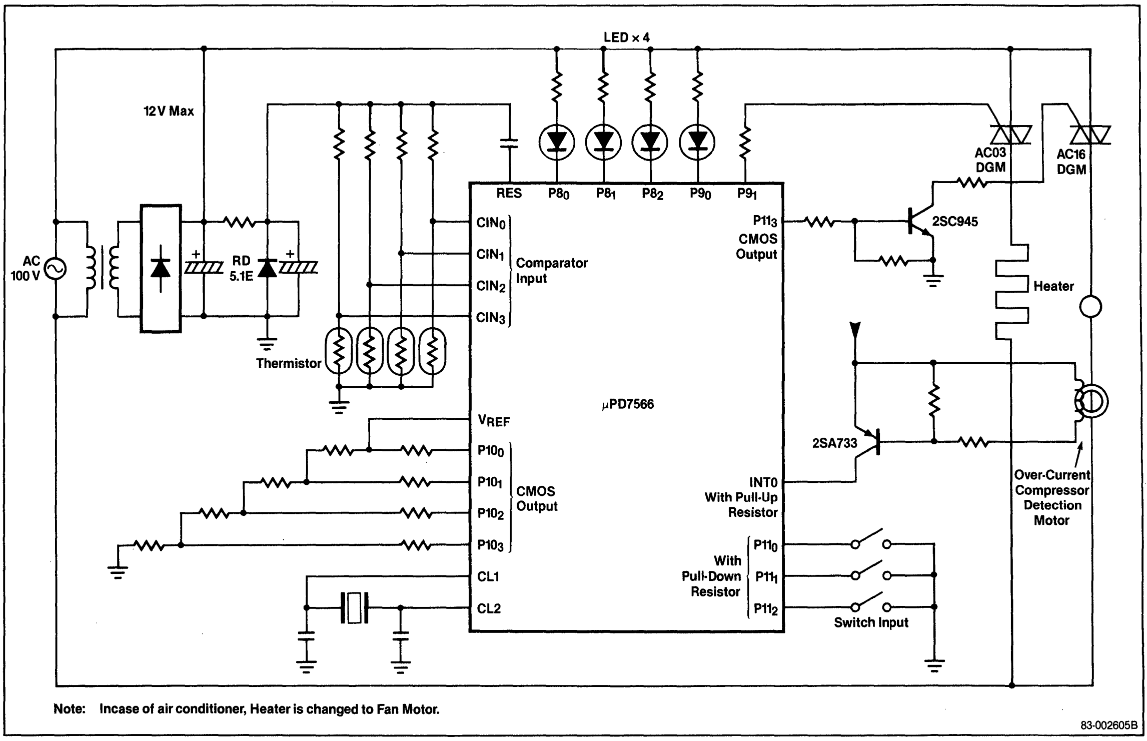


Table 5. Product Comparison

Item		µPD7554/54A	µPD7564/64A	µPD7556/56A	µPD7566/66A
Instruction cycle/system clock (5 V)	RC	4 µs/ 500 kHz		4 µs/ 500 kHz	
	External	2.86 µs/ 700 kHz		2.86 µs/ 700 kHz	
	Ceramic		3 µs/ 660 kHz		3 µs/ 660 kHz
Instruction set		47	47	45	45
ROM		1024 x 8	1024 x 8	1024 x 8	1024 x 8
RAM		64 x 4	64 x 4	64 x 4	64 x 4
I/O port total		16 (max)	15	20 (max)	19
Port 0		$P0_0$-$P0_3$	$P0_0$-$P0_3$	$P0_0$-$P0_1$	$P0_0$-$P0_1$
Port 1				$P1_0$-$P1_3$	$P0_1$-$P0_3$
Port 8		$P8_0$-$P8_2$ $P8_2$/CL2	$P8_0$-$P8_2$	$P8_0$-$P8_2$ $P8_3$/CL2	$P8_0$-$P8_2$
Port 9				$P9_0$-$P9_1$	$P9_0$-$P9_1$
Port 10		$P10_0$-$P10_3$	$P10_0$-$P10_3$	$P10_0$-$P10_3$	$P10_0$-$P10_3$
Port 11		$P11_0$-$P11_3$	$P11_0$-$P11_3$	$P11_0$-$P11_3$	$P11_0$-$P11_3$
Timer/Event counter		8-bit	8-bit	8-bit	8-bit
Serial interface		8-bit	8-bit		
Comparator				4-channel	4-channel
Process		CMOS	CMOS	CMOS	CMOS
Package		20-pin plastic SOP 20-pin shrink DIP	20-pin plastic SOP 20-pin shrink DIP	24-pin plastic SOP 24-pin shrink DIP	24-pin plastic SOP 24-pin shrink DIP

Figure 15. Refrigerator or Air Conditioner Circuitry



Note: Incase of air conditioner, Heater is changed to Fan Motor.

Figure 16. Rice Cooker Circuitry

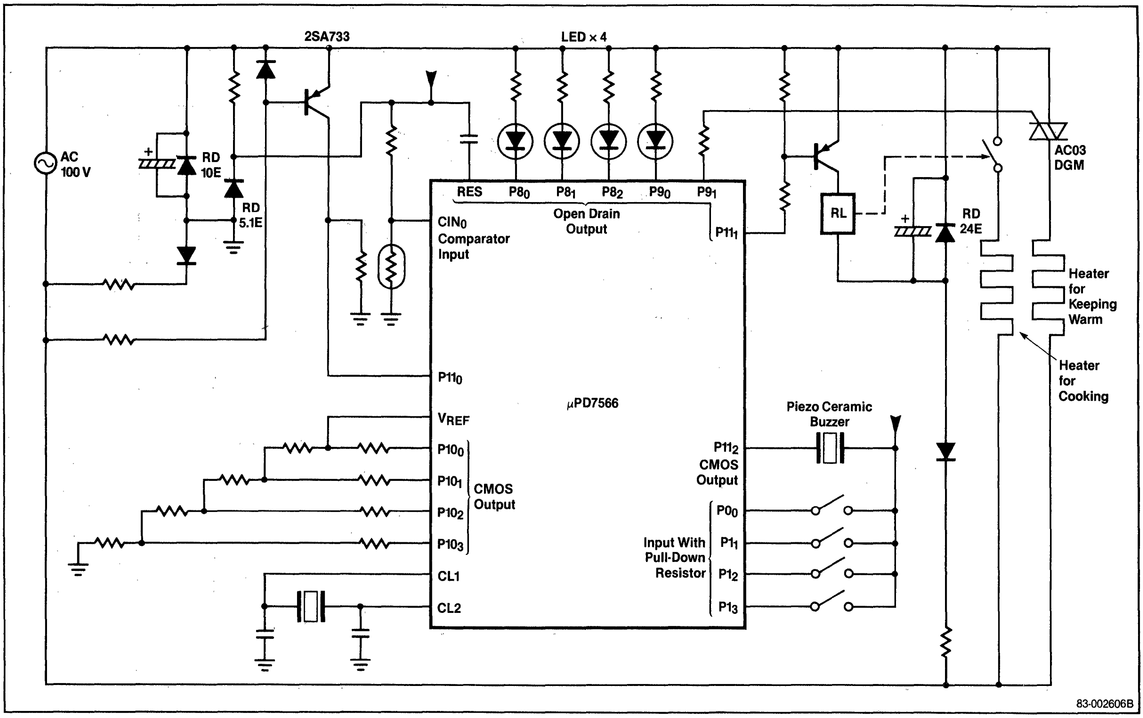


Figure 17. Washing Machine Circuitry

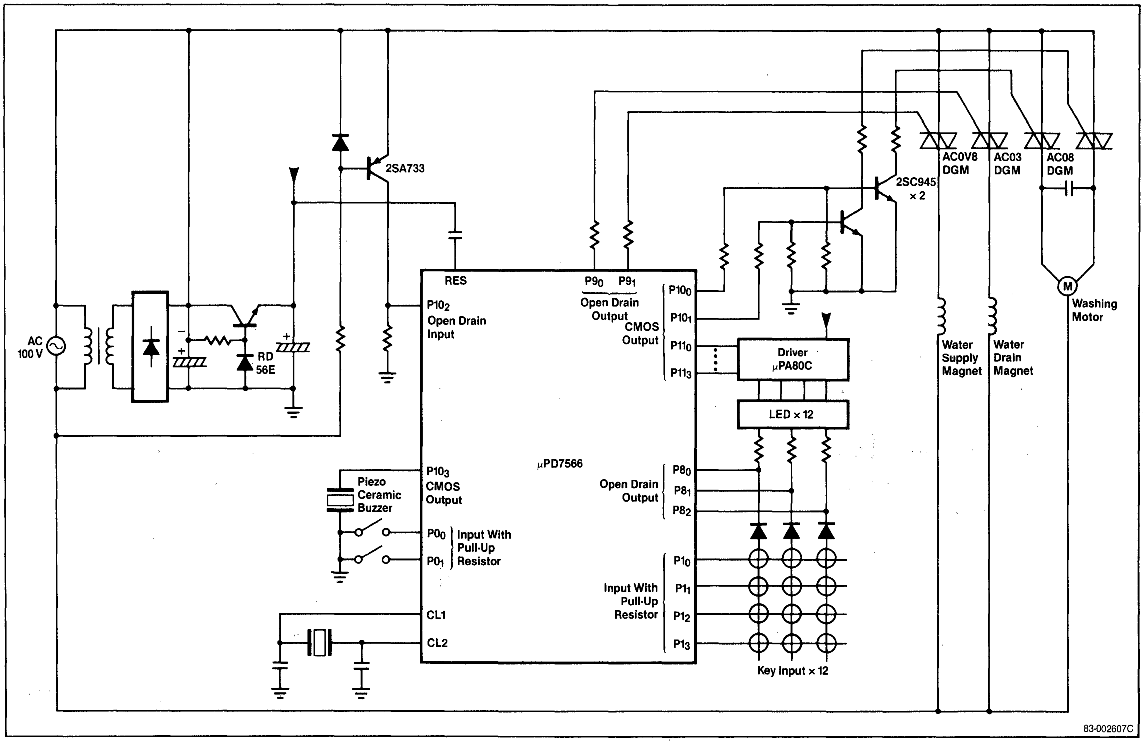
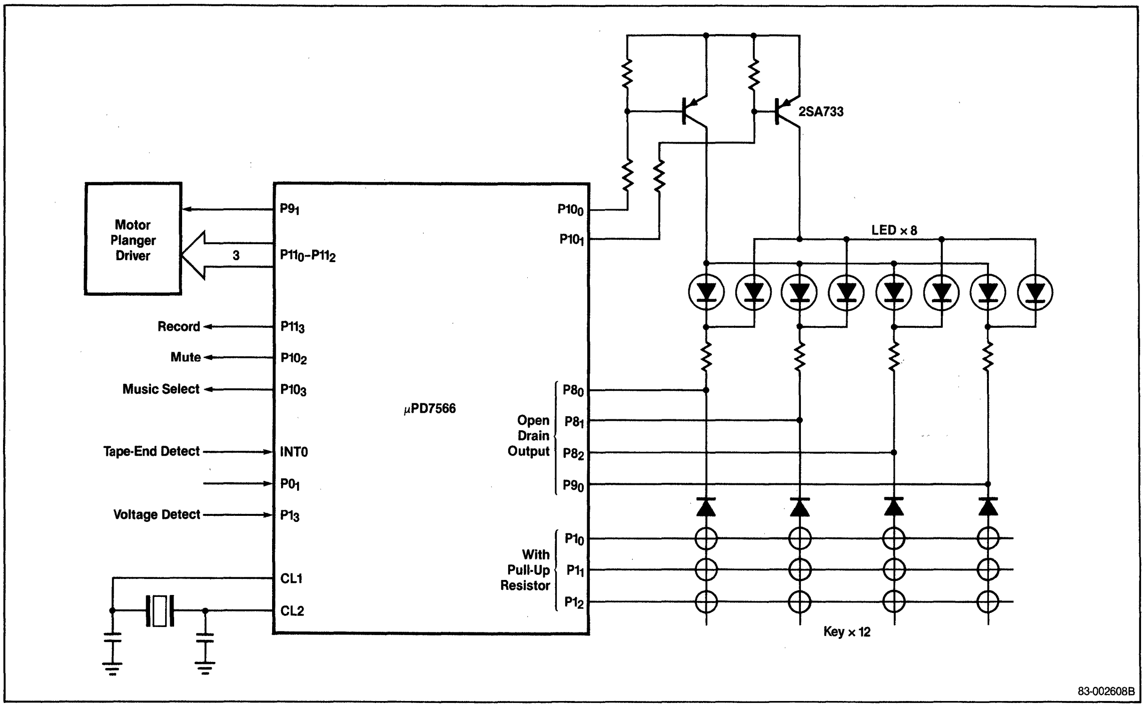


Figure 18. Tape Deck Controller Circuitry



ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C
Power supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_I Except ports 10, 11	−0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 1) (Note 2) μPD7556A/66A (Note 2)	−0.3 to V_{DD} +0.3 V −0.3 to +13 V −0.3 to +11 V
Output voltage, V_O Except ports 8, 10, 11	−0.3 to V_{DD} +0.3 V
Ports 8, 10, 11 (Note 1) (Note 2) μPD7556A/66A (Note 2)	−0.3 to V_{DD} +0.3 V −0.3 to +13 V −0.3 V to +11 V
Output current, high I_{OH} One port	−5 mA
All output ports, total	−15 mA
Output current, low I_{OL} $P0_1$, $P0_2$	5 mA
Ports 10, 11	15 mA
Ports 8, 9	30 mA
All ports, total	100 mA
Power dissipation, P_D (T_A = +70°C) Shrink DIP	480 mW
SOP	250 mW

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Notes:

(1) CMOS I/O or N-channel open drain + internal pull-up resistor.

(2) N-channel open drain I/O.

Capacitance

T_A = 25°C, V_{DD} = V_{SS} = 0 V; f = 1 MHz;
Unmeasured pins returned to V_{SS}

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I			15	pF	$P0_0$-$P0_1$; $P1_0$-$P1_3$; CIN_0-CIN_3;
Output capacitance	C_O			35	pF	Port 8
I/O capacitance	$C_{I/O}$			35	pF	Ports 8-11

DC Characteristics 1; V_{DD} = 2.5 to 3.3 V; μPD7556/56A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.8 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.3		V_{DD}	V	
Input high voltage ports 10, 11	V_{IH3}	0.8 V_{DD}		12 (Note 1); 9 (Note 2)	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1	V_{IL1}	0		0.2 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.3	V	
Input leakage current except CL1	I_{LI1}	−3		3	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11	I_{LI3}			10 (Note 1)	μA	V_I = 12 V
				10 (Note 2)	μA	V_I = 9 V
Output voltage high ports 8-11	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −80 μA
Output voltage low ports 10, 11	V_{OL}			0.5	V	I_{OL} = 350 μA
Output voltage low ports 8-9	V_{OL}			0.5	V	I_{OL} = 500 μA
Output leakage current	I_{LO1}	−3		3	μA	0 V ≤ V_O ≤ V_{DD}
Output leakage current ports 8-11	I_{LO2}			10 (Notes 1, 2)	μA	V_O = 12 V μPD7556; V_O = 9 V μPD7556A
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation; R oscillation (Note 3)	I_{DD1}		55	180	μA	V_{DD} = 3 V ±0.3 V; R = 150 kΩ ±2%
			40	150	μA	V_{DD} = 2.5 V; R = 150 kΩ ±2%
Supply current, HALT mode; R oscillation (Note 3)	I_{DD2}		25	80	μA	V_{DD} = 3 V ±0.3 V; R = 150 kΩ ±2%
			18	60	μA	V_{DD} = 2.5 V; R = 150 kΩ ±2%
Supply current, STOP mode (Note 3)	I_{DD3}		0.1	5	μA	
Supply current, data retention mode (Note 3)	I_{DDDR}		0.1	5	μA	V_{DDDR} = 2.0 V
Pull-up/down resistance, ports 0-1, RESET	RP1	23.5	47	70.5	kΩ	
Pull-up resistance, ports 10, 11	RP2	7.5	15	22.5	kΩ	

Notes:

(1) N-channel, open drain I/O ports, μPD7556.

(2) N-channel, open drain I/O ports, μPD7556A.

(3) Current in built-in pull-up/down resistors excluded.

DC Characteristics 2; V_{DD} = 2.7 to 6.0 V; µPD7556/56A/66/66A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.7 V_{DD}		V_{DD}	V	
Input high voltage CL1 (Note 2)	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	
Input high voltage ports 10, 11	V_{IH3}	0.7 V_{DD}		12 (Note 1); 9 (Note 2)	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1 (Note 3)	V_{IL1}	0		0.3 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.5	V	
Input leakage current except CL1 (Note 3)	I_{LI1}	−3		3	µA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	µA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11 (Note 4)	I_{LI3}			10	µA	V_I = 9 V µPD7556A/66A; or 12 V
Output voltage high $P0_1$, $P0_2$, ports 8-11	V_{OH}	V_{DD} − 2.0			V	V_{DD} = 4.5 to 6.0 V; I_{OH} = −1 mA
	V_{OH}	V_{DD} − 1.0			V	V_{DD} = 2.7 V; I_{OH} = −100 µA
Output voltage low ports 10, 11	V_{OL}			0.4	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 1.6 mA
				2.0	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 10 mA
				0.5	V	V_{DD} = 2.7 V; I_{OL} = 400 µA
Output voltage low ports 8-9	V_{OL}			2.0	V	V_{DD} = 4.5 to 6.0 V; I_{OL} = 15 mA
				0.5	V	V_{DD} = 2.7 V; I_{OL} = 600 µA
Output leakage current	I_{LO1}	−3		3	µA	0 V ≤ V_O ≤ V_{DD}
Output leakage current, port 8-11 (Note 4)	I_{LO2}			10	µA	V_O = 12 V µPD7556/66, V_O = 9 V µPD7556A/66A
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation; ceramic oscillation (Notes 3, 5)	I_{DD1}		650	2200	µA	V_{DD} = 5 V ±0.5 V; f_{CC} = 700 kHz
			120	360	µA	V_{DD} = 3 V ±10%; f_{CC} = 300 kHz
Supply current, normal operation; R oscillation (Note 3)	I_{DD1}		270	900	µA	V_{DD} = 5 V ±0.5 V; R = 56 kΩ ±2%
			80	240	µA	V_{DD} = 3 V ±10%; R = 100 kΩ ±2%
Supply current, HALT mode; ceramic oscillation (Note 3, 5)	I_{DD2}		450	1500	µA	V_{DD} = 5 V ±0.5 V; f_{CC} = 700 kHz
			65	200	µA	V_{DD} = 3.0 V ±10%; f_{CC} = 300 kHz
Supply current, HALT mode; R oscillation (Note 3)	I_{DD2}		120	400	µA	V_{DD} = 5 V ±0.5 V; R = 56 kΩ ±2%
			35	110	µA	V_{DD} = 3 V ±10%; R = 100 kΩ ±2%
Supply current, STOP mode (Note 3)	I_{DD3}		0.1	10	µA	V_{DD} = 5 V ±0.5 V
			0.1	5	µA	V_{DD} = 3 V ±10%
Supply current, data retention mode (Note 3)	I_{DDDR}		0.1	5	µA	V_{DDDR} = 2.0 V
Pull-up/down resistance, port 0, RESET	RP1	23.5	47	70.5	kΩ	
Pull-up resistance, ports 8-11	RP2	7.5	15	22.5	kΩ	

Notes:

(1) µPD7556/66.

(2) µPD7556A/66A.

(3) Current in built-in pull-up/down resistors excluded.

(4) N-channel, open-drain I/O ports.

(5) µPD7566/66A.

DC Characteristics 3; V_{DD} = 2.0 to 3.3 V; μPD7556A only

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.8 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.2		V_{DD}	V	
Input high voltage ports 10, 11	V_{IH3}	0.85 V_{DD}		9	V	
Input high voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1	V_{IL1}	0		0.15 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.2	V	
Input leakage current except CL1	I_{LI1}	−3		3	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current CL1	I_{LI2}	−10		10	μA	0 V ≤ V_I ≤ V_{DD}
Input leakage current ports 10, 11 (Note 1)	I_{LI3}			10	μA	V_I = 9 V
Output voltage high $P0_1$, $P0_2$, ports 8-11	V_{OH}	V_{DD} − 1.0			V	I_{OH} = −70 μA
Output voltage low $P0_1$, $P0_2$, ports 10, 11	V_{OL}			0.5	V	Ports 10, 11: I_{OL} = 300 μA
Output voltage low ports 8, 9	V_{OL}			0.5	V	I_{OL} = 400 μA
Output leakage current (Note 2)	I_{L01}	−3		3	μA	0 V ≤ V_O ≤ V_{DD}
Output leakage current ports 8-11 (Note 1)	I_{L02}			10	μA	V_O = 9 V
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation; R oscillation (Note 2)	I_{DD1}		38	130	μA	V_{DD} = 3.0 ±10%; R = 240 kΩ ±2%
			20	70	μA	V_{DD} = 2.0 V; R = 240 kΩ ±2%
Supply current, HALT mode; R oscillation (Note 2)	I_{DD2}		17	60	μA	V_{DD} = 3 V ±10%; R = 240 kΩ ±2%
			8	25	μA	V_{DD} = 2 V; R = 240 kΩ ±2%
Supply current, STOP mode (Note 2)	I_{DD3}		0.1	5	μA	
Supply current, data retention mode (Note 2)	I_{DDDR}		0.1	5	μA	V_{DDDR} = 2.0 V
Pull-up/down resistance, port 0, RESET	RP1	23.5	47	70.5	kΩ	
Pull-up resistance, ports 8-11	RP2	7.5	15	22.5	kΩ	

Notes:

(1) N-channel, open-drain I/O ports.

(2) Current in built-in pull-up/down resistors excluded.

Comparator

T_A = −10 to +70°C; V_{DD} = 3.0 to 6.0 V, GND = 0 V

		μPD7556/56A			μPD7566/66A				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
Input voltage range	V_{CIN}/V_{REF}	0		V_{DD}	0		V_{DD}	V	All comparators
Response time	T_{COMP}	2		4	2		4	MC(Note 1)	All comparators
Input voltage resolution	ΔV_{CIN}			100			100	mV	All comparators
			10	50		10	50	mV	All comparators; V_{DD} = 5 V ± 0.5 V
Input leakage current	I_{CIN}/I_{REF}	−3		3	−3		3	μA	All comparators
V_{REF} bias resistance (R1, R2)	BIAS		100			100		kΩ	(R1 = R2) typically
Comparator circuit current (Note 2)	I_{DDCMP}		50			50		μA	Comparator: V_{DD} = 5 V ± 0.5 V

Notes:

(1) Machine cycle.

(2) Excluding current through bias resistor.

AC Characteristics 1; V_{DD} = 2.5 to 3.3 V; μPD7556/56A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency, CL1, CL2	f_{CC}	140	180	220	kHz	R = 150 kΩ ±2%
		140	175	210	kHz	V_{DD} = 2.5 V; R = 150 kΩ ±2%
External clock frequency, CL1	f_C	10		250	kHz	50% duty
System clock rise time, CL1	t_{CR}			200	ns	
System clock fall time, CL1	t_{CF}			200	ns	
System clock pulse width, high	t_{CH}	2		50	μs	
System clock pulse width, low	t_{CL}	2		50	μs	
External clock frequency (PO_0)	f_{P00}	0		250	kHz	50% duty
PO_0 rise time	t_{CRPO}			200	ns	
PO_0 fall time	t_{CFPO}			200	ns	
PO_0 pulse width, high	t_{POOH}	2			μs	
PO_0 pulse width, low	t_{POOL}	2			μs	
INT0 high time	t_{IOH}	30			μs	
INT0 low time	t_{IOL}	30			μs	
RESET high time	t_{RSH}	30			μs	
RESET low time	t_{RSL}	30			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	

AC Characteristics 2; V_{DD} = 2.7 to 6.0 V; μPD7556/56A/66/66A

T_A = −10 to +70°C; GND = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency (Note 1)	f_{CC}	400	500	600	kHz	V_{DD} = 4.5 to 6.0 V; R = 56 kΩ ±2%
		200	250	300	kHz	V_{DD} = 3 V ±10%; R = 100 kΩ ±2%
External clock frequency, CL1	f_C	10		710	kHz	V_{DD} = 4.5 to 6.0 V; 50% duty
		10		350	kHz	V_{DD} = 2.7 V; 50% duty
System clock oscillation frequency (Note 2)	f_{CC}	290	700	710	kHz	V_{DD} = 4.5 to 6.0V
		290	500	510	kHz	V_{DD} = 4.0 to 6.0 V
		290	400	410	kHz	V_{DD} = 3.5 to 6.0 V
		290	300	310	kHz	V_{DD} = 2.7 to 6.0 V
Oscillator setup (Note 2)	t_{OS}	20			ms	OS stabilization time after minimum of operating voltage reached
System clock rise time, CL1	t_{CR}			200	ns	
System clock fall time, CL1	t_{CF}			200	ns	
System clock pulse width	t_{CH}	0.7		50	μs	
System clock pulse width, CL1	t_{CL}	1.45		50	μs	V_{DD} = 2.7 V
External clock frequency ($P0_0$)	f_{P00}	0		710	kHz	V_{DD} = 4.5 to 6.0 V; 50% duty
		0		350	kHz	V_{DD} = 2.7 V; 50% duty
$P0_0$ rise time	t_{CRP0}			200	ns	
$P0_0$ fall time	t_{CFP0}			200	ns	
$P0_0$ pulse width, high	t_{P00H}	0.7			μs	V_{DD} = 4.5 to 6.0 V
$P0_0$ pulse width, low	t_{P00L}	1.45			μs	V_{DD} = 2.7 V
INT0 high time	t_{IOH}	10			μs	
INT0 low time	t_{IOL}	10			μs	
RESET high time	t_{RSH}	10			μs	
RESET low time	t_{RSL}	10			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	

Notes:

(1) μPD7556/56A.

(2) μPD7566/66A.

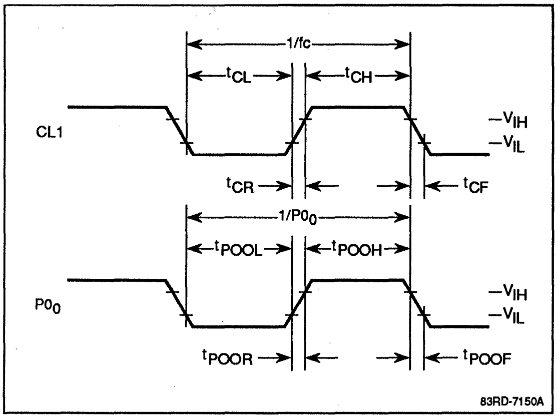
AC Characteristics 3; V_{DD} = 2.0 to 3.3 V; μPD7556A only

T_A = −10 to +70°C; GND = 0 V

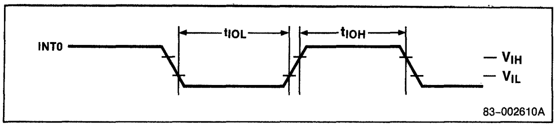
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency	f_{CC}	65	120	145	kHz	R = 240 kΩ ±2%
System clock oscillation frequency, CL1, CL2	f_{CC}	65	100	130	kHz	V_{DD} = 2.0 V; R = 240 kΩ ±2%
External clock frequency, CL1	f_C	10		150	kHz	50% duty
System clock rise time, CL1	t_{CR}			200	ns	
System clock fall time, CL1	t_{CF}			200	ns	
System clock pulse width, high	t_{CH}	3.3		50	μs	
System clock pulse width, low	t_{CL}	3.3		50	μs	
External clock frequency ($P0_0$)	f_{P00}	0		150	kHz	50% duty
$P0_0$ rise time	t_{CRP0}			200	ns	
$P0_0$ fall time	t_{CFP0}			200	ns	
$P0_0$ pulse width, high	t_{P00H}	3.3			μs	
$P0_0$ pulse width, low	t_{P00L}	3.3			μs	
INT0 high time	t_{IOH}	50			μs	
INT0 low time	t_{IOL}	50			μs	
RESET high time	t_{RSH}	50			μs	
RESET low time	t_{RSL}	50			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	

Timing Waveforms

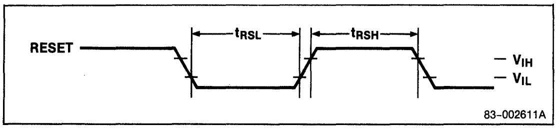
Clocks



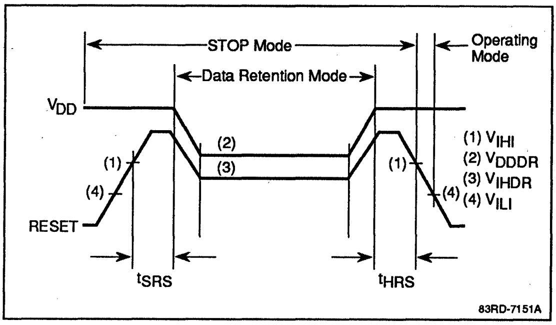
External Interrupt



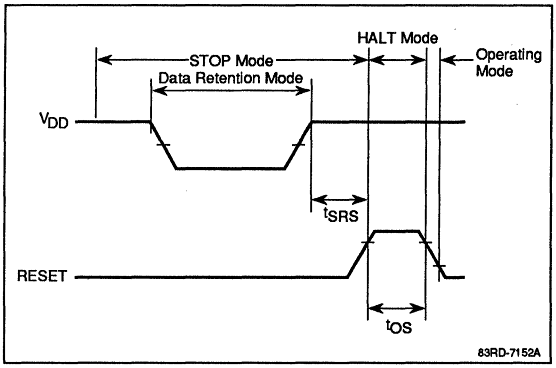
Reset



Data Retention Mode, μPD7556/56A



Data Retention Mode, μPD7566/66A



NEC
NEC Electronics Inc.

μPD75P56/P66 4-Bit, Single-Chip, One-Time Programmable (OTP) CMOS Microcomputers With Comparator

Description

The μPD75P56 and μPD75P66 are 1024 x 8-bit, on-chip, one-time programmable (OTP) ROM versions of the mask ROMs, μPD7556 and μPD7566. They have the same functions as, and are pin-compatible with, their mask ROMs. Because of their programming capabilities, the μPD75P56/P66 are suitable for evaluation and small lot production for system development. Their unique features will be described in this data sheet. For information about the base part μPD7556/66, please refer to its data sheet.

Features

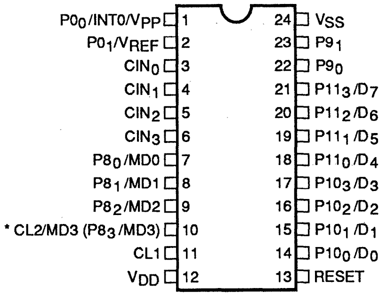
- □ 45 instructions (subset of μPD7500 set B)
- □ Instruction cycle:
  - — External clock (μPD75P56): 2.86 μs/700 kHz, 5 V
  - — RC oscillator (μPD75P56); 4 μs/500 kHz, 5 V
  - — Ceramic resonator (μPD75P66): 2.86 μs/700 kHz, 5 V
- □ Program memory (ROM) of 1024 x 8 bits
- □ Data memory (RAM) of 64 x 4 bits
- □ 8-bit timer/event counter
- □ 4-channel comparator
- □ I/O lines: 20-μPD75P56; 19-μPD75P66
- □ One-time programmable
  - — No mask option available
- □ Data memory retention at low supply voltage
- □ Standby (STOP/HALT) functions
- □ CMOS technology
- □ Low-power consumption
- □ Single power supply
  - — 4.5 to 6.0 V, normal operation

Ordering Information

Part Number	Package Type
μPD75P56CS-001	24-pin plastic shrink DIP (OTP)
μPD75P56CS-012	
μPD75P66CS-001	
μPD75P56G-511	24-pin plastic SOP (OTP)
μPD75P56G-512	
μPD75P66G-511	

Pin Configuration

24-Pin Plastic Shrink DIP or SOP (OTP)



* $P8_3$ can be selected when using external clock on μPD75P56 only. CL2 can be selected when using RC oscillator.

49NR-649A

Pin Identification

Symbol	Function
$P0_0$/INT0/V_{PP}	2-bit input port 0/testable input pin/ programming voltage supply pin for program memory write/verify
V_{REF}	Comparator reference voltage input pin
CIN_0 CIN_1 CIN_2 CIN_3	4-channel comparator inputs
$P8_0$-$P8_2$/MD_0-MD2 CL2/MD3 ($P8_3$/MD3)	4-bit output port 8/OTP operation mode/ connection for RC oscillator or ceramic resonator (No $P8_3$ on μPD75P66) (Note 1)
CL1	Connection for ceramic resonator or RC oscillator
V_{DD}	Power supply. 4.5 to 6.0 V (normal); 6.0 V (OTP)
RESET	Reset input pin
$P10_1$-$P10_3$/ D_0-D_3	4-bit I/O port 10/D_0-D_3 during programming write/verify
$P11_0$-$P11_3$/ D_4-D_7	4-bit I/O port 11/ D_4-D_7 during programming write/verify
$P9_0$-$P9_1$	2-bit output port 9
V_{SS}	Ground

Notes:

(1) MD0-MD3 are used as mode selection pins during programming.

PIN FUNCTIONS

$P0_0$/INT0/V_{PP}, V_{REF} (Port 0/Count clock input/Programming/ Comparator reference voltage input)

Two-bit input port 0/count clock input/programming/ comparator reference voltage input. INT0 is an edge-sensitive testable input pin that detects a signal at the rising edge. V_{PP} is the programming supply pin for programming memory write/verify. V_{REF} is the comparator reference voltage input pin. If $P0_0$/INT0/V_{REF} is unused; connect it to ground. The port is in the input state at reset.

CIN_0-CIN_3 (Comparator inputs)

Four-channel comparator inputs. Analog voltage input is compared with the reference voltage (V_{REF}). The comparison requires up to three machine cycles. To obtain the comparison result after changing the voltage applied to the V_{REF} pin by port output (OPL), and connecting the A/D converter via the resistor ladder, execute the input instruction (IP1 or IPL, L = 1) after waiting three machine cycles following the execution of OPL.

$P8_0$-$P8_2$/MD0-MD2, P_8/MD3 or CL2/MD3 (Port 8/Clock input 2/Mode selection for OTP)

Four-bit output port 8. This port can sink 15 mA and interface 12 V. $P8_3$ is a output port on the μPD75P56. On the μPD75P56, CL2 is one of the pins to which a resistor for RC oscillation is connected. On the μPD75P66 CL2 is one of the pins to which a ceramic resonator is connected. If any of $P8_0$-$P8_2$ pins are unused, leave them open. The port is in the high impedance state at reset. MD0-MD3 are used for OTP program memory write and read mode selection. There is no $P8_3$ on μPD75P66.

CL1 (Clock input 1)

On the μPD75P56, CL1 is one of the two pins to which a resistor for RC oscillation is connected. On the μPD75P66, CL1 is one of the two pins to which a ceramic resonator is connected.

V_{DD} (Power supply)

Positive power supply. 4.5 to 6.0 V for normal operation. 6.0 V for program memory write/verify.

RESET (Reset)

System reset input pin (active high). This pin is not internally connected to a pull-down resistor.

$P10_0$-$P10_3$/D_0-D_3 (Port 10/Data I/O)

Four-bit I/O port. This port can sink 10 mA and interface 12 V. If any of these pins are unused, connect them to ground or V_{DD} in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset. D_0-D_3 are four-bit I/O pins for program memory write/verify.

$P11_0$-$P11_3$/D_4-D_7 (Port 11/Data I/O)

Four-bit I/O port. This port can sink 10 mA and interface 12 V. If any of these pins are unused, connect them to ground or V_{DD} in the input state, or leave open in the output state. The port is in the high impedance or high-level output state at reset. D_4-D_7 are four-bit I/O pins for program memory write/verify.

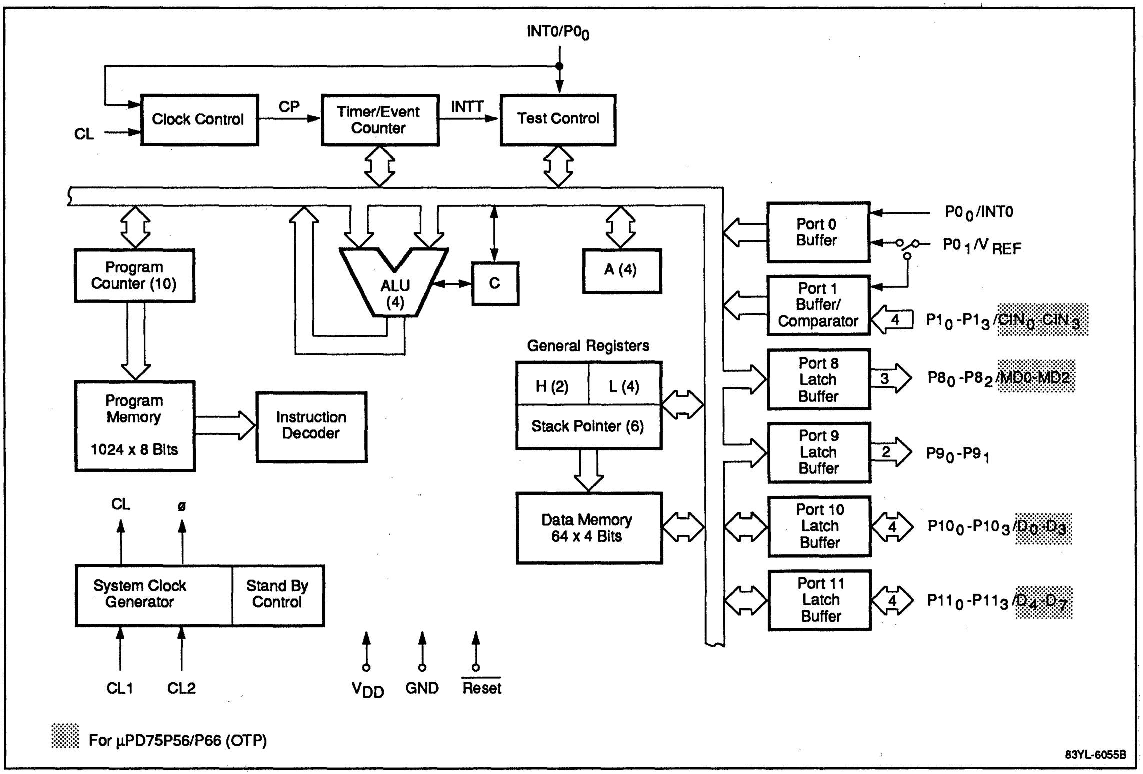
$P9_0$-$P9_1$ (Port 9)

Two-bit output port. This port sinks 15 mA and can interface to 12 V. If either of these pins is unused, leave it open. The port is in the high impedance state at reset.

V_{SS} (Ground)

Ground.

Block Diagram

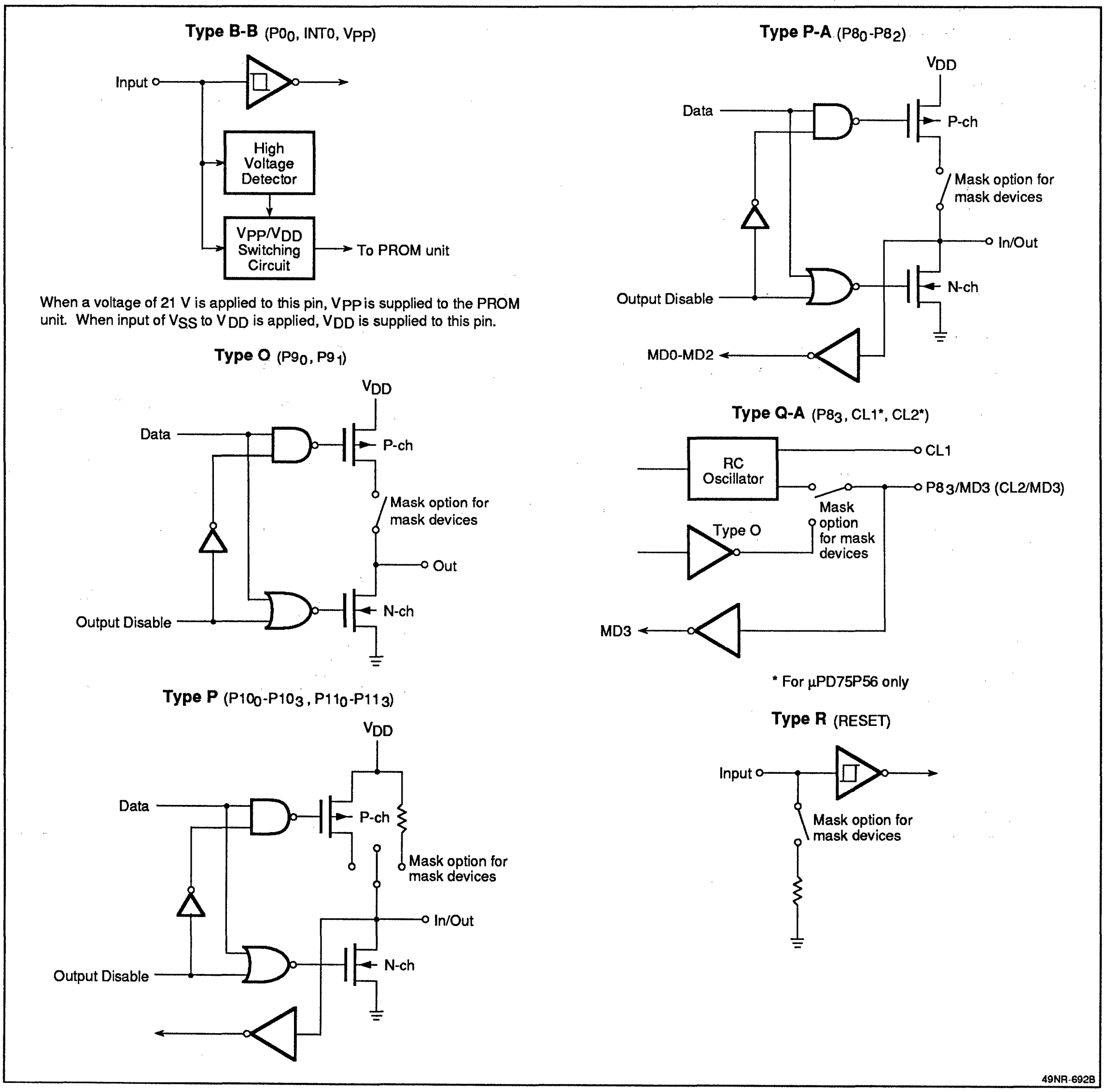


FUNCTIONAL DESCRIPTION

I/O Ports

Figure 1 shows the internal circuits at the I/O ports.

Figure 1. Interface at I/O Ports



Type B-B ($P0_0$, INT0, V_{PP})

When a voltage of 21 V is applied to this pin, V_{PP} is supplied to the PROM unit. When input of V_{SS} to V_{DD} is applied, V_{DD} is supplied to this pin.

Type O ($P9_0$, $P9_1$)

Type P ($P10_0$-$P10_3$, $P11_0$-$P11_3$)

Type P-A ($P8_0$-$P8_2$)

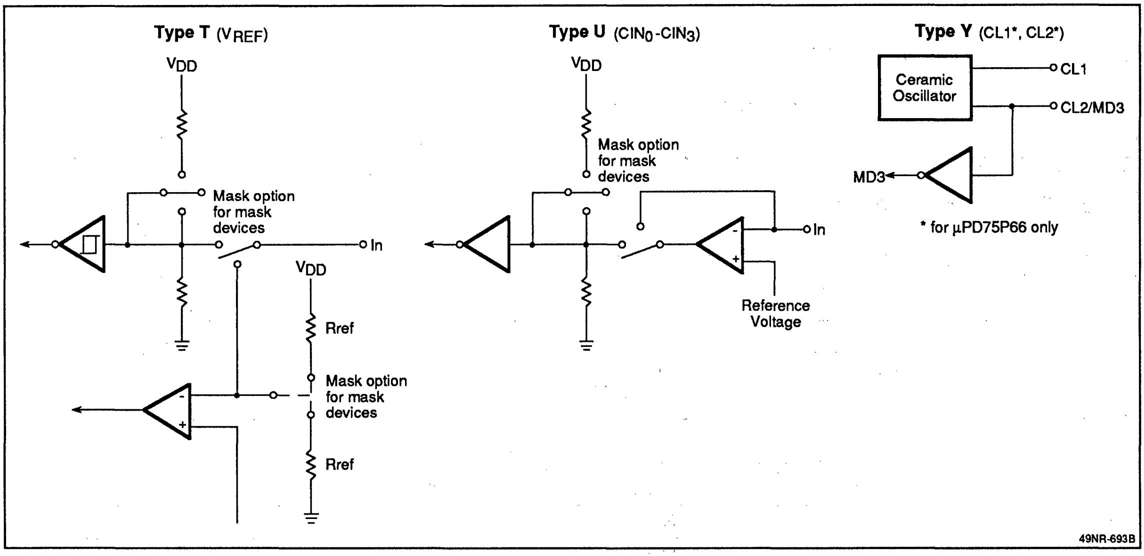
Type Q-A ($P8_3$, CL1*, CL2*)

* For μPD75P56 only

Type R (RESET)

49NR-692B

Figure 1. Interface at I/O Ports (cont)



µPD75P56/P66 and µPD7556/66 Comparisons

Table 1 compares the features of OTP µPD75P56/P66 and their mask ROMs, µPD7556/66.

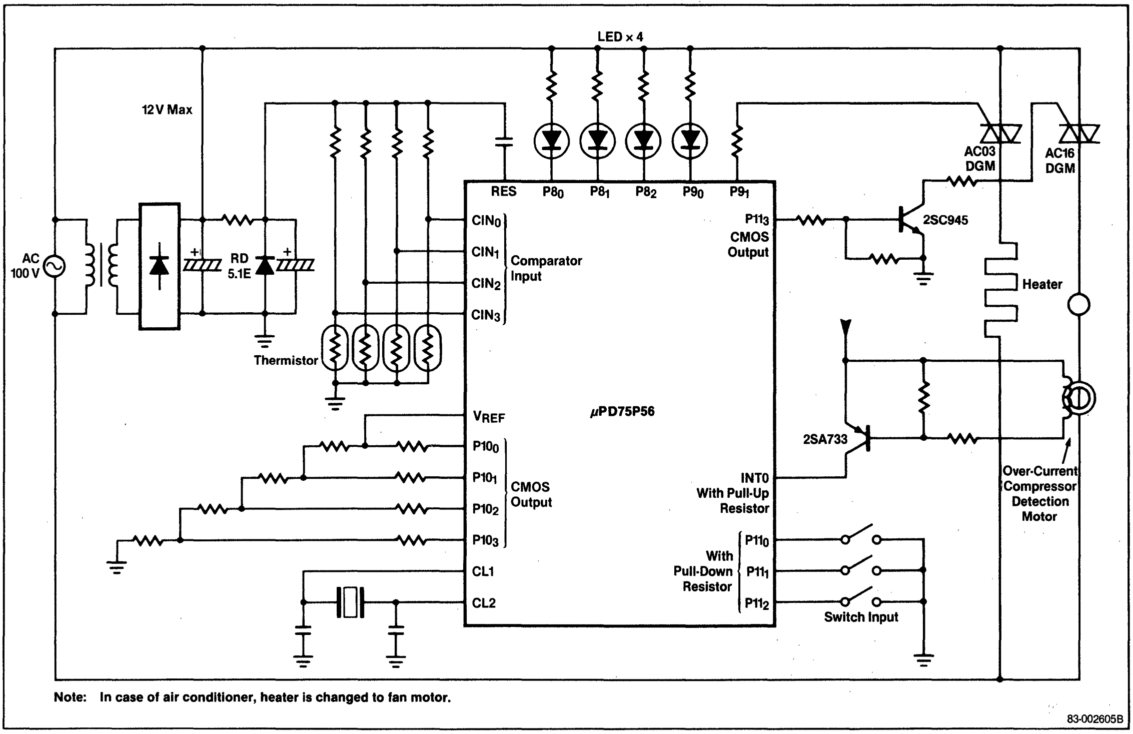
Table 1. Product Comparisons, µPD75P56/P66 and µPD7556/66

Item		µPD75P56 (OTP)	µPD75P66 (OTP)	µPD7556	µPD7566
Instruction cycle/system clock (5 V)	RC	4 µs/500 kHz		4 µs/500 kHz	
	External	2.86 µs/700 kHz		2.86 µs/700 kHz	
	Ceramic		2.86 µs/700 kHz		2.86 µs/700 kHz
Instruction set		45 (set B)	45 (set B)	45 (set B)	45 (set B)
ROM		1024 x 8	1024 x 8	1024 x 8	1024 x 8
RAM		64 x 4	64 x 4	64 x 4	64 x 4
I/O port total		20 (max)	19	20 (max)	19
Port 0		P0$_0$-P0$_1$	P0$_0$-P0$_1$	P0$_0$-P0$_1$	P0$_0$-P0$_1$
Port 1		CIN$_0$-CIN$_3$	CIN$_0$-CIN$_3$	P1$_0$-P1$_3$	P1$_1$-P1$_3$
Port 8		P8$_0$-P8$_2$/MD0-MD2 P8$_3$/MD3	P8$_0$-P8$_2$/MD0-MD2 CL2/MD3	P8$_0$-P8$_2$ P8$_3$/CL2	P8$_0$-P8$_2$ P8$_2$/CL2
Port 9		P9$_0$-P9$_1$	P9$_0$-P9$_1$	P9$_0$-P9$_1$	P9$_0$-P9$_1$
Port 10		P10$_0$-P10$_3$/D$_0$-D$_3$	P10$_0$-P10$_3$/D$_0$-D$_3$	P10$_0$-P10$_3$	P10$_0$-P10$_3$
Port 11		P11$_0$-P11$_3$/D$_4$-D$_7$	P11$_0$-P11$_3$/D$_4$-D$_7$	P11$_0$-P11$_3$	P11$_0$-P11$_3$
Timer/Event counter		8-bit	8-bit	8-bit	8-bit
Output and I/O pins		No on-chip resistor	No on-chip resistor	Mask options available	Mask options available
RESET		No pull-down resistor	No pull-down resistor	Mask options available	Mask options available
Comparator		4-channel	4-channel	4-channel	4-channel
Process		CMOS	CMOS	CMOS	CMOS
Package		24-pin plastic SOP (OTP) 24-pin shrink DIP (OTP)	20-pin plastic SOP (OTP) 20-pin shrink DIP (OTP)	24-pin plastic SOP 24-pin shrink DIP	24-pin plastic SOP 24-pin shrink DIP

μPD75P56 Application

Figure 2 gives an application example of a refrigerator or air conditioner circuitry.

Figure 2 Refrigerator or Air Conditioner Circuitry



Note: In case of air conditioner, heater is changed to fan motor.

OTP PROM (Program Memory Write and Verify)

The μPD75P56/P66 is a one-time programmable (OTP) PROM version of the μPD7556/66. The OTP is programmed by the pins and their functions listed in table 2. During OTP programming, addresses are updated by inputting clocks, instead of addresses, from the CL1 pin.

Table 2. OTP Access

Pin	Function
V_{PP}	OTP programming voltage pin (normally V_{DD})
CL1	Address update clock input during programming
MD0-MD3	Mode selection during OTP programming
D_0-D_7	8-bit data I/O pins during OTP programming
V_{DD}	Supply voltage pin: 4.5 to 6.0 V during normal operation; 6 V during OTP programming

Notes:

The μPD75P56/P66 has no erasure window. The program memory data cannot be erased with ultraviolet light.

OTP Operation Mode

The μPD75P56/P66 operates in the program memory write/verify mode when +6 V is applied to V_{DD} and +21 V to V_{PP}. Mode pins MD0-MD3 select the operation modes shown in table 3.

Table 3. OTP Operation Mode Selection (Note 1)

V_{PP} = +21 V; V_{DD} = +6 V

MD0	MD1	MD2	MD3	Operating Mode
H	L	H	L	Program memory address clear (Note 2)
L	H	H	H	Program memory write (Note 3)
L	L	H	H	Program memory verify (Note 4)
H	X	H	H	Program inhibit (Note 5)

Notes:

(1) X = L or H.

(2) While HLHL is applied, the program counter is cleared.

(3) While LHHH is applied, data applied to D_0-D_7 is written to the OTP.

(4) While LLHH is applied, the OTP contents at the address which the program counter indicates output to D_0-D_7.

(5) While HXHH is applied, the OTP is nonaccessible, and D_0-D_7 are set to high impedance.

Program Memory Write Procedure. The program memory write procedure follows (Data can be written at high speeds.):

(1) Connect unused pins to V_{SS} through a pull-down resistor. RESET is pulled up to V_{DD} through a resistor. Set CL1 low.

(2) Supply 5 V to V_{DD} and V_{PP}.

(3) Select the program memory address clear mode (HLHL).

(4) Change the voltage on V_{DD} to 6 V, and on V_{PP} to 21 V.

(5) Select the program inhibit mode (HXHH).

(6) Write data in the 1 ms write mode (LHHH).

(7) Select the program inhibit mode (HXHH).

(8) Select the verify mode (LLHH). After data is written, proceed to step 9; if data is not written, repeat steps 6-8.

(9) Perform one additional write.

(10) Select the program inhibit mode (HXHH).

(11) Increment the program memory address by one, by inputting four pulses to CL1.

(12) Repeat steps 6-11 until the end address occurs.

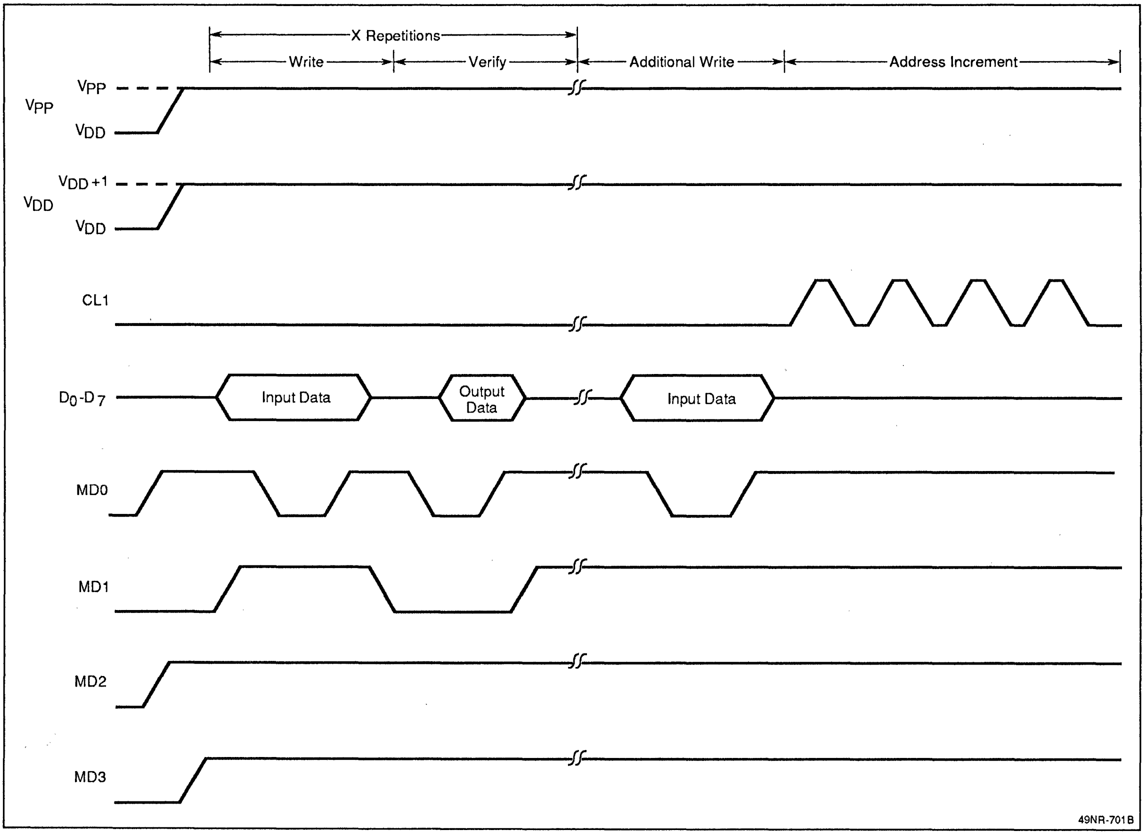
(13) Select the program memory address clear mode (HLHL).

(14) Change the voltage on V_{DD} and V_{PP} to 5 V.

(15) Turn off power.

The timing for steps 2-11 is shown in figure 3.

Figure 3. Timing Diagram for OTP Program Memory Write



3

Program Memory Read Procedure. The program memory read procedure follows:

(1) Connect unused pins to V_{SS} through a pull-down resistor. RESET is pulled up to V_{DD} through a resistor. Set CL1 low.

(2) Supply 5 V to V_{DD} and V_{PP}.

(3) Select the program memory address clear mode (HLHL).

(4) Change the voltage on V_{DD} to 6 V, and on V_{PP} to 21 V.

(5) Select the program inhibit mode (HXHH).

(6) Select the verify mode (LLHH). Data is read from 000H. Each time four clock pulses are input to the X1 pin, data from one address is output.

(7) Pulse input to the CL1. Program memory address is updated at the rising edge of the third pulse. Address after updated one (+1) is updated every four pulses. Repeat update to last address.

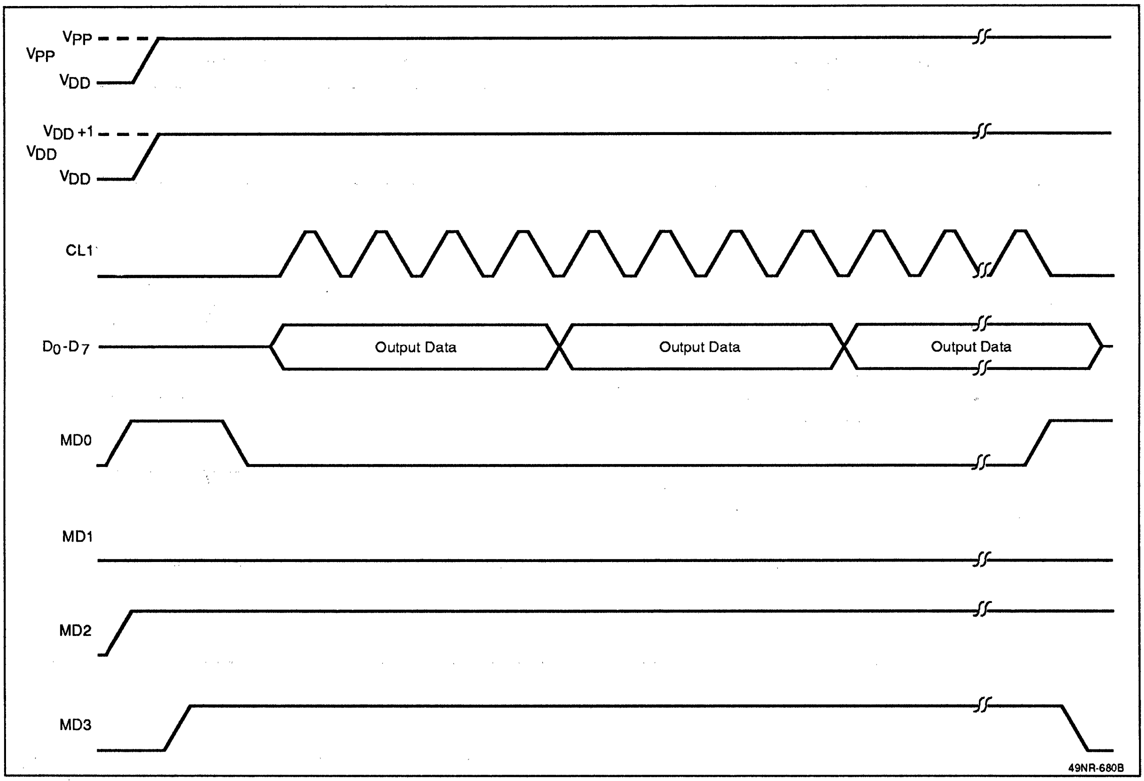
(8) Select the program memory address clear mode (HLHL).

(9) Change the voltage on V_{DD} and V_{PP} to 5 V.

(10) Turn off power.

The timing for steps 2-9 is shown in figure 4.

Figure 4. Timing Diagram for Program Memory Read



ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	-65 to +150°C
Power supply voltage, V_{DD}	-0.3 to +7.0 V
Input voltage, V_I Except ports 10, 11	-0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 1) (Note 2)	-0.3 to V_{DD} +0.3 V -0.3 to +13 V
Output voltage, V_O Except ports 10, 11	-0.3 to V_{DD} +0.3 V
Ports 10, 11 (Note 1) (Note 2)	-0.3 to V_{DD} +0.3 V -0.3 to +13 V
Output current, high I_{OH} One port	-5 mA
All output ports, total	−15 mA
Output current, low I_{OL} Ports 8, 9	30 mA
Other ports	15 mA
All ports, total	100 mA
Power dissipation, P_D (T_A = +70°C) Shrink DIP	480 mW
SOP	250 mW

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Notes:

(1) CMOS I/O or N-channel open drain + internal pull-up resistor.

(2) N-channel open drain I/O.

Capacitance

T_A = 25°C, V_{DD} = V_{SS} = 0 V; f = 1 MHz
Unmeasured pins returned to V_{SS}

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I			50	pF	$P0_0$-$P0_1$; $P1_0$-$P1_3$
				15		CIN_0-CIN_3
Output capacitance	C_O			35	pF	Ports 8, 9
I/O capacitance	$C_{I/O}$			35	pF	Ports 10, 11

DC Characteristics, Normal Operation; V_{DD} = 4.5 to 6.0 V; V_{SS} = 0 V

T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.7 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	
Input high voltage ports 10, 11 (Note 1)	V_{IH3}	0.7 V_{DD}		12	V	
Input low- and high-level voltage RESET	V_{IHDR}	0.9 V_{DDDR}		V_{DDDR} + 0.2	V	Data retention mode
Input low voltage except CL1	V_{IL1}	0		0.3 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.5	V	
Input leakage current except CL1	I_{LI1}	−3		3	µA	$0\ V < V_I < V_{DD}$
Input leakage current CL1	I_{LI2}	−10		10	µA	$0\ V < V_I < V_{DD}$
Input leakage current ports 10, 11 (Note 1)	I_{LI3}			10	µA	$V_I = 12\ V$
Output voltage high $P0_1$, $P0_2$, ports 8-11	V_{OH}	V_{DD} − 2.0			V	$I_{OH} = -1$ mA
Output voltage low ports 10, 11	V_{OL}			0.4	V	$I_{OL} = 1.6$ mA
				2.0	V	$I_{OL} = 10$ mA
Output voltage low ports 8, 9	V_{OL}			2.0	V	$I_{OL} = 15$ mA
Output leakage current	I_{LO1}	−3		3	µA	$0\ V \leq V_O < V_{DD}$
Output leakage current, port 8-11 (Note 1)	I_{LO2}			10	µA	$V_O = 12\ V$
Supply voltage, data retention mode	V_{DDDR}	2.0		6.0	V	
Supply current, normal operation	I_{DD1}		400	1400	µA	µPD75P56: V_{DD} = 5 V ±10%; R = 56Ω ±2%
			700	2300	µA	µPD75P66: V_{DD} = 5 V ±10%; f_{CC} = 700 kHz
Supply current, HALT mode	I_{DD2}		120	400	µA	µPD75P56: V_{DD} = 5 V ±0.5 V; R = 56Ω ±2%
			450	1500	µA	µPD75P66: V_{DD} = 5 V ±10%; f_{CC} = 700 kHz
Supply current, STOP mode	I_{DD3}		0.1	10	µA	V_{DD} = 5 V ±10%
			0.1	10	µA	V_{DD} = 5 V ±10%
Supply current, data retention mode	I_{DDDR}		0.1	5	µA	V_{DDDR} = 2.0 V

Notes:

(1) N-channel, open-drain I/O ports.

DC Characteristics, Programming mode; V_{DD} = 6.0 V ±0.25 V; V_{PP} = 21 ±0.5 V; V_{SS} = 0 V

T_A = 25°C; (Notes 1 and 2)

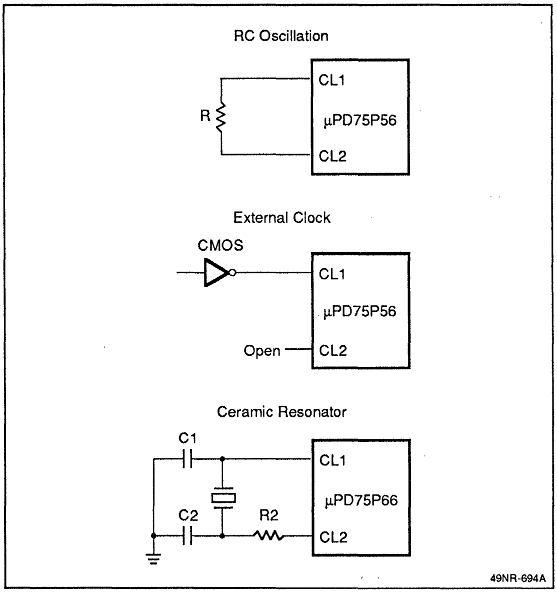
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input high voltage except CL1	V_{IH1}	0.7 V_{DD}		V_{DD}	V	
Input high voltage CL1	V_{IH2}	V_{DD} − 0.5		V_{DD}	V	
Input low voltage except CL1	V_{IL1}	0		0.3 V_{DD}	V	
Input low voltage CL1	V_{IL2}	0		0.5	V	
Input leakage current	I_{LI}			10	μA	V_{IN} = V_{IL} or V_{IH}
Output voltage high	V_{OH}	V_{DD} − 2.0			V	I_{OH} = −1 mA
Output voltage low	V_{OL}			0.4	V	I_{OL} = 1.6 mA
V_{DD} supply current	I_{DD}			30	mA	
V_{PP} power current	I_{PP}			30	mA	MD0 = V_{IL}, MD1 = V_{IH}

Notes:

(1) V_{PP}, including an overshoot, should not exceed +22 V.

(2) Apply V_{DD} before V_{PP}, and cut off after V_{PP}.

Figure 5. Recommended Circuits



Comparator

T_A = −10 to +70°C; V_{DD} = 4.5 to 6.0 V, V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage range	V_{CIN}/V_{REF}	0		V_{DD}	V	
Response time	T_{COMP}	2		4	MC(Note 1)	
Input voltage resolution	ΔV_{CIN}		10	50	mV	
Input leakage current	I_{CIN}/I_{REF}	−3		3	μA	
V_{REF} bias resistance	R_{REF}		100		kΩ	
Comparator circuit current (Note 2)	I_{DDCMP}		50		μA	f_{CC} = 500 kHz

Notes:

(1) Machine cycle.

(2) Excluding current through bias resistor.

AC Characteristics, Normal Operation; V_{DD} = 4.5 to 6.0 V; V_{SS} = 0 V

T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
System clock oscillation frequency CL1, CL2	f_{CC}	400	500	600	kHz	μPD75P56: R = 56 kΩ ±2%
		290		710	kHz	μPD75P66: ceramic resonator
System clock input frequency, CL1	f_C	10		710	kHz	μPD75P56: 50% duty
Oscillation stabilization time	t_{OS}	20			ms	OS stabilization after minimum of operating voltage reached. (Note 1)
System clock rise time, CL1	t_{CR}			0.2	μs	
System clock fall time, CL1	t_{CF}			0.2	μs	
System clock pulse width, CL1	t_{CH}	0.7		50	μs	
System clock pulse width, CL1	t_{CL}	0.7		50	μs	
Event input frequency ($P0_0$)	f_{P0}	0		710	kHz	50% duty
$P0_0$ rise time	t_{POR}			0.2	μs	
$P0_0$ fall time	t_{POF}			0.2	μs	
$P0_0$ pulse width, high	t_{P0H}	0.7			μs	
$P0_0$ pulse width, low	t_{P0L}	0.7			μs	
INT0 high time	t_{IOH}	10			μs	
INT0 low time	t_{IOL}	10			μs	
RESET high time	t_{RSH}	10			μs	
RESET low time	t_{RSL}	10			μs	
RESET setup time	t_{SRS}	0			μs	
RESET hold time	t_{HRS}	0			μs	

Notes:

(1) Hold the RESET signal at a high level until oscillation becomes stable.

AC Characteristics, Programming Mode; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 21 ±0.5 V, V_{SS} = 0 V

T_A = 25°C

Parameter	Symbol	Note 1	Min	Typ	Max	Unit	Conditions
Address setup time for MD0 ↓ (Note 2)	t_{AS}	t_{AS}	2			μs	
MD1 setup time for MD0 ↓	t_{MIS}	t_{OES}	2			μs	
Data setup for MD0 ↓	t_{DS}	t_{DS}	2			μs	
Address hold time for MD0 ↑ (Note 2)	t_{AH}	t_{AH}	2			μs	
Data hold time for MD0↑	t_{DH}	t_{DH}	2			μs	
MD0 ↑ to data output float delay time	t_{DF}	t_{DF}	0		200	ns	
V_{PP} setup time for MD3 ↑	t_{VPS}	t_{VPS}	2			μs	
V_{DD} setup time for MD3 ↑	t_{VDS}	t_{VCS}	2			μs	
Initial program pulse width	t_{PW}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{OPW}	t_{OPW}	0.95		21.0	ms	
MD0 setup time for MD1 ↑	t_{MOS}	t_{CES}	2			μs	
MD0 ↓ to data output delay time	t_{DV}	t_{DV}			1 (Note 3)	μs	MD0 = MD1 = V_{IL}
MD1 hold time for MD0 ↑	t_{M1H}	t_{OEH}	2			μs	t_{M1H} + t_{MIR} ≥ 50 μs
MD1 recovery time for MD0 ↓	t_{M1R}	t_{OR}	2			μs	
Program counter reset time	t_{PCR}		10			μs	
CL1 input low- and high-level widths	t_{XH}, t_{XL}		0.7			μs	
CL1 input frequency	f_X				710	kHz	
Initial mode set time	t_I		2			μs	
MD3 setup time for MD1 ↑	t_{M3S}		2			μs	
MD3 hold time for MD1 ↓	t_{M3H}		2			μs	
MD3 setup time for MD0 ↓	t_{M3SR}		2			μs	During program memory read
Address to data output delay time (Note 2)	t_{DAD}	t_{ACC}	2			μs	
Address to data output hold time (Note 2)	t_{HAD}	t_{OH}	0		300	ns	
MD3 hold time for MD0 ↑	t_{M3HR}		2			μs	
MD3 ↓ to data output float delay time	t_{DFR}		2			μs	

Notes:

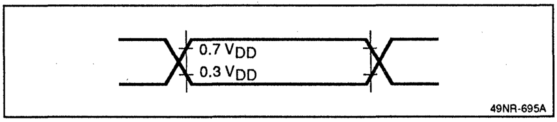
(1) Symbol of the corresponding μPD27C256.

(2) "1" is added to the internal address signal at the rising edge of the third CL1 input. The signal is not input to the pin.

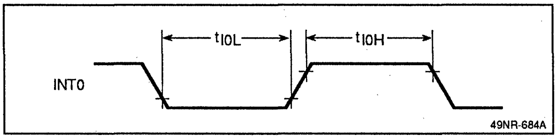
(3) During CMOS output.

Timing Waveforms

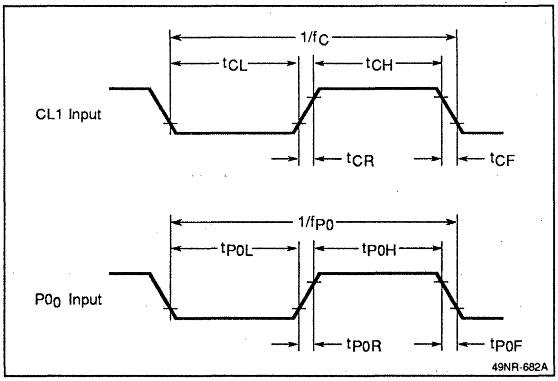
AC Timing Measurement Points



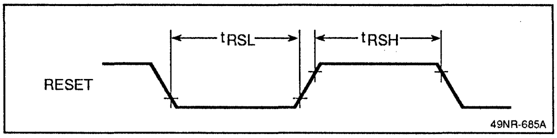
Test Input Timing



Clock Timing

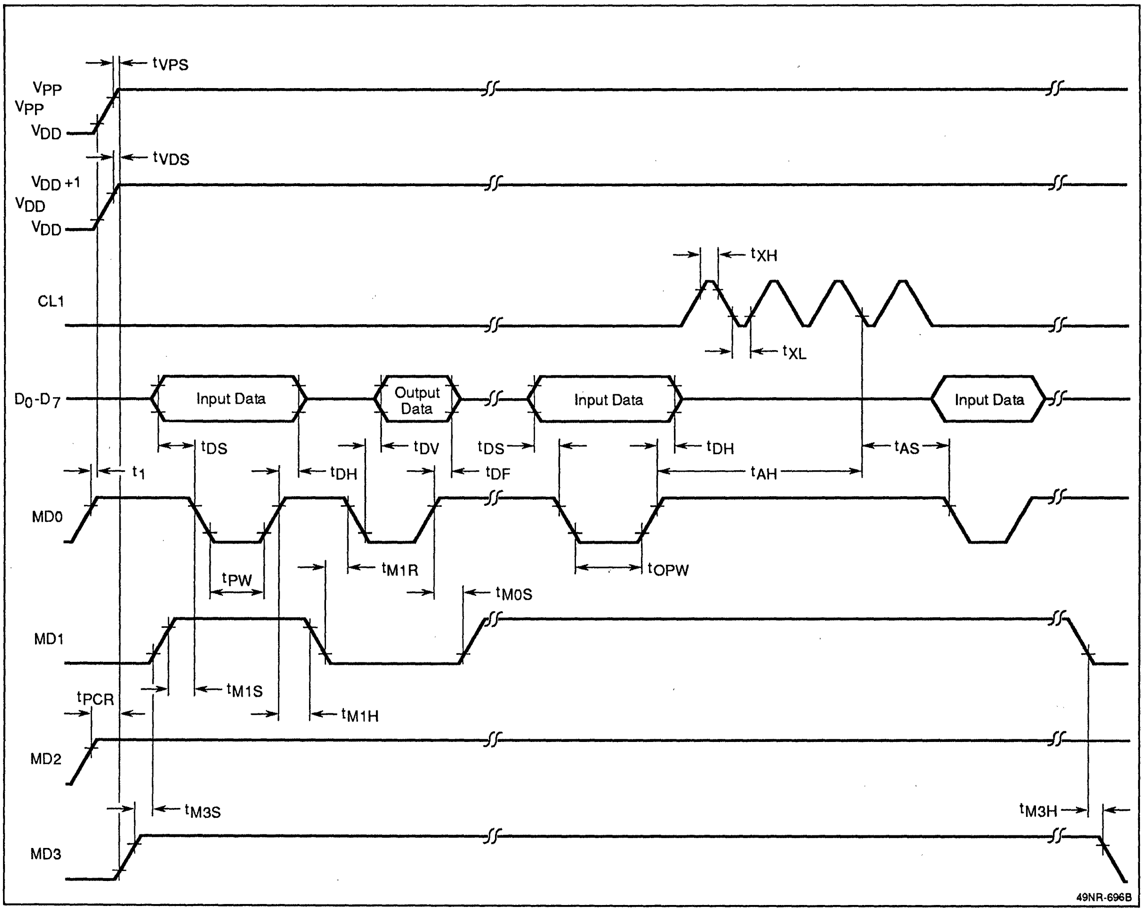


RESET Timing



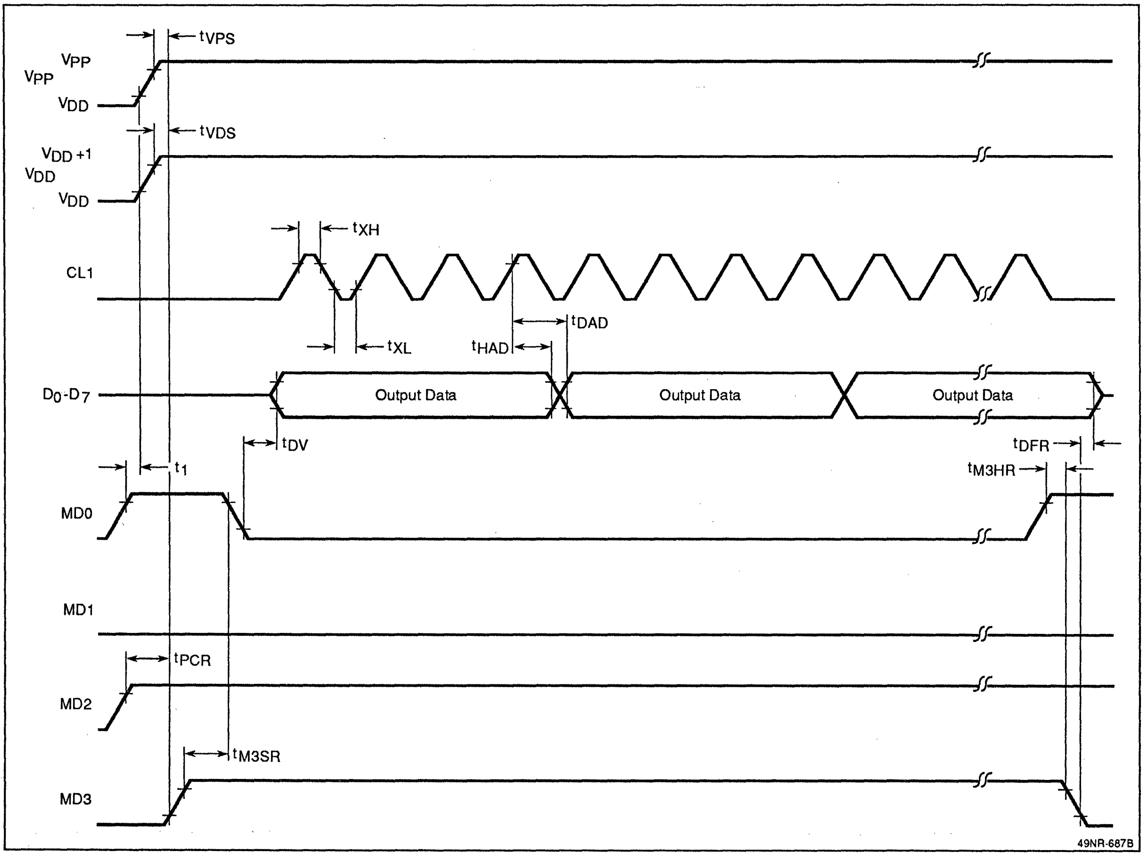
Timing Waveforms (cont)

Program Memory Write (OTP)



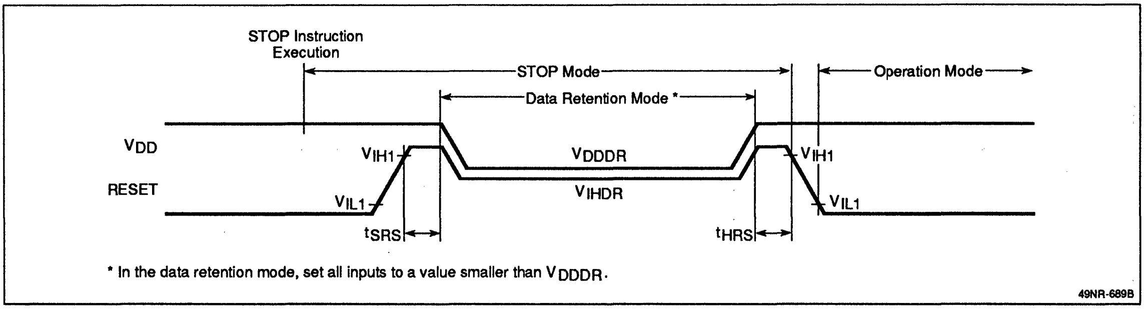
Timing Waveforms (cont)

Program Memory Read (OTP)



Timing Waveforms (cont)

Data Retention Timing μPD75P56

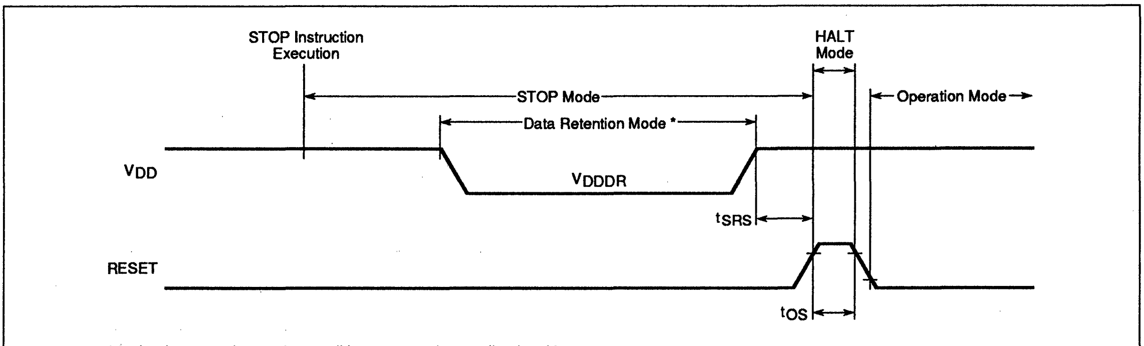


* In the data retention mode, set all inputs to a value smaller than V_{DDDR}.

49NR-689B

3

Data Retention Timing μPD75P66

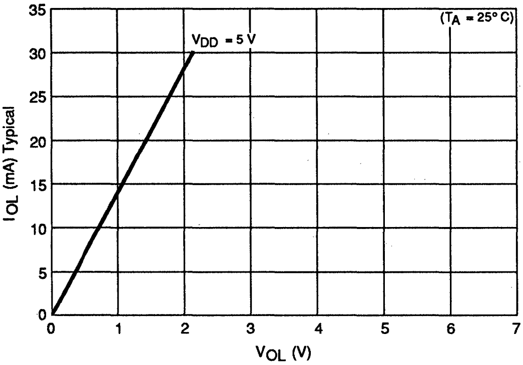


* In the data retention mode, set all inputs to a value smaller than V_{DDDR}.

49NR-688B

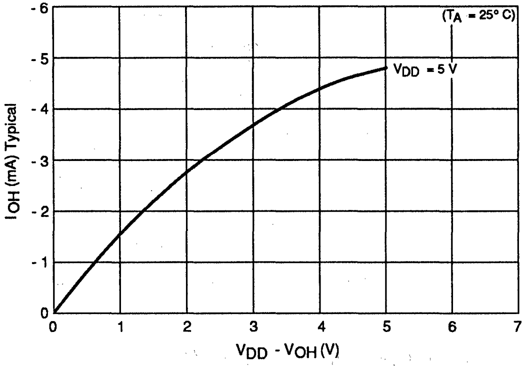
Operating Characteristics

I_{OL} vs V_{OL} (Ports 8, 9)



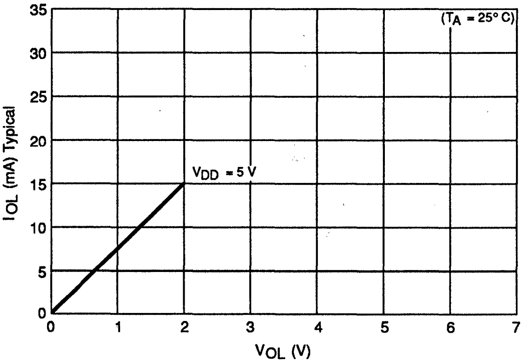
* The absolute maximum rating is 30 mA per pin.

I_{OH} vs V_{OH}



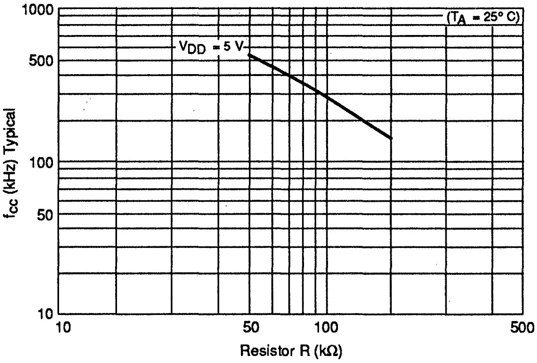
* The absolute maximum rating is -5 mA per pin.

I_{OL} vs V_{OL} (Ports 10, 11)



* The absolute maximum rating is 15 mA per pin.

f_{CC} vs R



* This curve shows a device characteristic example and does not guarantee the ratings.

49NR-697B

Section 4
μPD75000 Series:
4-Bit, High-Integration Microcomputers

NEC
NEC Electronics Inc.

μPD7500x/75P008 General-Purpose 4-Bit Microcomputers With Multiple I/Os

Description

The μPD7500x/75P008 is a family of single-chip CMOS microcomputers containing CPU, ROM, RAM, I/O ports, several timer/counters, vectored interrupts, subsystem clock, and serial interface.

The instruction set allows the user to manipulate RAM data and I/O ports in 1-, 4-, and 8-bit units. The devices are ideally suited for controlling VCRs, telephones, and meters.

Both EPROM and OTP versions are available. See ordering information.

Features

□ 103 instructions
—Bit manipulation
—4-bit and 8-bit transfer
—1-byte relative branch
—GETI instruction, to convert one 2-byte or two 1-byte instructions into a single 1-byte instruction

□ Fast execution time (@ 4.19 MHz)
—High-speed cycle: 0.95 μs
—Lower-voltage cycles: 1.91 and 15.3 μs

□ Program ROM
—μPD75004: 4096 bytes
—μPD75006: 6016 bytes
—μPD75008/P008: 8064 bytes

□ 512 x 4 bits of RAM
—Allows operation on 1, 4, or 8 bits

□ Bit sequential buffer
—16-bit, bit manipulation memory

□ Eight 4-bit registers

□ Accumulators
—1-bit (CY)
—4-bit (A)
—8-bit (XA)

□ 26 I/O lines
—8 N-channel open drain; can withstand 10 V
—18 outputs directly drive LEDs (I_{sink} = 15 mA rms)

□ One external event input

□ Three timers
—8-bit basic interval timer
—8-bit timer/event counter
—14-bit watch timer

□ 8-bit serial interface
—SBI mode
—2- or 3-wire mode: data transfer can be full duplex or receive only, and can be MSB or LSB first

□ Vectored interrupts
—Three external interrupts
—Three internal interrupts
—Nine inputs which each generate one interrupt request

□ Eight input-only lines

□ Standby modes
—HALT mode: stops CPU only
—STOP mode: stops main system clock

□ Mask option pull-up resistors for ports 4 and 5

□ Operates with oscillator or ceramic resonator

□ CMOS operation with V_{DD} from 2.7 to 6.0 V

□ Power consumption @ 5 V and 4.19 MHz
—Normal mode: 2.5 mA typical
—HALT mode: 0.5 mA typical
—STOP mode: 0.5 μA typical

□ Programmable version:
—μPD75P008 OTP

Ordering Information

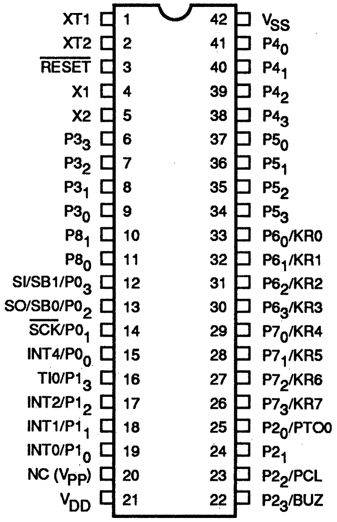
Part Number	Package Type	ROM
μPD75004CU-xxx	42-pin plastic SDIP	Mask ROM
μPD75004GB-xxx-3B4	44-pin plastic QFP	Mask ROM
μPD75006CU-xxx	42-pin plastic SDIP	Mask ROM
μPD75006GB-xxx-3B4	44-pin plastic QFP	Mask ROM
μPD75008CU-xxx	42-pin plastic SDIP	Mask ROM
μPD75008GB-xxx-3B4	44-pin plastic QFP	Mask ROM
μPD75P008CU	42-pin plastic SDIP	OTP
μPD75P008GB-3B4	44-pin plastic QFP	OTP

Notes:

(1) xxx indicates ROM code suffix

Pin Configurations

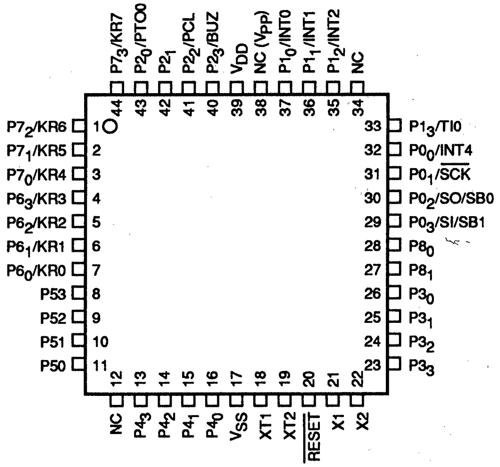
42-Pin Plastic SDIP



Note : V_{PP} is the programming pin in the 75P008 and should be connected to V_{DD}. It is not connected in the mask ROM ports.

83RD-6363A

44-Pin Plastic QFP



Note : V_{PP} is the programming pin in the 75P008 and should be connected to V_{DD}. It is not connected in the mask ROM ports.

83RD-6364A

Pin Identification

Symbol	Function
NC (V_{PP})	No connection (programming voltage for μPD75P008
$P0_0$/INT4	Port 0 input; interrupt 4
$P0_1$/$\overline{SCK}$	Port 0 input; serial clock
$P0_2$/SO/SB0	Port 0 input; serial out; serial interface
$P0_3$/SI/SB1	Port 0 input; serial in; serial interface
$P1_0$/INT0	Port 1 input; interrupt 0
$P1_1$/INT1	Port 1 input; interrupt 1
$P1_2$/INT2	Port 1 input; interrupt 2
$P1_3$/TI0	Port 1 input; timer 0 input
$P2_0$/PTO0	Port 2 I/O; timer/event counter output
$P2_1$	Port 2 I/O
$P2_2$/PCL	Port 2 I/O; clock output
$P2_3$/BUZ	Port 2 I/O; buzzer output
$P3_0$-$P3_3$	Port 3 I/O
$P4_0$-$P4_3$	Port 4 I/O
$P5_0$-$P5_3$	Port 5 I/O
$P6_0$/KR0	Port 6 I/O; key scan input 0
$P6_1$/KR1	Port 6 I/O; key scan input 1
$P6_2$/KR2	Port 6 I/O; key scan input 2
$P6_3$/KR3	Port 6 I/O; key scan input 3
$P7_0$/KR4	Port 7 I/O; key scan input 4
$P7_1$/KR5	Port 7 I/O; key scan input 5
$P7_2$/KR6	Port 7 I/O; key scan input 6
$P7_3$/KR7	Port 7 I/O; key scan input 7
$P8_0$-$P8_1$	Port 8 I/O
$\overline{RESET}$	Reset input
V_{DD}	Positive power supply
V_{SS}	Ground
X1, X2	Main clock inputs
XT1, XT2	Subsystem clock inputs

PIN FUNCTIONS

$P0_0$-$P0_3$, INT4, $\overline{SCK}$, SO/SB0, SI/SB1 (Port 0, Interrupt 4, Serial Interface)

These pins can be used as 4-bit input port 0. $P0_0$ can also be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. $P0_1$-$P0_3$ may also be used for the serial interface in the SBI, 2-wire or 3-wire mode. SI is the serial input, SO is the serial output, and $\overline{SCK}$ is the serial clock. Reset causes these pins to default to the port 0 input mode.

$P1_0$-$P1_3$, INT0-INT2, TI0 (Port 1, Edge-Triggered Interrupts, Timer Input)

These pins can be used as 4-bit input port 1. $P1_0$ and $P1_1$ can also be used for edge-triggered interrupts INT0 and INT1. $P1_2$ can be used for INT2, which is also an edge-triggered input, but one which generates an interrupt request and does not cause an interrupt. $P1_3$ can be used as an input clock to the timer/event counter to count external events. Reset causes these pins to default to the port 1 input mode.

$P2_0$-$P2_3$, PTO_0, PCL, BUZ (Port 2, Timer/Event Counter, Clock, or Buzzer Output)

These pins can be used as 4-bit I/O port 2. When used as an output the data is latched. When used as an input port the port outputs are three-state. $P2_0$ can also be used as the output of the timer/event counter flip flop (TOUT); $P2_2$ can be used as the output for the clock generator (PCL); and $P2_3$ can be used to output square waves for a buzzer. Reset causes these pins to default to the port 2 input mode.

$P3_0$-$P3_3$ (Port 3)

These pins are used for I/O port 3. Each bit in this port can be independently programmed to be either an input or an output. This port has latched outputs, and can directly drive LEDs. A reset signal causes this port to default to the input mode.

$P4_0$-$P4_3$, $P5_0$-$P5_3$ (Ports 4 and 5)

Port 4 and 5 are 4-bit I/O ports which can be combined together to function as a single 8-bit port. They have latched outputs. Port 4 will directly drive LEDs. Outputs are N-channel open drain, and can withstand up to 10 volts; pull-up resistor mask options are available for these ports. A reset signal causes these ports to default to the input mode.

$P6_0$-$P6_3$, $P7_0$-$P7_3$, KR0-KR7 (Ports 6, 7, and Edge Detection)

Ports 6 and 7 are 4-bit I/O ports with latched outputs. Each pin of port 6 can be independently programmed to be either an input or an output, while port 7 can be programmed to be either all inputs or all outputs. Ports 6 and 7 can be paired together to function as one 8-bit port. Alternately, these pins may be used to detect the falling edge of inputs KR0-KR3 (port 6) and KR4-KR7 (port 7). A reset signal causes these ports to default to the input mode.

$P8_0$-$P8_1$ (Port 8)

Port 8 is a 2-bit I/O port. Outputs are latched. A reset signal causes this port to default to the input mode.

NC/V_{PP} (No Connection/Programming Pin)

This pin may be left unconnected when using the μPD7500x. When using the programmable devices, this pin is used to input the programming voltage during the EPROM write/verify cycles. During normal operation of the programmable device, this pin should be tied to V_{DD}.

X1, X2 (Main System Clock Inputs)

These pins are the main system clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

XT1, XT2 (Subsystem Clock Inputs)

These pins are the subsystem clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

$\overline{\text{RESET}}$ (Reset)

This is the reset input, and it is active low.

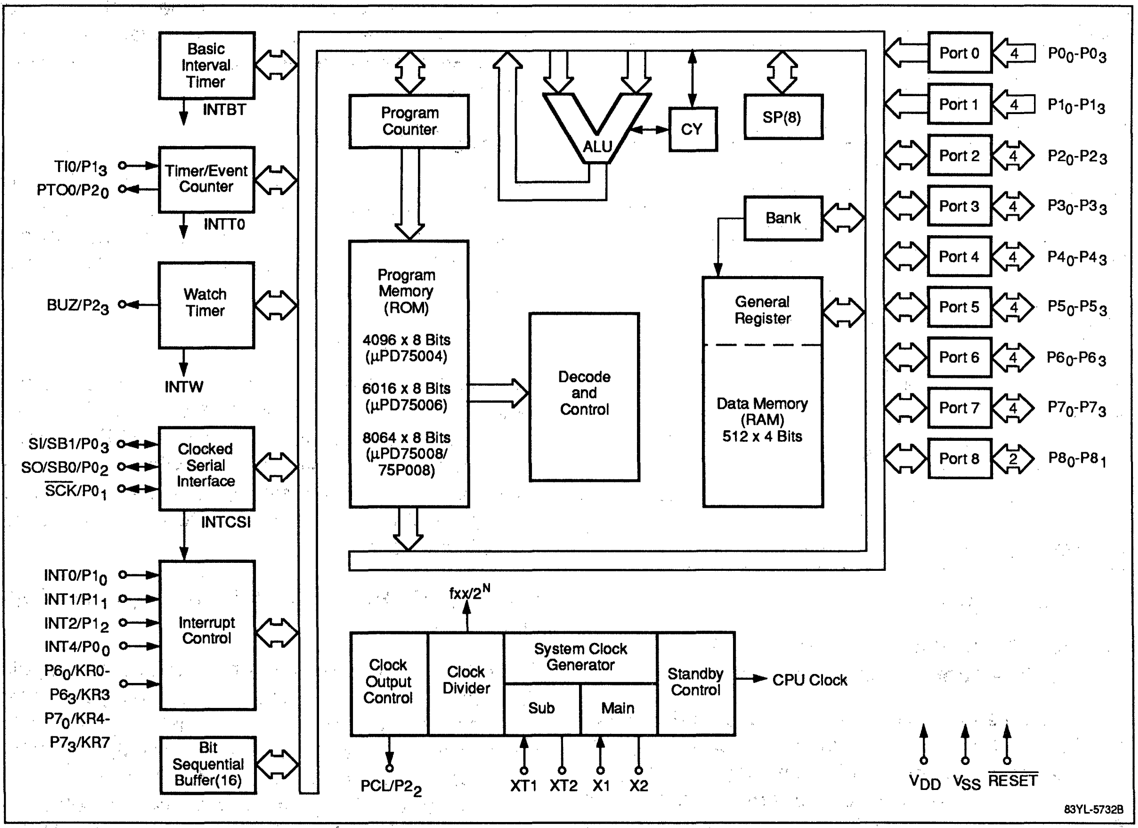
V_{DD} (Power Supply)

The system positive power supply pin.

V_{SS} (Ground)

System ground.

Block Diagram



Product Comparison

Item	μPD75004	μPD75006	μPD75008	μPD75P008CU/GB
Program memory	Mask ROM 000H–FFFH 4096 x 8 bits	Mask ROM 0000H–177FH 6016 x 8 bits	Mask ROM 0000H–1F7FH 8064 x 8 bits	OTP 0000H–1F7FH 8064 x 8 bits
Data memory	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4
3-byte branch instruction	None	Provided	Provided	Provided
Other instructions	Provided	Provided	Provided	Provided
Program counter	12 bits	13 bits	13 bits	13 bits
Pull-up resistor, ports 0–3; 6–8	Can be specified by software			
Pull-up resistor, ports 4, 5	Mask option	Mask option	Mask option	Not provided
Operating voltage range	2.7 to 6.0 V	2.7 to 6.0 V	2.7 to 6.0 V	5 V ± 5%
Package	42-pin plastic shrink DIP 44-pin plastic QFP (bent)			42-pin plastic shrink DIP 44-pin plastic QFP (bent)

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

$T_A = 25°C$

Supply voltage, V_{DD}	−0.3 to +7.0 V
Programming voltage, V_{PP} (μPD75P008 only)	-0.3 to +13.5 V
Input voltage, V_{IN1}	−0.3 to V_{DD} + 0.3 V
Input voltage, V_{IN2} (Ports 4 and 5 with open drain)	−0.3 to 11 V
Output voltage, V_O	−0.3 to V_{DD} + 0.3 V
High-level output current, I_{OH} Single pins	−10 mA
All pins	−30 mA
Low-level output current, I_{OL} (Note 1) Ports 0, 3-5 (one port pin)	30 mA peak, 15 mA rms
All ports except 0, 3-5	20 mA peak, 10 mA rms
Total of ports 0, 3-5, 8	160 mA peak, 120 mA rms
Total of ports 2, 6, 7	66 mA peak, 33 mA rms
Storage temperature, T_{STG}	−65 to +150°C
Operating temperature, T_{OPT} (μPD7500x)	−40 to +85°C
Operating temperature, T_{OPT} (μPD75P008 only)	−10 to +70°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Notes:

(1) Effective value = Peak value x $(\text{Duty})^{1/2}$

Capacitance

$T_A = 25°C$; $V_{DD} = 0$ V

Parameter	Symbol	Min	Max	Unit	Conditions
Input capacitance	C_{IN}		15	pF	f = 1 MHz; all unmeasured pins returned to ground
Output capacitance	C_{OUT}		15	pF	
I/O capacitance	C_{IO}		15	pF	

Main System Clock Oscillator Characteristics

μPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
μPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 1A)	Oscillation frequency (Note 1)	f_{XX}	1.0		5.0	MHz	After V_{DD} reaches the minimum oscillator operating voltage range.
	Oscillation stabilization time (Note 2)				4 (Note 3)	ms	
Crystal resonator (Figure 1A)	Oscillation frequency (Note 1)	f_{XX}	1.0	4.19	5.0	MHz	
	Oscillation stabilization time (Note 2)				10 (Notes 3, 4)	ms	
					30 (Notes 3, 5)	ms	
External clock (Figure 1B)	X1 input frequency (Note 1)	f_{XX}	1.0		5.0	MHz	
	X1 input low- and high-level width	t_{XH}, t_{XL}	100		500	ns	

Notes:

(1) The oscillation frequency and X1 input frequency are included only to show the characteristics of the oscillators. Refer to the AC Characteristics table for actual instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillator to stabilize after voltage is applied or the STOP mode is released.

(3) Values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

(4) V_{DD} = 4.5 to 6.0 V for 7500x or 4.5 to 5.5 V for μPD75P008.

(5) For μPD7500x only at V_{DD} = 2.7 – 6.0 V

Subsystem Clock Oscillator Characteristics

μPD7500x: T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V
μPD75P008: T_A = −10 to +70°C; V_{DD} = 4.5 to 5.5 V

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 2A)	Oscillation frequency	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time			1.0	2	s	See note 4 under Main System Oscillator Characteristics
					2	s	See note 5 under Main System Oscillator Characteristics
External clock (Figure 2B)	XT1 input frequency	f_{XT}	32		100	kHz	
	XT1 input low- and high-level width	t_{XTH}, t_{XTL}	5		15	μs	

Recommended Oscillator Circuit Constants (For 7500x only)

Main system clock = Ceramic; T_A = −40 to +85°C

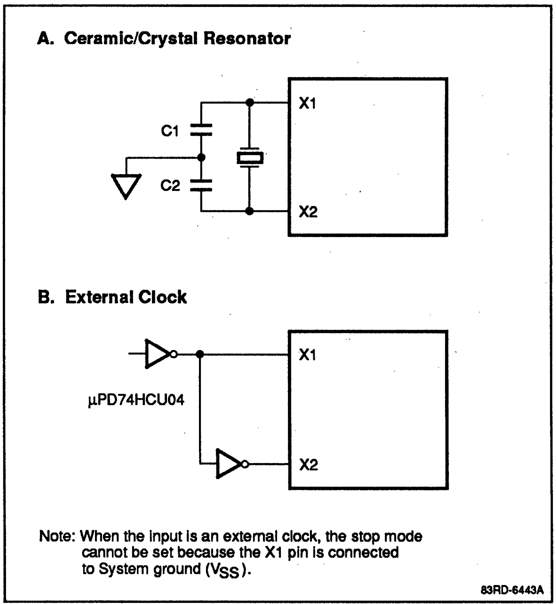
	Part Number	Frequency	C1	C2	Oscillation Voltage	
Manufacturer	(Note 1)	(MHz)	(pF)	(pF)	Min (V)	Max (V)
Murata	CSA x.xxMK	1.0–1.99	30	30	2.7	6.0
	CSA x.xxMG093	2.0–2.44	30	30	2.7	6.0
	CST x.xxMG093	2.0–2.44	(Note 2)	(Note 2)	2.7	6.0
	CSA x.xxMGU	2.45–5.0	30	30	2.7	6.0
	CST x.xxMGU	2.45–5.0	(Note 2)	(Note 2)	2.7	6.0
	CSA x.xxMG	2.0–5.0	30	30	3.0	6.0
	CST x.xxMG	2.0–5.0	(Note 2)	(Note 2)	3.0	6.0
Kyocera	KBR 1000H	1.0	100	100	2.7	6.0
	KBR 2.0MS	2.0	47	47	2.7	6.0
	KBR 4.0MS	4.0	33	33	2.7	6.0
	KBR 5.0M	5.0	33	33	3.0	6.0

Notes:

(1) x.xx indicates frequency.

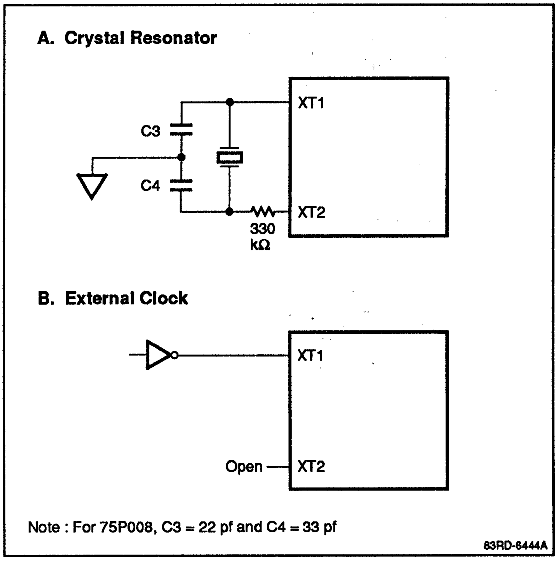
(2) C1 and C2 not required; they are in the oscillator.

Figure 1. Main System Clock Configurations



Note: When the input is an external clock, the stop mode cannot be set because the X1 pin is connected to System ground (V_{SS}).

Figure 2. Subsystem Clock Configurations



Note : For 75P008, C3 = 22 pf and C4 = 33 pf

Recommended Oscillator Circuit Constants (For 7500x only)

Main system clock = Crystal; T_A = −20 to +70°C

		Frequency	C1 (Note 1)	C2	Oscillation Voltage	
Manufacturer	Part Number	(MHz)	(pF)	(pF)	Min (V)	Max (V)
Kinseki	HC-6U	1.0–2.0	20	22	2.7	6.0
	HC-18U, HC-43/U, HC-49/U	2.0–5.0	20	22	2.7	6.0

Notes:

(1) Keep C1 between 15 and 33 pF when adjusting the oscillation frequency.

Recommended Oscillator Circuit Constants (For 7500x only)

Subsystem clock = Crystal; T_A = −10 to +60°C

		Frequency	C3 (Note 1)	C4	Oscillation Voltage	
Manufacturer	Part Number	(MHz)	(pF)	(pF)	Min (V)	Max (V)
Kinseki	P-3	32.768	18	18	2.7	6.0

Notes:

(1) Keep C3 between 10 and 33 pF when adjusting the oscillation frequency.

DC Characteristics

µPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
µPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	$0.7V_{DD}$		V_{DD}	V	Ports 2, 3, 8
	V_{IH2}	$0.8V_{DD}$		V_{DD}	V	Ports 0, 1, 6, 7, and $\overline{\text{RESET}}$
	V_{IH3}	$0.7V_{DD}$		V_{DD}	V	Ports 4 and 5; built-in pull-up resistor
		$0.7V_{DD}$		10	V	Ports 4 and 5 with open drain
	V_{IH4}	V_{DD}−0.5		V_{DD}	V	X1, X2, XT1
Low-level input voltage	V_{IL1}	0		$0.3V_{DD}$	V	Ports 2, 3, 4, 5, 8
	V_{IL2}	0		$0.2V_{DD}$	V	Ports 0, 1, 6, 7; $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2, XT1
High-level output voltage	V_{OH1} (Note 1)	V_{DD}−1.0			V	Ports 0, 2, 3, 6, 7, 8; I_{OH} = −1 mA
	V_{OH2} (Note 2)	V_{DD}−0.5			V	Ports 0, 2, 3, 6, 7, 8; V_{DD} = 2.7 to 6.0 V; I_{OH} = −100 µA
Low-level output voltage	V_{OL1}		0.4	2.0	V	Ports 4 and 5; (Note 1); I_{OL} = 15 mA;
			0.6	2.0	V	Port 3; (Note 1); I_{OL} = 15 mA
				0.4	V	Ports 0, 2-8; (Note 1); I_{OL} = 1.6 mA
				0.5 (Note 2)	V	Ports 0, 2-8; V_{DD} = 4.5 to 6.0 V; I_{OL} = 400 µA
	V_{OL2}			$0.2V_{DD}$ (Note 1)	V	SB0, 1 open drain; pull-up resistance ≥ 1kΩ
				$0.2V_{DD}$ (Note 2)	V	SB0, 1 open drain; V_{DD} = 2.7 to 6.0 V; pull-up resistance ≥ 5kΩ
High-level input leakage current	I_{LIH1}			3	µA	All except X1, X2, and XT1; V_{IN} = V_{DD}
	I_{LIH2}			20	µA	X1, X2, and XT1; V_{IN} = V_{DD}
	I_{LIH3}			20	µA	Ports 4, 5 with open drain; V_{IN} = 10 V

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Low-level input leakage current	I_{LIL1}			−3	µA	All except X1, X2, and XT1; V_{IN} = 0 V
	I_{LIL2}			−20	µA	X1, X2, and XT1; V_{IN} = 0 V
High-level output leakage current	I_{LOH1}			3	µA	All except ports 4 and 5 with open drain; V_{OUT} = V_{DD}
	I_{LOH2}			20	µA	Ports 4 and 5 with open drain; V_{OUT} = 10 V
Low-level output leakage current	I_{LOL}			−3	µA	V_{OUT} = 0 V
Built-in pull-up resistor	R_{L1}	15	40	80	kΩ	Ports 0-3, 6-8 (except $P0_0$); V_{IN} = 0 V; V_{DD} = 5.0 V ± 10%
		30 (Note 2)		300 (Note 2)	kΩ	Ports 0-3, 6-8 (except $P0_0$); V_{IN} = 0 V; V_{DD} = 3.0 V ± 10%
	R_{L2} (Note 2)	15	40	70	kΩ	Ports 4, 5; V_{OUT} = V_{DD} −2.0 V; V_{DD} = 5.0 V ± 10%
		10		60	kΩ	Ports 4, 5; V_{OUT} = V_{DD} −2.0 V; V_{DD} = 3.0 V ± 10%
Supply current (Note 3)	I_{DD1}	(Note 2)	2.5	8.0	mA	V_{DD} = 5.0 V ± 10% (Notes 4, 6)
		(Note 2)	0.35	1.2	mA	V_{DD} = 3.0 V ± 10% (Notes 4, 7)
		(Note 8)	5	15	mA	V_{DD} = 5 V ±10%; (Notes 4, 6)
	I_{DD2}		500	1500	µA	HALT mode; V_{DD} = 5 V ± 10% (Note 4)
		(Note 2)	150	450	µA	HALT mode; V_{DD} = 3 V ± 10%
	I_{DD3}	(Notes 2, 5)	30	90	µA	V_{DD} = 3 V ± 10%
		(Notes 5, 8)	350	1000	µA	V_{DD} = 5 V ±10%
	I_{DD4}	(Notes 2, 5)	5	15	µA	HALT mode; V_{DD} = 3 V ± 10%
		(Notes 5, 8)	35	100	µA	HALT mode V_{DD} = 5 V ±10%
	I_{DD5}		0.5	20	µA	STOP mode; XT1 = 0 V; V_{DD} = 5.0 V ± 10%
		(Note 2)	0.1	10	µA	STOP mode; XT1 = 0 V; V_{DD} = 3.0 V ± 10%
		(Note 2)	0.1	5	µA	STOP mode; XT1 = 0 V; V_{DD} = 3.0 V ± 10%; T_A = 25°C

Notes:

(1) V_{DD} = 4.5 to 6.0 V for 7500x and V_{DD} = 4.5 to 5.5 V for 75P008.

(2) For 7500x only.

(3) Does not include pull-up resistor current.

(4) 4.19 MHz crystal oscillator; C1 = C2 = 22 pF.

(5) 32.768 kHz crystal oscillator.

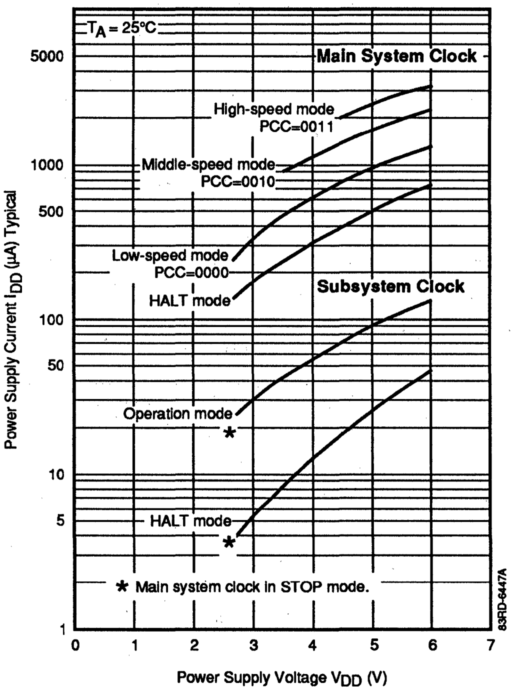
(6) When operated in the high-speed mode with the processor clock control register (PCC) set to 0011.

(7) When operated in the low-speed mode with the PCC set to 0000.

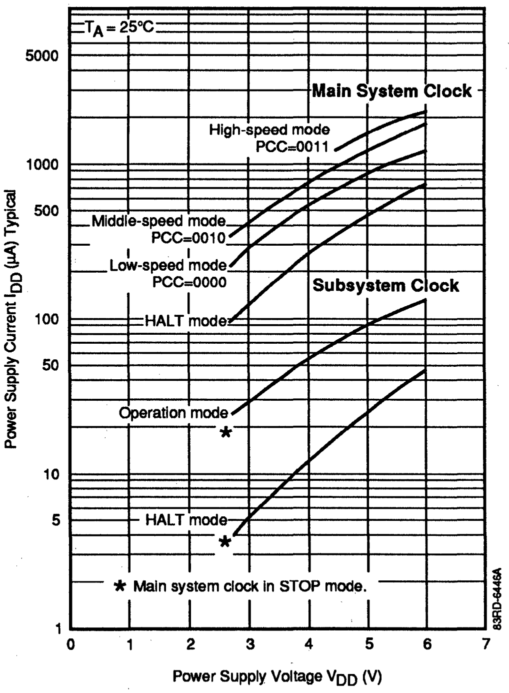
(8) For 75P008 only.

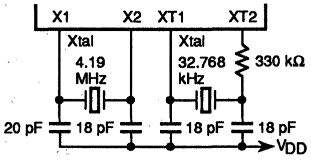
DC Characteristics

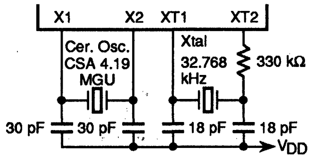
I_{DD} vs V_{DD} (Crystal Oscillator at 4.19 MHz)



I_{DD} vs V_{DD} (Ceramic Oscillator at 4.19 MHz)



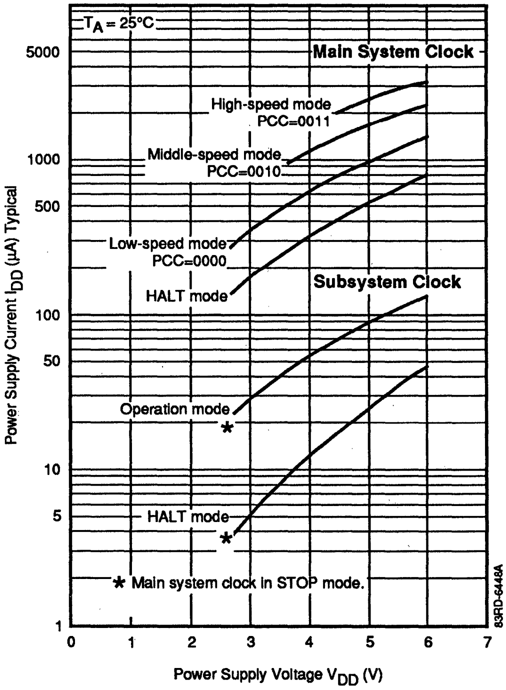




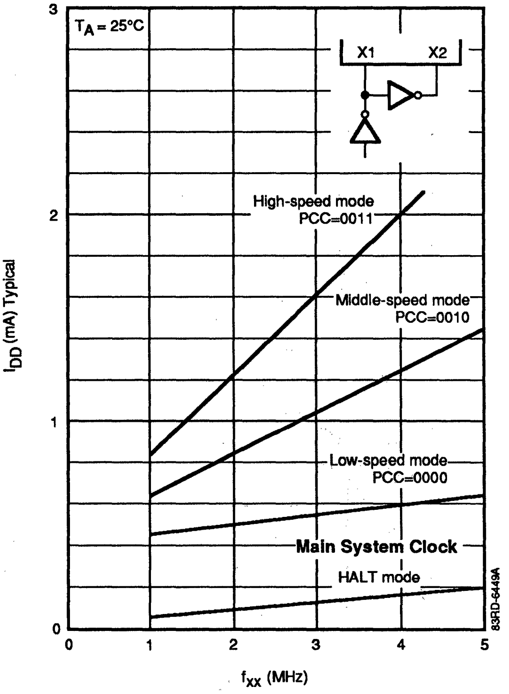
Note: Values of I_{DD} are about 10% larger using a ceramic oscillator as compared to a crystal oscillator.

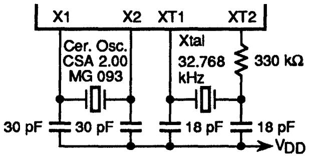
DC Characteristics (cont)

I_{DD} vs V_{DD} (Ceramic Oscillator at 2.00 MHz)

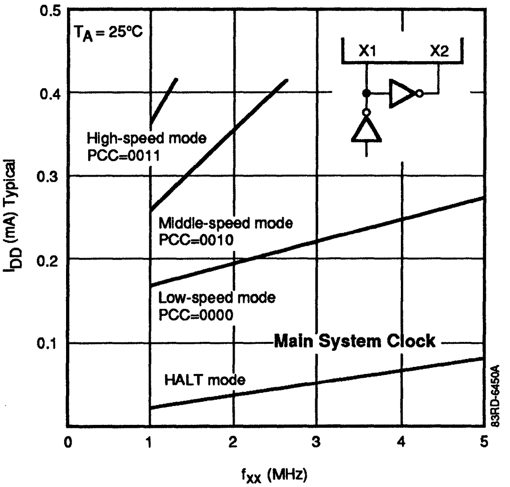


I_{DD} vs f_{XX} (V_{DD} = 5 V)

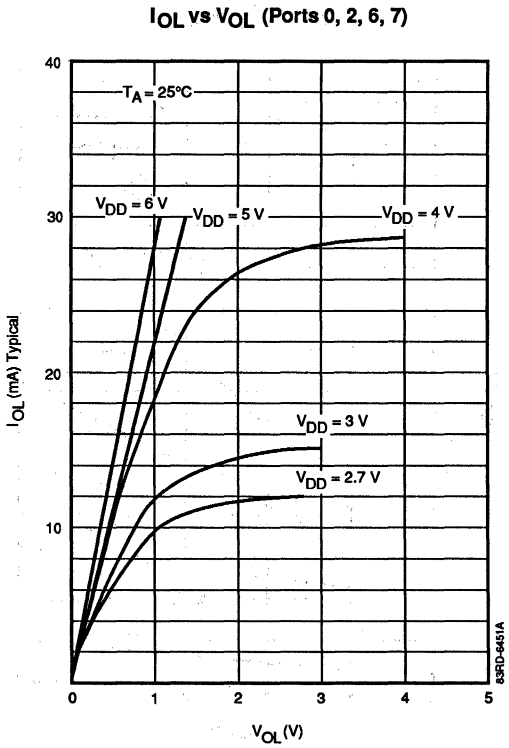


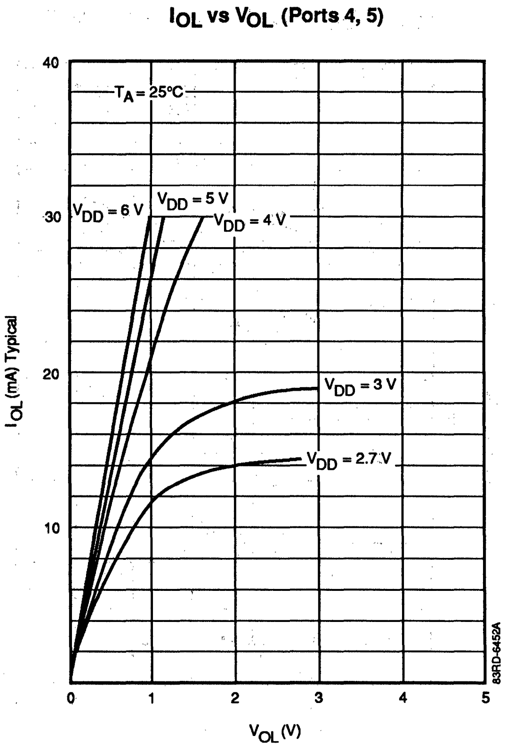


I_{DD} vs f_{XX} (V_{DD} = 3 V)



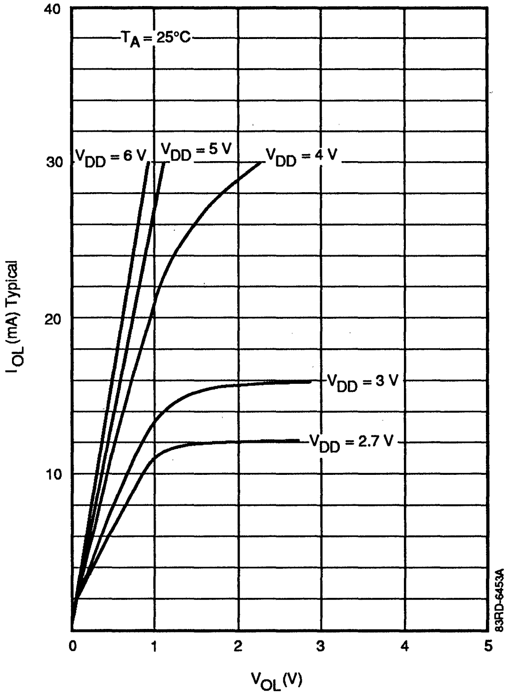
DC Characteristics (cont)



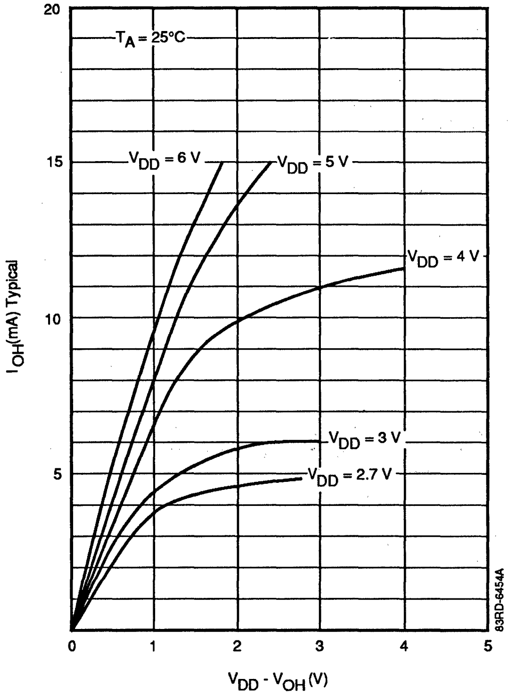


DC Characteristics (cont)

I_{OL} vs V_{OL} (Port 3)



I_{OH} vs V_{DD} - V_{OH}



AC Characteristics

μPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
μPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Note 1)	t_{CY}	0.95		64	μs	Main system clock (Note 2)
		3.8	(Note 3)	64	μs	Main system clock; V_{DD} = 2.7 to 6.0 V
		114	122	125	μs	Subsystem clock
TI0 input frequency	f_{TI}	0		1	MHz	(Note 2)
		0	(Note 3)	275	kHz	V_{DD} = 2.7 to 6.0 V
TI0 input low- and high-level width	t_{IH}, t_{IL}	0.48			μs	(Note 2)
		1.8	(Note 3)		μs	V_{DD} = 2.7 to 6.0 V
Interrupt inputs low- and high-level width	t_{INTH}, t_{INTL}	(Note 4)			μs	INT0
		10			μs	INT1, INT2, INT4
		10			μs	KR0-KR7
$\overline{\text{RESET}}$ low-level width	t_{RSL}	10			μs	

Notes:

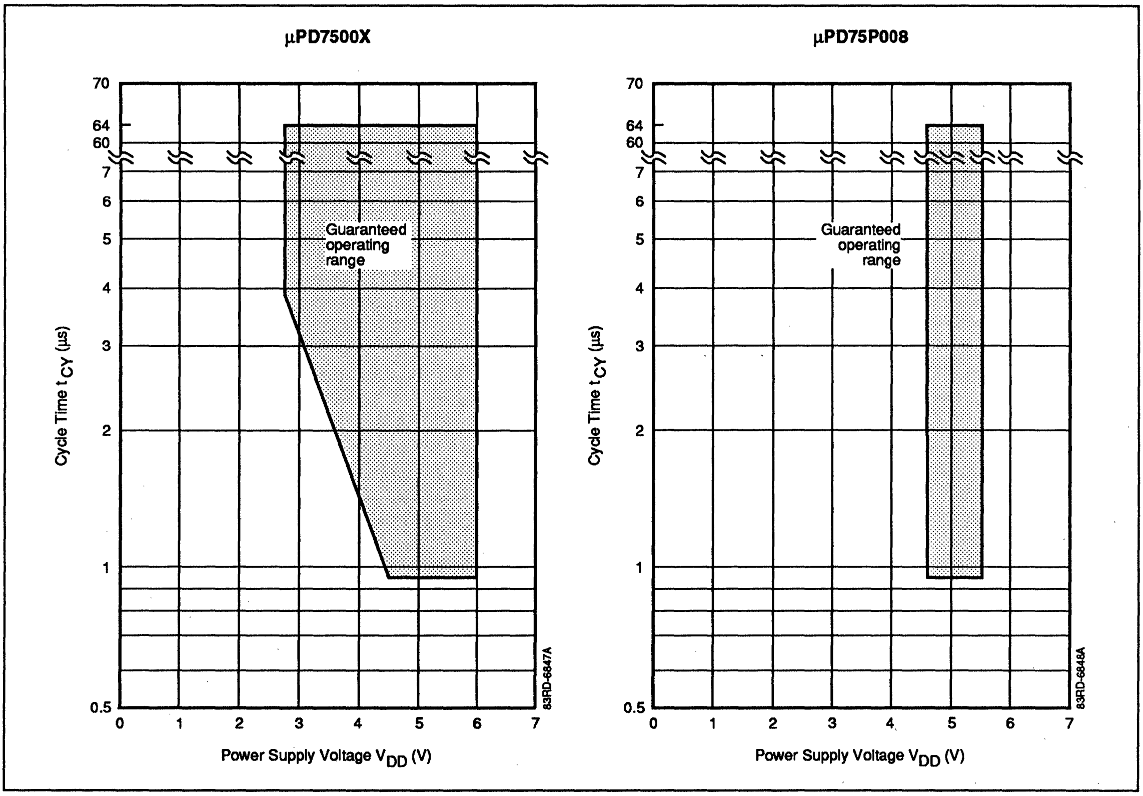
(1) Cycle time (minimum instruction execution time) is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC).

(2) V_{DD} = 4.5 to 6.0 V for 7500x and V_{DD} = 4.5 to 5.5 V for 75P008.

(3) For 7500x only.

(4) $2t_{CY}$ or 128/fx, depending on the setting of the interrupt mode register (IM0).

Figure 3. Guaranteed Operating Range



Serial Transfer Operation

2-line/3-line Serial I/O mode ($\overline{SCK}$...internal clock output)
µPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
µPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	1600			ns	(Note 1)
		3800	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KL1}, t_{KH1}	$0.5t_{KCY}-50$			ns	(Note 1)
		$0.5t_{KCY}-150$	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
SI vs. $\overline{SCK}$ ↑ setup time	t_{SIK1}	150			ns	
SI vs. $\overline{SCK}$ ↑ hold time	t_{KSI1}	400			ns	
$\overline{SCK}$ ↓ → SO output delay time (Note 3)	t_{KSO1}			250	ns	(Note 1)
			(Note 2)	1000	ns	V_{DD} = 2.7 to 6.0 V

Serial Transfer Operation

2-line/3-line Serial I/O mode ($\overline{SCK}$...external clock output)
µPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
µPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY2}	800			ns	(Note 1)
		3200	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KL2}, t_{KH2}	400			ns	(Note 1)
		1600	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
SI vs. $\overline{SCK}$ ↑ setup time	t_{SIK2}	100			ns	
SI vs. $\overline{SCK}$ ↑ hold time	t_{KSI2}	400			ns	
$\overline{SCK}$ ↓ → SO output delay time (Note 3)	t_{KSO2}			300	ns	(Note 1)
			(Note 2)	1000	ns	V_{DD} = 2.7 to 6.0 V

Notes:

(1) V_{DD} = 4.5 to 6.0 V for 7500x and V_{DD} = 4.5 to 5.5 V for 75P008.

(2) For 7500x only.

(3) The rising edge of the output delay time must be less than 600 ns. For example, if SB0 and SB1 are pulled up with 5 kΩ resistors, the total capacitance of the serial bus line must be no greater than 120 pF.

SBI Mode

$\overline{SCK}$...internal clock output (master)
µPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
µPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY3}	1600			ns	(Note 1)
		3800	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KL3}, t_{KH3}	$0.5t_{KCY3}-50$			ns	(Note 1)
		$0.5t_{KCY3}-150$	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
SB0, SB1 vs. $\overline{SCK}$ ↑ setup time	t_{SIK3}	150			ns	
SB0, SB1 vs. $\overline{SCK}$ ↑ hold time	t_{KSI3}	$0.5t_{KCY3}$			ns	
$\overline{SCK}$ ↓ → SB0, SB1 output delay time	t_{KSO3}	0		250	ns	(Note 1)
		0	(Note 2)	1000	ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ ↑ → SB0, SB1 ↓	t_{KSB}	t_{KCY3}			ns	
SB0, SB1 ↓ → $\overline{SCK}$ ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	

SBI Mode

$\overline{SCK}$...external clock output (slave)
µPD7500x: T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V
µPD75P008: T_A = −10 to +70°C, V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY4}	800			ns	(Note 1)
		3200	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low and high level width	t_{KL4}, t_{KH4}	400			ns	(Note 1)
		1600	(Note 2)		ns	V_{DD} = 2.7 to 6.0 V
SB0, SB1 vs. $\overline{SCK}$ ↑ setup time	t_{SIK4}	100			ns	
SB0, SB1 vs. $\overline{SCK}$ ↑ hold time	t_{KSI4}	$0.5t_{KCY4}$			ns	
$\overline{SCK}$ ↓ → SB0, SB1 output delay time	t_{KSO4}	0		300	ns	(Note 1)
		0	(Note 2)	1000	ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ ↑ → SB0, SB1 ↓	t_{KSB}	t_{KCY4}			ns	
SB0, SB1 ↓ → $\overline{SCK}$ ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	

Notes:

(1) V_{DD} = 4.5 to 6.0 V for 7500x and V_{DD} = 4.5 to 5.5 V for 75P008.

(2) For 7500x only.

Data Memory STOP Mode Low Voltage Data Retention Characteristics

μPD7500x: T_A = −40 to +85°C
μPD75P008: T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		(Note 1)	V	
Data retention current (Note 2)	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V
Release signal SET time	t_{SREL}	0			μs	
Oscillation stabilization time (Note 3)	t_{WAIT}		$2^{17}/f_{XX}$		s	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

(1) Max = 6.0 V for 7500x and 5.5 V for 75P008.

(2) Pull-up resistor current, comparator circuit current, and power-on-reset current is not included in this table.

(3) Oscillation stabilization WAIT time is the time during which the CPU is stopped to prevent unstable operation when the oscillation is started. WAIT time depends on the resonator vendor's specifications. The wait time generated by the chip should be ≧ vendor's spec and the setting of the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{XX} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{XX}$ (approx 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (approx 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (approx 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (approx 1.95 ms)

DC Programming Characteristics (For 75P008 only)

T_A = 25 ±5°C; V_{DD} = 6.0 ±0.25 V; V_{PP} = 12.5 ±0.3 V; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	$0.7V_{DD}$		V_{DD}	V	All except X1, X2
	V_{IH2}	V_{DD}−0.5		V_{DD}	V	X1, X2
Low-level input voltage	V_{IL1}	0		$0.3V_{DD}$	V	All except X1, X2
	V_{IL2}	0		0.4	V	X1, X2
Input leakage current	I_{LI}			10	μA	V_{IN} = V_{IL} or V_{IH}
High-level output voltage	V_{OH}	V_{DD}−1.0			V	I_{OH} = −1 mA
Low-level output voltage	V_{OL}			0.4	V	I_{OL} = 1.6 mA
V_{DD} supply current	I_{DD}			30	mA	
V_{PP} supply current	I_{PP}			30	mA	MD0 = V_{IL}; MD1 = V_{IH}

Notes:

(1) V_{PP} must not exceed +13.5 V, including over shoot.

(2) V_{DD} must be applied before V_{PP} and is turned off after V_{PP} is removed.

AC Programming Characteristics (For 75P008 only)

T_A = 25 ± 5°C; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V

Parameter	Symbol	(Note 1)	Min	Max	Unit	Conditions
Address setup time (Note 2)	t_{AS}	t_{AS}	2		μs	
MD1 to MD0 ↓ setup	t_{M1S}	t_{OES}	2		μs	
Data to MD0 ↓ setup	t_{DS}	t_{DS}	2		μs	
Address hold from MD0 ↑ (Note 2)	t_{AH}	t_{AH}	2		μs	
Data hold from MD0 ↑	t_{DH}	t_{DH}	2		μs	
Data output float delay from MD0 ↑	t_{DF}	t_{DF}	0	130	ns	
V_{PP} setup to MD3 ↑	t_{VPS}	t_{VPS}	2		μs	
V_{DD} setup to MD3 ↑	t_{VDS}	t_{VCS}	2		μs	
Initialized program pulse width	t_{PW}	t_{PW}	0.95	1.05	ms	
Additional program pulse width	t_{OPW}	t_{OPW}	0.95	21	ms	
MD0 setup to MD1 ↑	t_{MOS}	t_{CES}	2		μs	
Data output delay from MD0 ↓	t_{DV}	t_{DV}		1	μs	MD0 = MD1 = V_{IL}
MD1 hold to MD0 ↑	t_{M1H}	t_{OEH}	2		μs	$t_{M1H} + t_{M1R} \geq 50$ μs
MD1 recovery from MD0 ↓	t_{M1R}	t_{OR}	2		μs	$t_{M1H} + t_{M1R} \geq 50$ μs
Program counter reset	t_{PCR}		10		μs	
X1 input low- and high-level width	t_{XH}, t_{XL}		0.125		μs	
X1 input frequency	f_X			4.19	MHz	
Initial mode set	t_I		2		μs	
MD3 setup to MD1 ↑	t_{M3S}		2		μs	
MD3 hold from MD1 ↓	t_{M3H}		2		μs	
MD3 setup to MD0 ↓	t_{M3SR}		2		μs	During program read cycle
Data delay from address (Note 2)	t_{DAD}	t_{ACC}	2		μs	
Data output hold from address (Note 2)	t_{HAD}	t_{OH}	0	130	ns	
MD3 output hold from MD0 ↑	t_{M3HR}		2		μs	
Data output float delay from MD3 ↓	t_{DFR}		2		μs	

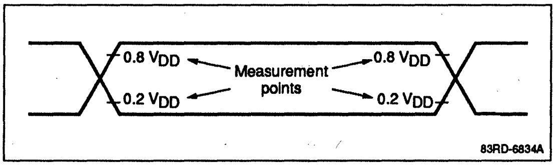
Notes:

(1) These symbols correspond to those of the μPD27C256 EPROM.

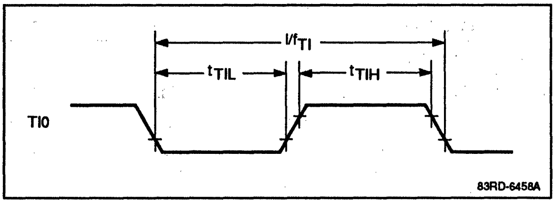
(2) The internal address signal is incremented by the rising edge of the fourth X1 pulse; it is not connected to an external pin.

Timing Waveforms

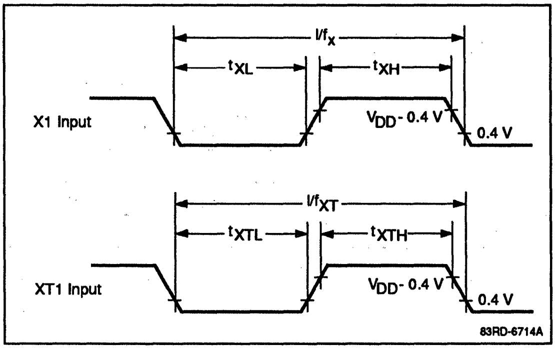
AC Timing Measurements Points (Except X1 and XT1)



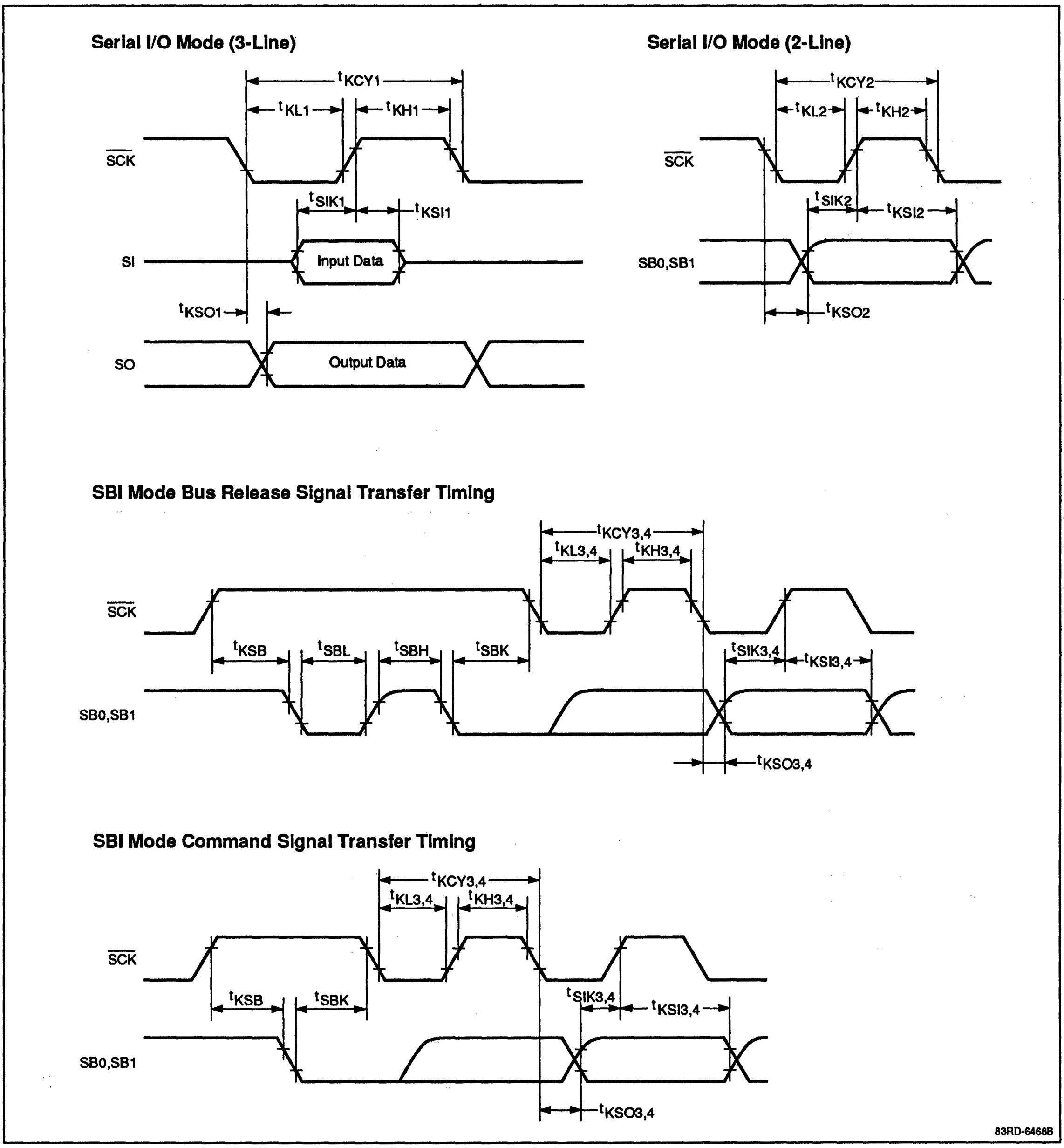
TI0 Timing



Clock Timing



Serial Transfer Timing


Serial I/O Mode (3-Line)

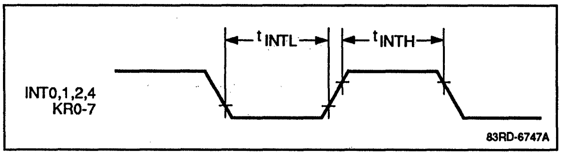
Serial I/O Mode (2-Line)

SBI Mode Bus Release Signal Transfer Timing

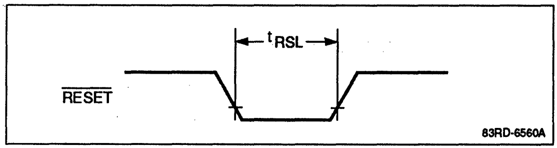
SBI Mode Command Signal Transfer Timing

83RD-6468B

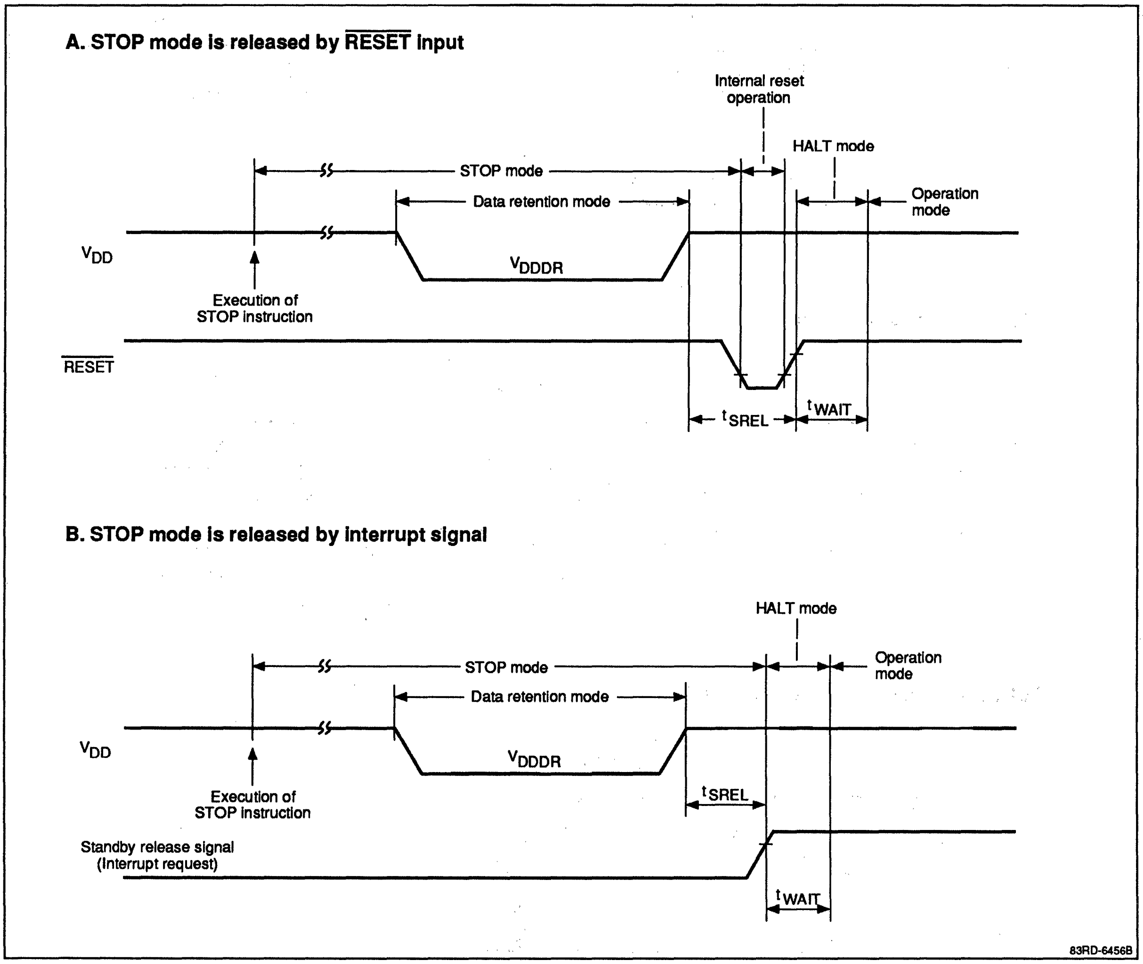
Interrupt Input Timing



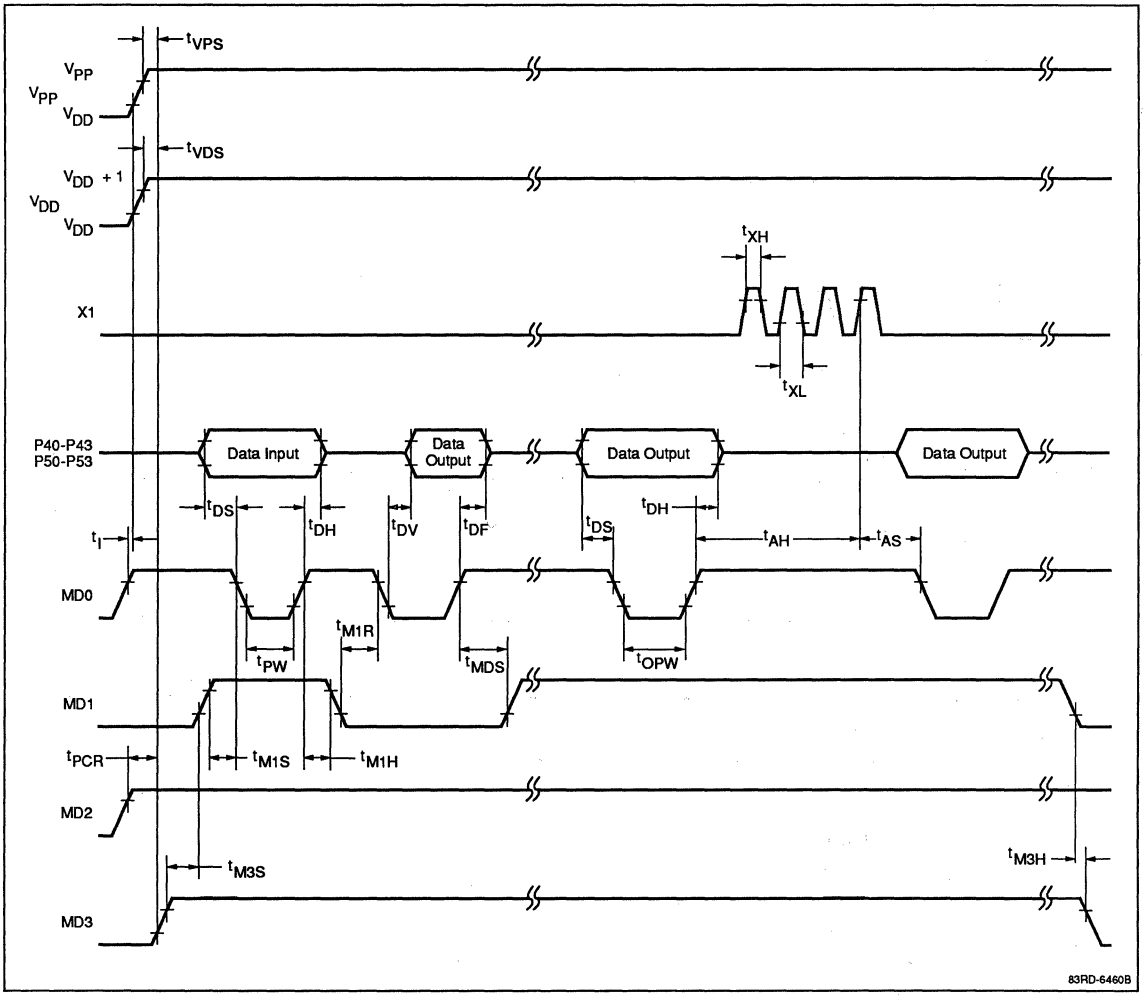
RESET Input Timing



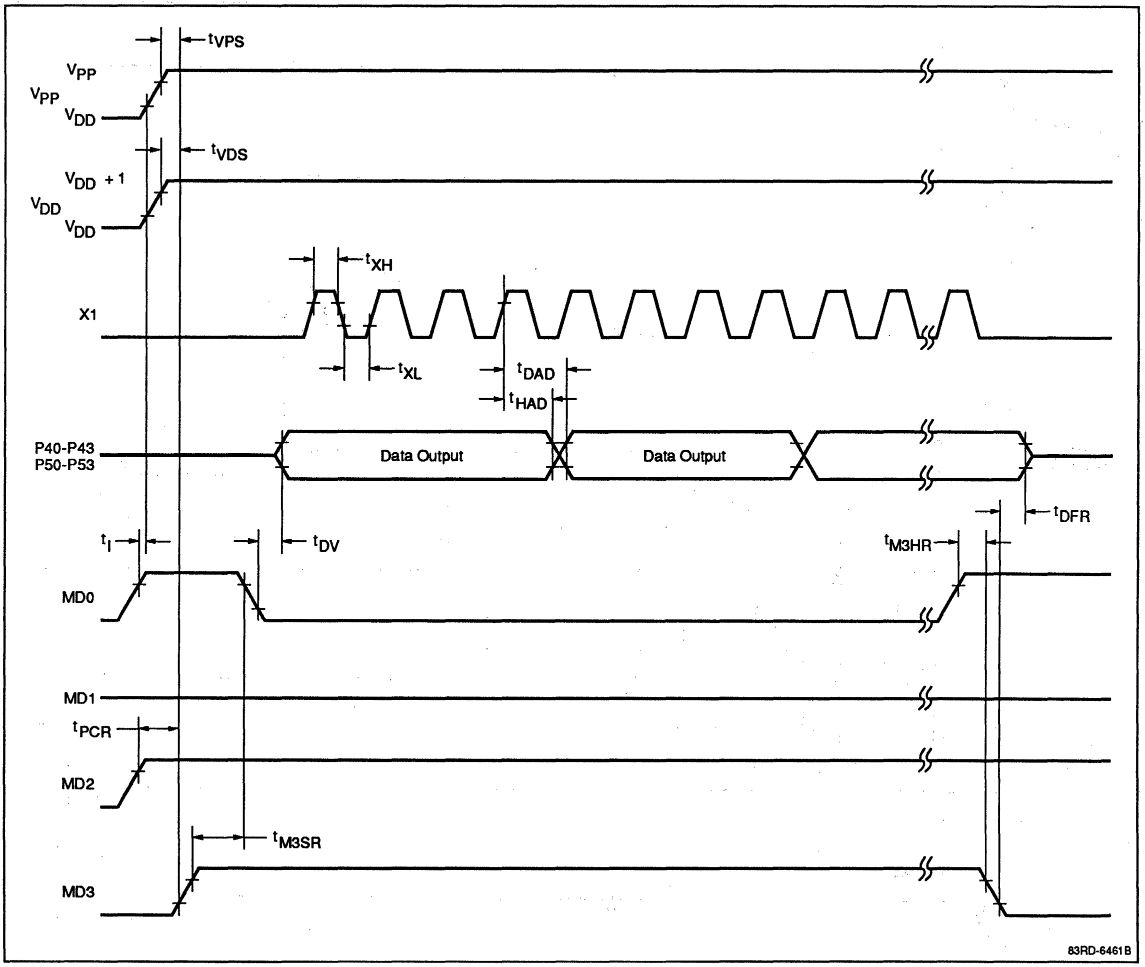
Data Retention Timing


A. STOP mode is released by RESET input
Internal reset operation
HALT mode
STOP mode
Data retention mode
Operation mode
VDD
VDDDR
Execution of STOP instruction
RESET
tSREL
tWAIT
B. STOP mode is released by interrupt signal
HALT mode
STOP mode
Operation mode
Data retention mode
VDD
VDDDR
Execution of STOP instruction
tSREL
Standby release signal (Interrupt request)
tWAIT
83RD-6456B

Program Memory Write/Verify Timing (μPD75P008)



Program Memory Read Timing (μPD75P008)



NEC
NEC Electronics Inc.

μPD75028/75P036 General-Purpose 4-Bit Microcomputers With A/D Converter

Description

The μPD75028/P036 are high-performance single-chip CMOS microcomputers containing CPU, ROM, RAM, I/O ports, several timer/counters, A/D converter, vectored interrupts, subsystem clock, and serial interface.

The instruction set allows the user to manipulate RAM data and I/O ports in 1-, 4-, and 8-bit units. The devices are ideally suited for controlling VCRs, telephones, and meters.

Features

- □ 103 instructions
  - — Bit manipulation
  - — 4-bit and 8-bit transfer
  - — GETI instruction, to convert one 2-byte or two 1-byte instructions into a single 1-byte instruction
  - — 1-byte relative branch
- □ Fast execution time (@ 4.19 MHz)
  - — High-speed cycle: 0.95 μs
  - — Lower-voltage cycles: 1.91 and 15.3 μs
- □ 8064 bytes of program ROM: μPD75028
- □ 16256 bytes of program ROM: μPD75P036
- □ 512 x 4 bits of RAM μPD75028
- □ 1024 x 4 bits of RAM μPD75P036
- □ Bit sequential buffer
  - — 16-bit, bit manipulation memory
- □ Eight 4-bit registers
- □ Accumulators
  - — 1-bit (CY)
  - — 4-bit (A)
  - — 8-bit (XA)
- □ 40 I/O lines
  - — 12 N-channel open drain; can withstand 10 V
  - — 12 outputs directly drive LEDs
- □ 8 input-only lines
- □ One external event input
- □ Four timers
  - — 8-bit basic interval timer
  - — 8-bit timer/event counter
  - — 14-bit watch timer
  - — 16-bit multifunction timer/event counter which can be used as an 8-bit timer/event counter, PWM output, 16-bit free-running timer, or 16-bit counter for an integrating A/D converter
- □ A/D converter
  - — 8-channel, 8-bit
  - — Reference voltage can be between AV_{REF+} and AV_{REF-}
- □ Four zero cross detection pins
- □ 8-bit serial interface
  - — SBI mode
  - — 2- or 3-wire mode: data transfer can be full duplex or receive only, and can be MSB or LSB first
- □ Vectored interrupts
  - — Three external interrupts
  - — Four internal interrupts
  - — Nine inputs which each generate one interrupt request
- □ Standby modes
  - — HALT mode: stops CPU only
  - — STOP mode: stops main clock generator
- □ Operates with oscillator or ceramic resonator
- □ OTP version: μPD75P036
  - — 16256 bytes of program ROM
  - — 1024 x 4 bits of RAM
- □ CMOS operation, with V_{DD} from 2.7 to 6.0 V

Ordering Information

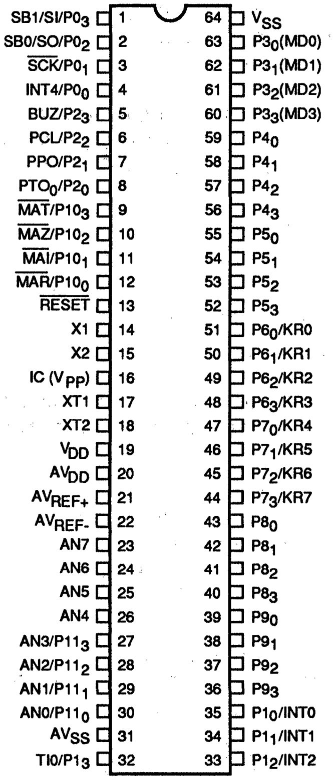
Part Number	Package Type	ROM
μPD75028CW-xxx	64-pin plastic SDIP	Mask ROM
μPD75028GC-xxx-AB8	64-pin plastic QFP	Mask ROM
μPD75P036CW-xxx	64-pin plastic SDIP	OTP
μPD75P036GC-xxx-AB8	64-pin plastic QFP	OTP

Notes:

(1) xxx indicates ROM code

Pin Configurations

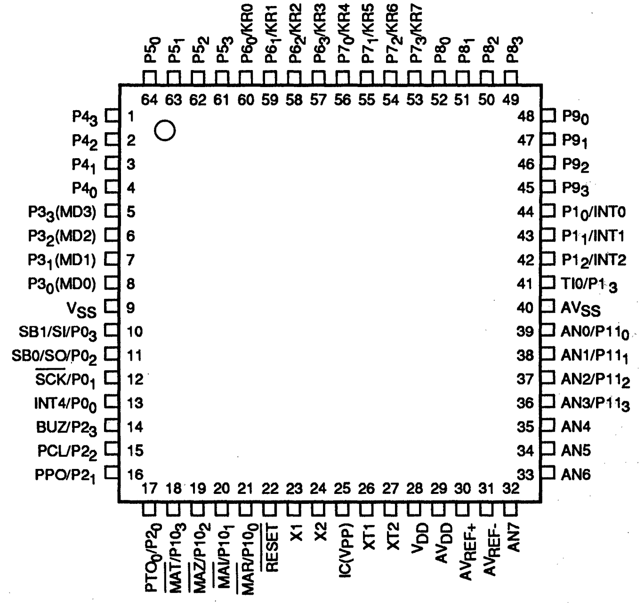
64-Pin Shrink DIP



IC: Should be connected to V$_{DD}$
Used as V$_{PP}$ programming pin in μPD75P036.

83RD-6622A

64-Pin QFP



IC: Should be connected to V$_{DD}$. Used as V$_{PP}$ programming pin in µPD75P036.

83RD-6623B

Pin Identification

Symbol	Function
AN_4-AN_7	Inputs for A/D converter
AV_{DD}	A/D converter positive power supply
AV_{SS}	A/D converter ground
AV_{REF+} AV_{REF-}	A/D converter reference voltages
IC (V_{PP})	Internally connected (V_{PP} for μPD75P036)
$P0_0$/INT4	Port 0 input; interrupt 4
$P0_1/\overline{SCK}$	Port 0 input; serial clock
$P0_2$/SO/SB0	Port 0 input; serial out; serial interface
$P0_3$/SI/SB1	Port 0 input; serial in; serial interface
$P1_0$/INT0	Port 1 input; interrupt 0
$P1_1$/INT1	Port 1 input; interrupt 1
$P1_2$/INT2	Port 1 input; interrupt 2
$P1_3/TI_0$	Port 1 input; timer 0 input
$P2_0/PTO_0$	Port 2 I/O; timer/event counter output
$P2_1$/PPO	Port 2 I/O; multifunction timer output
$P2_2$/PCL	Port 2 I/O; clock output
$P2_3$/BUZ	Port 2 I/O; buzzer output
$P3_0$-$P3_3$	Port 3 I/O
$P4_0$-$P4_3$	Port 4 I/O
$P5_0$-$P5_3$	Port 5 I/O
$P6_0$/KR0	Port 6 I/O; key scan input 0
$P6_1$/KR1	Port 6 I/O; key scan input 1
$P6_2$/KR2	Port 6 I/O; key scan input 2
$P6_3$/KR3	Port 6 I/O; key scan input 3
$P7_0$/KR4	Port 7 I/O; key scan input 4
$P7_1$/KR5	Port 7 I/O; key scan input 5
$P7_2$/KR6	Port 7 I/O; key scan input 6
$P7_3$/KR7	Port 7 I/O; key scan input 7
$P8_0$-$P8_3$	Port 8 I/O
$P9_0$-$P9_3$	Port 9 I/O
$P10_0/\overline{MAR}$	Port 10 I/O; multifunction timer/event counter output
$P10_1/\overline{MAI}$	Port 10 I/O; multifunction timer/event counter output
$P10_2/\overline{MAZ}$	Port 10 I/O; multifunction timer/event counter output
$P10_3/\overline{MAT}$	Port 10 I/O; multifunction timer/event counter input
$P11_0/\overline{AN0}$	Port 11 input; A/D converter input 0
$P11_1/\overline{AN1}$	Port 11 input; A/D converter input 1
$P11_2/\overline{AN2}$	Port 11 input; A/D converter input 2
$P11_3/\overline{AN3}$	Port 11 input; A/D converter input 3
$\overline{RESET}$	Reset input
X1, X2	Main clock inputs
XT1, XT2	Subsystem clock inputs
V_{DD}	Positive power supply
V_{SS}	Ground

PIN FUNCTIONS

$P0_0$/INT4, $P0_1/\overline{SCK}$, $P0_2$/SO/SB0, $P0_3$/SI/SB1

These pins can be used as 4-bit input port 0. Or, $P0_0$ can also be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. $P0_1$-$P0_3$ may also be used for the serial interface in the SBI or 2- or 3-wire mode. SI is the serial input, SO is the serial output, and $\overline{SCK}$ is the serial clock. Reset causes these pins to default to the Port 0 input mode.

$P1_0$/INT0, $P1_1$/INT1, $P1_2$/INT2, $P1_3$/TI0

These pins can be used as 4-bit input port 1. Or, $P1_0$ and $P1_1$ can also be used for edge-triggered interrupts INT0 and INT1. $P1_2$ can be used for INT2, which is also an edge-triggered input, but one which generates an interrupt request and does not cause an interrupt. $P1_3$ can be used as an input clock to the timer/event counter to count external events. Reset causes these pins to default to the Port 1 input mode.

$P2_0$/PTO0, $P2_1$/PPO, $P2_2$/PCL, $P2_3$/BUZ

These pins can be used as 4-bit I/O port 2. When used as an output the data is latched. When used as an input port, the port outputs are three-state. $P2_0$ can also be used as the output of the timer/event counter flip flop (TOUT); $P2_1$ can also be used as the output for the multifunction timer/event counter T flip flop; $P2_2$ can be used as the output (PCL) for the clock generator; and $P2_3$ can be used to output square waves for a buzzer. Reset causes these pins to default to the Port 2 input mode.

$P3_0$-$P3_3$

These pins are used for I/O port 3. Each bit in this port can be independently programmed to be either an input or an output. This port has latched outputs, and can directly drive LEDs. A reset signal causes this port to default to the input mode.

$P4_0$-$P4_3$, $P5_0$-$P5_3$

Port 4 and 5 are identical 4-bit I/O ports which can be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. Outputs are N-channel open drain, and can withstand up to 10 volts; pull-up resistor mask options are available for these ports. A reset signal causes these ports to default to the input mode.

$P6_0$/KR0, $P6_1$/KR1, $P6_2$/KR2, $P6_3$/KR3 $P7_0$/KR4, $P7_1$/KR5, $P7_2$/KR6, $P7_3$/KR7

Ports 6 and 7 are 4-bit I/O ports with latched outputs. Each pin of port 6 can be independently programmed to be either an input or an output, while port 7 can be programmed to be either all inputs or all outputs. Alternately, these pins may be used to detect the falling edge of inputs KR0-KR3 (port 6) and KR4-KR7 (port 7). A reset signal causes these ports to default to the input mode.

$P8_0$-$P8_3$, $P9_0$-$P9_3$

Ports 8 and 9 are identical 4-bit I/O ports. Outputs are latched. A reset signal causes these ports to default to the input mode.

$P10_0$/$\overline{MAR}$, $P10_1$/$\overline{MAI}$, $P10_2$/$\overline{MAZ}$, $P10_3$/$\overline{MAT}$

These pins are used for I/O port 10. Outputs are N-channel open drain which can withstand up to 10 volts. $P10_0$-$P10_2$ can also be used as the $\overline{MAR}$, $\overline{MAI}$, and $\overline{MAZ}$ outputs from the multifunction timer/event counter's A/D control logic. $P10_3$ can be used as the input $\overline{MAT}$ to the multifunction timer/event counter's A/D control logic. A reset signal causes this port to default to the input mode.

$P11_0$/AN0, $P11_1$/AN1, $P11_2$/AN2, $P11_3$/AN3

These pins are used for I/O port 11, or can alternately be used as A/D converter inputs AN0-AN3. A reset signal causes this port to default to the input mode.

AN4-AN7

A/D converter inputs AN4-AN7.

AV_{DD}

A/D converter positive power supply.

AV_{SS}

A/D converter analog ground.

AV_{REF+}, AV_{REF-}

A/D converter positive and negative reference voltages.

IC/V_{PP}

This pin should be connected to V_{DD} when using the μPD75028. For the μPD75P036, this pin is used as the programming voltage input during the EPROM write/verify cycles. When the device is not being programmed, this pin should be tied to V_{DD}.

X1, X2

These pins are the main system clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

XT1, XT2

These pins are the subsystem clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

$\overline{RESET}$

This is the reset input, and it is active low.

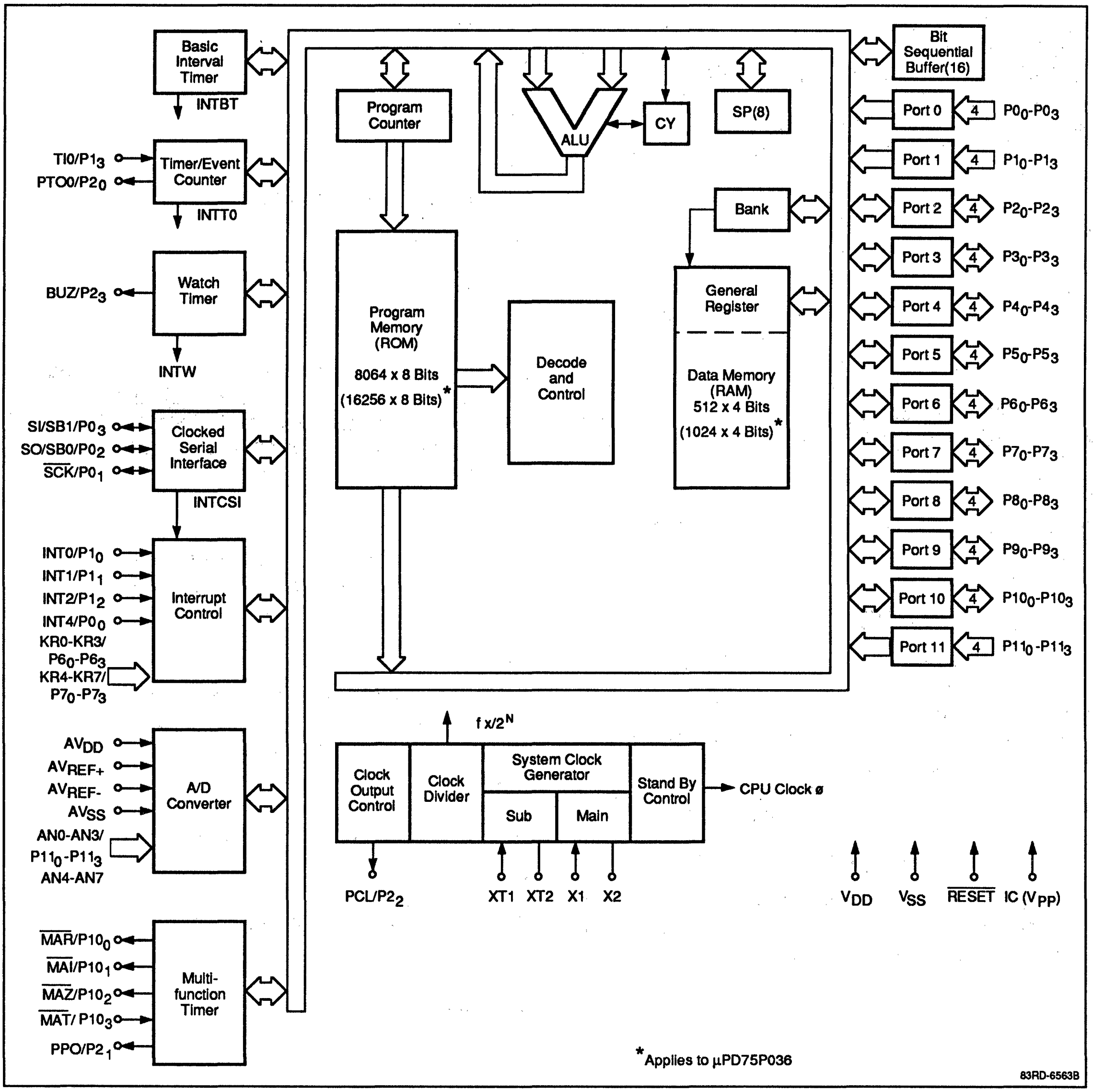
V_{DD}

The system positive power supply pin.

V_{SS}

System ground.

Block Diagram



Specifications

ROM	8064 bytes (μPD75028)
	16256 bytes (μPD75P036)
RAM	512 x 4 bits (μPD75028)
	1024 x 4 bits (μPD75P036)
General-purpose registers	4 bits x 8 or 8 bits x 4
Instruction cycle	0.95 μs/1.91 μs/15.3 μs (with main system clock operating at 4.19 MHz)
	122 μs (with subsystem clock operating at 32 kHz)
I/O ports	48 total lines. There are 12 N-channel open-drain I/O ports, each tolerating as much as 10 volts. Pull-up resistor mask-option is available in the μPD75028 only. The remaining 36 lines are standard CMOS, including 12 input ports and 24 I/O ports. Of these, 27 have software-selectable pullup resistors, and four have software-selectable pulldown resistors.
A/D converter	8-bit x 8-channel
	Low voltage operation possible (V_{DD} = 2.7 to 6.0 V)

Timer/Counter	Three timers. These include an 8-bit timer/event counter, an 8-bit basic interval timer, and a clock timer.
Multifunction timer	This can be used as an 8-bit timer/event counter, PWM output, 16-bit free-running timer, or 16-bit counter for an integrating A/D converter.
Serial interface	NEC standard serial bus interface (SBI)
	Clock serial interface
External interrupts	Three vector interrupts, one test input.
Internal interrupts	Four vector interrupts, one test input.
Bit sequential buffer	16-bit, on-chip
Clock output (PCL)	CPU clock Φ: 524 kHz, 262 kHz, 65.6 kHz (with main system clock operating at 4.19 MHz)
Buzzer output (BUZ)	2 kHz, 4 kHz, 32 kHz (with subsystem clock operating at 32.768 kHz)
Package	64-pin plastic shrink DIP (750 mil)
	64-pin plastic QFP (14 x 14 mm)
Operating voltage	V_{DD} = 2.7 to 6.0 V

NEC
NEC Electronics Inc.

μPD75048/75P056 General-Purpose 4-Bit Microcomputers With EEPROM and A/D Converter

Description

The μPD75048 is a single-chip CMOS microcomputer containing CPU, ROM, EEPROM, RAM, I/O ports, several timer/counters, A/D converter, vectored interrupts, subsystem clock, and serial interface.

The instruction set allows the user to manipulate RAM data and I/O ports in 1-, 4-, and 8-bit units. The devices are ideally suited for controlling devices which require EEPROM, such as meters requiring individual calibration.

Features

- □ 103 instructions
  - — Bit manipulation
  - — 4-bit and 8-bit transfer
  - — GETI instruction, to convert one 2-byte or two 1-byte instructions into a single 1-byte instruction
  - — 1-byte relative branch instruction
- □ Fast execution time (@ 4.19 MHz)
  - — High-speed cycle: 0.95 μs
  - — Lower-voltage cycles: 1.91 and 15.3 μs
- □ 8064 bytes of program ROM: μPD75048
- □ 16256 bytes of program ROM: μPD75P056
- □ 1024 x 4 bits of EEPROM
- □ 512 x 4 bits of RAM
  - — Allows operation on 1, 4, or 8 bits
- □ Bit sequential buffer
  - — 16-bit, bit manipulation memory
- □ Eight 4-bit registers
- □ Accumulators
  - — 1-bit (CY)
  - — 4-bit (A)
  - — 8-bit (XA)
- □ 48 I/O lines
  - — 12 N-channel open drain; can withstand 10 V
  - — 12 outputs directly drive LEDs
  - — 43 lines can have an on-chip pullup/pulldown resistor
- □ One external event input
- □ Four timers
  - — 8-bit basic interval timer
  - — 8-bit timer/event counter
  - — 14-bit watch timer
  - — 16-bit multifunction timer/event counter which can be used as an 8-bit timer/event counter, PWM output, 16-bit free-running timer, or 16-bit counter for an integrating A/D converter
- □ A/D converter
  - — 8-channel, 8-bit
- □ Four zero cross detection pins
- □ 8-bit serial interface
  - — SBI mode
  - — 2- or 3-wire mode: data transfer can be full duplex or receive only, and can be MSB or LSB first
- □ Vectored interrupts
  - — Three external interrupts
  - — Four internal interrupts
  - — Nine inputs which each generate one interrupt request
- □ Standby modes
  - — HALT mode: stops CPU only
  - — STOP mode: stops main clock generator
- □ Operates with oscillator or ceramic resonator
- □ OTP version: μPD75P056
- □ CMOS operation, with V_{DD} from 2.7 to 6.0 V

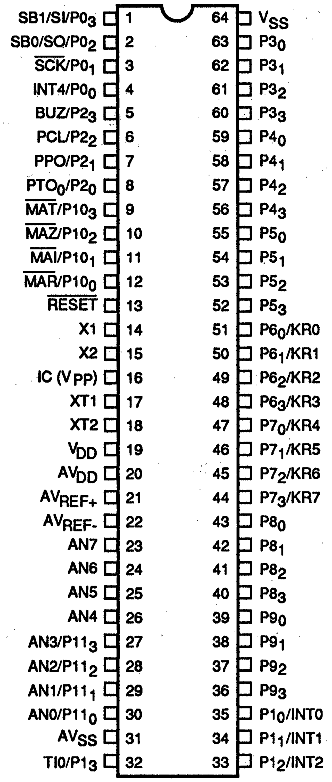
Ordering Information

Part Number	Package Type	ROM
μPD75048CW-xxx	64-pin plastic SDIP	Mask ROM
μPD75048GC-xxx-AB8	64-pin plastic QFP	Mask ROM
μPD75P056CW*	64-pin plastic SDIP	OTP
μPD75P056GC-AB8*	64-pin plastic QFP	OTP

*Under Development

Pin Configurations

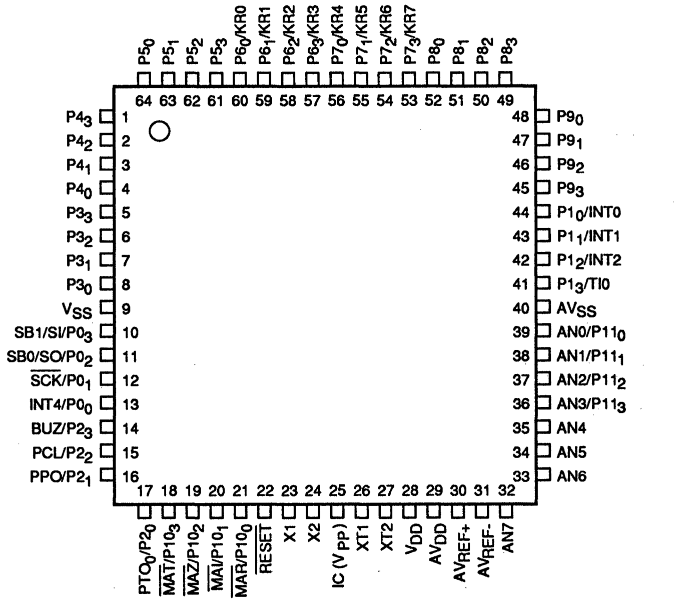
64-Pin SDIP



IC: Should be connected to V_{DD}. Used as V_{PP} program pin in μPD75P056

83RD-6620A

64-Pin QFP



IC: Should be connected to V_{DD}. Used as V_{PP} programming pin in μPD75P056

83RD-6621B

Pin Identification

Symbol	Function
$P0_0$/INT4	Port 0 input; interrupt 4
$P0_1$/$\overline{\text{SCK}}$	Port 0 input; serial clock
$P0_2$/SO/SB0	Port 0 input; serial out; serial interface
$P0_3$/SI/SB1	Port 0 input; serial in; serial interface
$P1_0$/INT0	Port 1 input; interrupt 0
$P1_1$/INT1	Port 1 input; interrupt 1
$P1_2$/INT2	Port 1 input; interrupt 2
$P1_3$/TI0	Port 1 input; timer 0 input
$P2_0$/PTO0	Port 2 I/O; timer/event counter output
$P2_1$/PPO	Port 2 I/O; multifunction timer output
$P2_2$/PCL	Port 2 I/O; clock output
$P2_3$/BUZ	Port 2 I/O; buzzer output
$P3_0$-$P3_3$	Port 3 I/O
$P4_0$-$P4_3$	Port 4 I/O
$P5_0$-$P5_3$	Port 5 I/O
$P6_0$/KR0	Port 6 I/O; key scan input 0
$P6_1$/KR1	Port 6 I/O; key scan input 1
$P6_2$/KR2	Port 6 I/O; key scan input 2
$P6_3$/KR3	Port 6 I/O; key scan input 3
$P7_0$/KR4	Port 7 I/O; key scan input 4
$P7_1$/KR5	Port 7 I/O; key scan input 5
$P7_2$/KR6	Port 7 I/O; key scan input 6
$P7_3$/KR7	Port 7 I/O; key scan input 7
$P8_0$-$P8_3$	Port 8 I/O
$P9_0$-$P9_3$	Port 9 I/O
$P10_0$/$\overline{\text{MAR}}$	Port 10 I/O; multifunction timer/event counter output
$P10_1$/$\overline{\text{MAI}}$	Port 10 I/O; multifunction timer/event counter output
$P10_2$/$\overline{\text{MAZ}}$	Port 10 I/O; multifunction timer/event counter output
$P10_3$/$\overline{\text{MAT}}$	Port 10 I/O; multifunction timer/event counter input
$P11_0$/AN0	Port 11 I/O; A/D converter input 0
$P11_1$/AN1	Port 11 I/O; A/D converter input 1
$P11_2$/AN2	Port 11 I/O; A/D converter input 2
$P11_3$/AN3	Port 11 I/O; A/D converter input 3
AN4-AN7	A/D converter inputs 4-7
AV_{DD}	A/D converter positive power supply
AV_{SS}	A/D converter ground
AV_{REF+} AV_{REF-}	A/D converter reference voltages

Symbol	Function
IC (V_{PP})	Internally connected (Programming voltage for μPD75P056)
X1, X2	Main clock inputs
XT1, XT2	Subsystem clock inputs
$\overline{\text{RESET}}$	Reset input
V_{DD}	Positive power supply
V_{SS}	Ground

PIN FUNCTIONS

$P0_0$/INT4, $P0_1$/$\overline{\text{SCK}}$, $P0_2$/SO/SB0, $P0_3$/SI/SB1

These pins can be used as 4-bit input port 0. Or, $P0_0$ can also be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. $P0_1$-$P0_3$ may also be used for the serial interface in the SBI or 2- or 3-wire mode. SI is the serial input, SO is the serial output, and $\overline{\text{SCK}}$ is the serial clock. Reset causes these pins to default to the port 0 input mode.

$P1_0$/INT0, $P1_1$/INT1, $P1_2$/INT2, $P1_3$/TI0

These pins can be used as 4-bit input port 1. Or, $P1_0$ and $P1_1$ can also be used for edge-triggered interrupts INT0 and INT1. $P1_2$ can be used for INT2, which is also an edge-triggered input, but one which generates an interrupt request and does not cause an interrupt. $P1_3$ can be used as an input clock to the timer/event counter to count external events. Reset causes these pins to default to the port 1 input mode.

$P2_0$/PTO0, $P2_1$/PPO, $P2_2$/PCL, $P2_3$/BUZ

These pins can be used as 4-bit I/O port 2. When used as an output, the data is latched. When used as an input port, the port outputs are three-state. $P2_0$ can also be used as the output of the timer/event counter flip flop (TOUT); $P2_1$ can also be used as the output for the multifunction timer/event counter T flip flop; $P2_2$ can be used as the output (PCL) of the clock generator; and $P2_3$ can be used to output square waves for a buzzer. Reset causes these pins to default to the port 2 input mode.

$P3_0$-$P3_3$

These pins are used for I/O port 3. Each bit in this port can be independently programmed to be either an input or an output. This port has latched outputs, and can directly drive LEDs. A reset signal causes this port to default to the input mode.

$P4_0$-$P4_3$, $P5_0$-$P5_3$

Port 4 and port 5 are identical 4-bit I/O ports which can be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. Outputs are N-channel open drain, and can withstand up to 10 volts; pullup resistor mask options are available for these ports. A reset signal causes these ports to default to the input mode.

$P6_0$/KR0, $P6_1$/KR1, $P6_2$/KR2, $P6_3$/KR3 $P7_0$/KR4, $P7_1$/KR5, $P7_2$/KR6, $P7_3$/KR7

Ports 6 and 7 are 4-bit I/O ports with latched outputs. Each pin of port 6 can be independently programmed to be either an input or an output, while port 7 can be programmed to be either all inputs or all outputs. Alternately, these pins may be used to detect the falling edge of inputs KR0-KR3 (port 6) and KR4-KR7 (port 7). A reset signal causes these ports to default to the input mode.

$P8_0$-$P8_3$, $P9_0$-$P9_3$

Ports 8 and 9 are identical 4-bit I/O ports. Outputs are latched. A reset signal causes these ports to default to the input mode.

$P10_0/\overline{MAR}$, $P10_1/\overline{MAI}$, $P10_2/\overline{MAZ}$, $P10_3/\overline{MAT}$

These pins are used for I/O Port 10. Outputs are N-channel open drain which can withstand up to 10 volts. $P10_0$-$P10_2$ can also be used as the $\overline{MAR}$, $\overline{MAI}$, and $\overline{MAZ}$ outputs from the multifunction timer/event counter's A/D control logic. $P10_3$ can be used as the input $\overline{MAT}$ to the multifunction timer/event counter's A/D control logic. A reset signal causes this port to default to the input mode.

$P11_0$/AN0, $P11_1$/AN1, $P11_2$/AN2, $P11_3$/AN3

These pins are used for I/O Port 11, or can alternately be used as A/D converter inputs AN0-AN3. A reset signal causes this port to default to the input mode.

AN4-AN7

A/D converter inputs AN4-AN7.

AV_{DD}

A/D converter positive power supply.

AV_{SS}

A/D converter analog ground.

AV_{REF+}, AV_{REF-}

A/D converter positive and negative reference voltages.

IC/V_{PP}

This pin should be connected to V_{DD} when using the μPD75048. For the μPD75P056, this pin is used as the programming voltage input during the EPROM write/verify cycles. When the device is not being programmed, this pin should be connected to V_{DD}.

X1, X2

These pins are the main system clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

XT1, XT2

These pins are the subsystem clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

$\overline{RESET}$

This is the reset input, and it is active low.

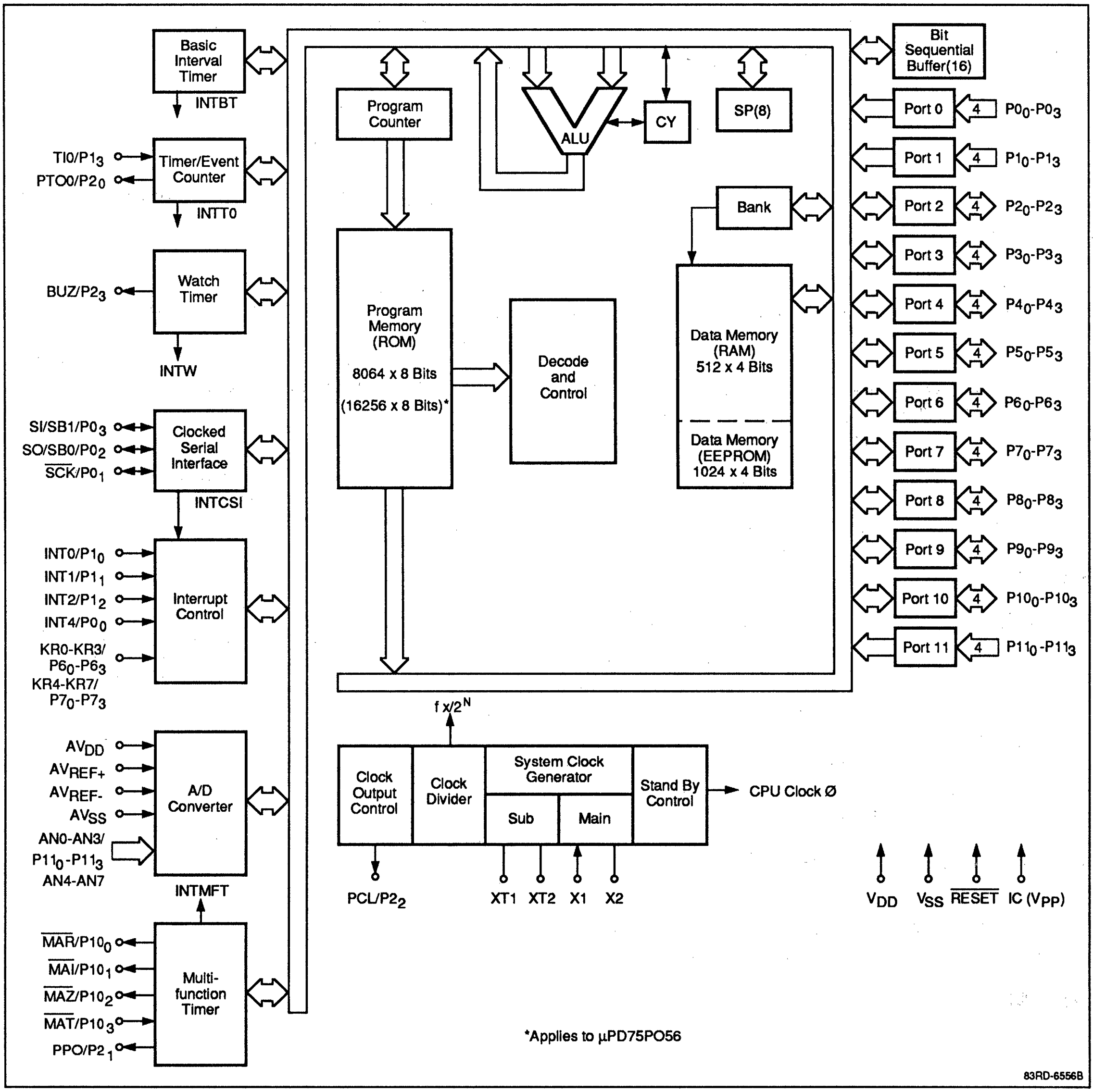
V_{DD}

The system positive power supply pin.

V_{SS}

System ground.

Block Diagram



*Applies to µPD75PO56

Specifications

ROM	8064 bytes (μPD75048) 16256 bytes (μPD75P056)
RAM	512 x 4 bits
EEPROM	1024 x 4 bits
General-purpose registers	4 bits x 8 or 8 bits x 4
Instruction cycle	0.95 μs/1.91 μs/15.3 μs (with main system clock operating at 4.19 MHz)
	122 μs (with subsystem clock operating at 32 kHz)
I/O Ports	48 total lines. There are 12 N-channel open-drain I/O ports, each tolerating as much as 10 volts. (Pullup resistor mask-option is available in the μPD75048 only). The remaining 36 lines are standard CMOS, including 12 input ports and 24 I/O ports. Of these, 27 have software-selectable pullup resistors, and four have software-selectable pulldown resistors.
A/D converter	8-bit x 8-channel
	Low-voltage operation possible (V_{DD} = 2.7 to 6.0 V)
Timer/Counter	Three. These include an 8-bit timer/event counter, an 8-bit basic interval timer, and a clock timer.

Multifunction timer	This can be used as an 8-bit timer/event counter, PWM output, 16-bit free-running timer, or 16-bit counter for an integrating A/D converter.
Serial interface	NEC standard serial bus interface (SBI)
	Clock serial interface
External interrupts	Three vector interrupts, one test input.
Internal interrupts	Six vector interrupts, one test input.
Bit sequential buffer	16-bit, on-chip
Clock output (PCL)	CPU clock φ: 524 kHz, 262 kHz, 65.6 kHz (with main system clock operating at 4.19 MHz)
Buzzer output (BUZ)	2 kHz, 4 kHz, 32 kHz (with subsystem clock operating at 32.768 kHz)
Package	64-pin plastic SDIP (750 mil)
	64-pin plastic QFP (14 x 14 mm)
Operating voltage	V_{DD} = 2.7 to 6.0 V EEPROM target specification V_{DD} = 2.7 to 6.0 V

4

NEC
NEC Electronics Inc.

μPD751xx/75P1xx
High-End 4-Bit Microcomputers

4

Description

The μPD751xx/P1xx is a family of high-performance single-chip CMOS microcomputers containing CPU, ROM, RAM, I/O ports, comparator, interval timer, two timer/counters, vectored interrupts, and a serial interface.

The instruction set allows the user to manipulate RAM data and I/O ports in 1-, 4-, and 8-bit units. The devices are ideally suited for controlling VCRs, telephones, and meters.

Both EPROM and OTP versions are available. See ordering information.

Features

□ 136 instructions
— Bit manipulation
— 4-bit and 8-bit transfer, arithmetic, logical, comparison, and increment/decrement instructions
— 1-byte relative branch
— GETI instruction, to convert one 2-byte, one 3-byte, or two 1-byte instructions into a single 1-byte instruction

□ Fast execution time
(Main system clock @ 4.19 MHz)
— High-speed cycle: 0.95 μs
— Lower-voltage cycles: 1.91 and 15.3 μs

□ Program ROM
— μPD75104/104A: 4096 bytes
— μPD75106: 6016 bytes
— μPD75108/108A/P108: 8064 bytes
— μPD75112: 12160 bytes
— μPD75116/P116: 16256 bytes

□ Data memory (RAM)
— μPD75104/104A/106: 320 x 4 bits
— Others: 512 x 4 bits
— Allows operation on 1, 4, or 8 bits

□ Bit sequential buffer
— 16-bit, bit manipulation memory

□ Four banks of eight 4-bit registers

□ Accumulators
— 1-bit (CY)
— 4-bit (A)
— 8-bit (XA)

□ 58 I/O lines
— All outputs directly drive LEDs (I_{sink} = 15 mA rms)
— 12 N-channel open-drain, can withstand 12 V
— 44 I/O lines
— 14 input-only lines

□ 4-input programmable threshold comparator

□ Three timers
— One 8-bit basic interval timer
— Two 8-bit timer/event counters

□ 8-bit serial interface
— Data transfer can be full duplex or receive only, and can be MSB or LSB first

□ Vectored interrupts
— Two-level nesting
— Three external interrupts
— Four internal interrupts
— Two inputs which generate an interrupt request

□ Standby modes
— HALT mode: stops CPU only
— STOP mode: stops main system clock

□ Power-on-reset and power-on flag
(always provided with μPD75P108, never on μPD75P116, and available on the others as a mask option)

□ Mask option port pull-up resistors
(not available on μPD75P108/P116)

□ Operates with oscillator or ceramic resonator

□ CMOS operation, with V_{DD} from 2.7 to 6.0 V

□ Low operating current (@5 V and 4.19 MHz)
— Normal operation: 3.0 mA typical
— HALT mode: 0.5 mA typical
— STOP mode: 0.1 μA typical

□ Programmable versions
— OTP & EPROM: μPD75P108
— OTP: μPD75P116
— OTP, low voltage: μPD75P108B (Note)

Note: Low voltage target spec of 2.7 to 6.0 V operation. Contact your local NEC Sales Office for latest information; none of the electrical specifications in this data sheet directly apply to this part.

Ordering Information

Part Number	Package Type	ROM
μPD75104CW-xxx	64-pin plastic SDIP (750 mil)	Mask ROM
μPD75104G-xxx-1B	64-pin plastic QFP (resin thickness = 2.05 mm; pitch = 1.0 mm)	Mask ROM
μPD75104GF-xxx-3BE	64-pin plastic QFP (resin thickness = 2.7 mm; pitch = 1.0 mm)	Mask ROM
μPD75104AGC-xxx-AB8	64-pin plastic QFP (resin thickness = 2.55 mm; pitch = 0.8 mm)	Mask ROM
μPD75106CW-xxx	64-pin plastic SDIP (750 mil)	Mask ROM
μPD75106G-xxx-1B	64-pin plastic QFP (resin thickness = 2.05 mm; pitch = 1.0 mm)	Mask ROM
μPD75106GF-xxx-3BE	64-pin plastic QFP (resin thickness = 2.7 mm; pitch = 1.0 mm)	Mask ROM
μPD75108CW-xxx	64-pin plastic SDIP (750 mil)	Mask ROM
μPD75108G-xxx-1B	64-pin plastic QFP (resin thickness = 2.05 mm; pitch = 1.0 mm)	Mask ROM
μPD75108GF-xxx-3BE	64-pin plastic QFP (resin thickness = 2.7 mm; pitch = 1.0 mm)	Mask ROM
μPD75108AG-xxx-22	64-pin plastic QFP (resin thickness = 1.5 mm; pitch = 0.8 mm)	Mask ROM
μPD75108AGC-xxx-AB8	64-pin plastic QFP (resin thickness = 2.55 mm; pitch = 0.8 mm)	Mask ROM
μPD75P108CW	64-pin plastic SDIP (750 mil)	OTP
μPD75P108DW	64-pin shrink CERDIP (w/ 350-mil window)	EPROM
μPD75P108G-1B	64-pin plastic QFP (resin thickness = 2.05 mm; pitch = 1.0 mm)	OTP
μPD75P108BCW (Note 2)	64-pin plastic SDIP	Low voltage OTP
μPD75P108BGF-3BE (Note 2)	64-pin plastic QFP	Low voltage OTP
μPD75112CW-xxx	64-pin plastic SDIP (750 mil)	Mask ROM
μPD75112GF-xxx-3BE	64-pin plastic QFP (resin thickness = 2.7 mm; pitch = 1.0 mm)	Mask ROM
μPD75116CW-xxx	64-pin plastic SDIP (750 mil)	Mask ROM
μPD75116GF-xxx-3BE	64-pin plastic QFP (resin thickness = 2.7 mm; pitch = 1.0 mm)	Mask ROM
μPD75P116CW-xxx	64-pin plastic SDIP (750 mil)	OTP
μPD75P116GF-xxx-3BE	64-pin plastic QFP (resin thickness = 2.7 mm; pitch = 1.0 mm)	OTP

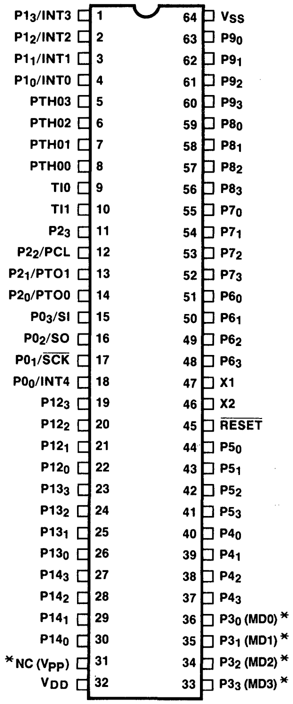
Notes:

(1) xxx indicates ROM code suffix.

(2) Contact your local NEC sales office for latest information.

Pin Configurations

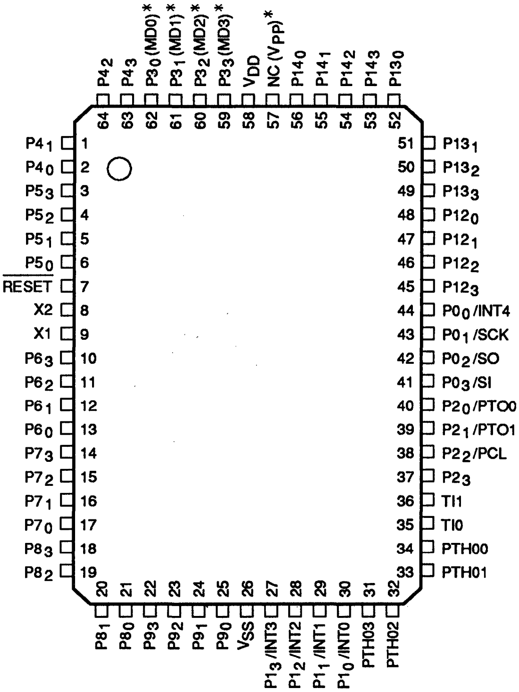
64-Pin Plastic SDIP and 64-Pin Ceramic SDIP w/Window



* V_{PP} and MD0-MD3 are for programming the μPD75P108/75P116

83-007156A

64-Pin Plastic QFP (All Parts Except μPD75104A/108A)

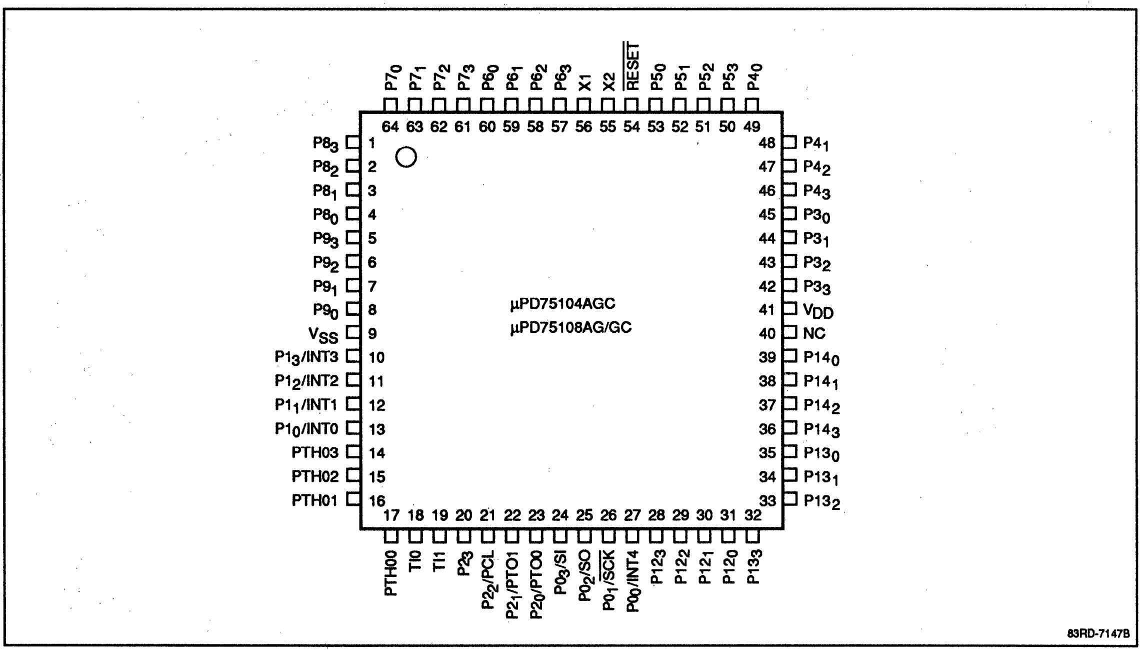


*Vpp and MD0-MD3 are for programming the μPD75P108/75P116

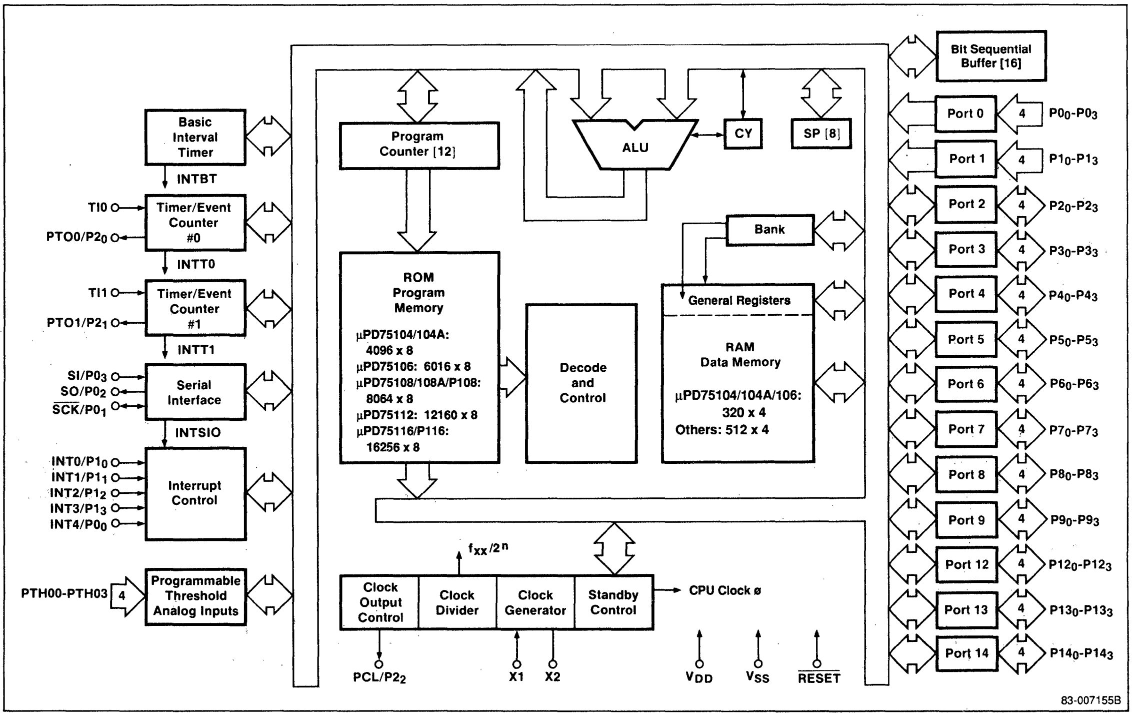
83YL-7157A

Pin Configurations (cont)

64-Pin Plastic QFP (μPD75104A/108A only)



Block Diagram



Pin Identification

Symbol	Function
$P0_0$/INT4	Port 0 input; interrupt 4
$P0_1$/SCK	Port 0 input; serial clock
$P0_2$/SO	Port 0 input; serial out
$P0_3$/SI	Port 0 input; serial in
$P1_0$/INT0	Port 1 input; interrupt 0
$P1_1$/INT1	Port 1 input; interrupt 1
$P1_2$/INT2	Port 1 input; interrupt 2
$P1_3$/INT3	Port 1 input; interrupt 3
$P2_0$/PTO0	Port 2 I/O; timer/event counter 0
$P2_1$/PTO1	Port 2 I/O; timer/event counter 1
$P2_2$/PCL	Port 2 I/O; clock output
$P2_3$	Port 2 I/O
$P3_0$/MD0	Port 3 I/O; programming mode select 0 (μPD75P108/P116)
$P3_1$/MD1	Port 3 I/O; programming mode select 1 (μPD75P108/P116)
$P3_2$/MD2	Port 3 I/O; programming mode select 2 (μPD75P108/P116)
$P3_3$/MD3	Port 3 I/O; programming mode select 3 (μPD75P108/ P116)
$P4_0$-$P4_3$	Port 4 I/O
$P5_0$-$P5_3$	Port 5 I/O
$P6_0$-$P6_3$	Port 6 I/O
$P7_0$-$P7_3$	Port 7 I/O
$P8_0$-$P8_3$	Port 8 I/O
$P9_0$-$P9_3$	Port 9 I/O
$P12_0$-$P12_3$	Port 12 I/O
$P13_0$-$P13_3$	Port 13 I/O
$P14_0$-$P14_3$	Port 14 I/O
PTH00-PTH03	4-bit programmable threshold comparator analog input port
$\overline{\text{RESET}}$	Reset input
TI0/TI1	Event timer/counter external input
V_{DD}	Positive power supply
V_{SS}	Ground
X1, X2	Main clock inputs
NC/V_{PP}	No connection; programming pin for μPD75P108/P116

PIN FUNCTIONS

$P0_0$/INT4, $P0_1$/$\overline{\text{SCK}}$, $P0_2$/SO, $P0_3$/SI

These pins can be used as the 4-bit input port 0. $P0_0$ can be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. $P0_1$-$P0_3$ may also be used for the serial interface; SI is the serial input, SO is the serial output, and $\overline{\text{SCK}}$ is the serial clock. Reset causes these pins to default to the Port 0 input mode.

$P1_0$/INT0, $P1_1$/INT1, $P1_2$/INT2, $P1_3$/INT3

These pins can be used as 4-bit input port 1. They can also be used, respectively, for edge-triggered interrupts INT0, INT1, INT2, and INT3. INT0 and INT1 are triggered by rising or falling edges, while INT2 and INT3 respond to rising edges only and generate an interrupt request but not an interrupt. Reset causes these pins to default to the Port 1 input mode. Individual pull-up resistors can be provided by mask option in the μPD75104A/108A.

$P2_0$/PTO0, $P2_1$/PTO1, $P2_2$/PCL, $P2_3$

These pins can be used as 4-bit I/O port 2. This port has latched outputs, and can directly drive LEDs. PTO0 and PTO1 are the timer/event counter output pins; PCL is the clock output pin. Reset causes these pins to default to the Port 2 input mode.

$P3_0$/MD0, $P3_1$/MD1, $P3_2$/MD2, $P3_3$/MD3

These pins are used for I/O Port 3. Each bit in this port can be independently programmed to be either an input or an output. This port has latched outputs, and can directly drive LEDs. $P3_0$-$P3_3$ are used as the programming mode select pins for the μPD75P108/P116 during EPROM/OTP programming and verification. A reset signal causes this port to default to the input mode.

$P4_0$-$P4_3$, $P5_0$-$P5_3$

Port 4 and Port 5 are identical 4-bit I/O ports which can be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. A reset signal causes these ports to default to the input mode. Individual pull-up resistors are available as a mask option in the μPD75104A/108A.

$P6_0$-$P6_3$, $P7_0$-$P7_3$

Port 6 and Port 7 are 4-bit I/O ports; port 6 is I/O bit programmable. These ports may be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. A reset signal causes these ports to default to the input mode. Individual pull-up resistors are available as a mask option in the μPD75104A/108A

$P8_0$-$P9_3$, $P9_0$-$P9_3$

Port 8 and Port 9 are identical 4-bit I/O ports which can be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. A reset signal causes these ports to default to the input mode. Individual pull-up resistors are available as a mask option in the μPD75104A/108A

$P12_0$-$P12_3$, $P13_0$-$P13_3$

Port 12 and Port 13 are identical 4-bit I/O ports which can be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. Outputs are N-channel open drain, and can withstand up to 12 volts; pull-up resistor mask options are available for these ports. A reset signal causes these ports to default to the input mode.

$P14_0$-$P14_3$

Port 14 is a 4-bit I/O port. Latched outputs will directly drive LEDs. Outputs are N-channel open drain, and can withstand up to 12 volts; pull-up resistor mask options are available for this port. A reset signal causes the port to default to the input mode.

PTH00-PTH03

4-channel comparator with 4-bit resolution and on-chip resistor ladder.

TI0, TI1

External event input for the timer/event counters. Each pin can also act as an edge-triggered vectored interrupt and a 1-bit input port.

NC/V_{PP}

This pin may be left unconnected when using the μPD751xx. For the μPD75P108/P116, this pin is used as the programming voltage input during the EPROM write/verify cycles. When the devices are not being programmed, this pin should be connected to V_{DD}. Pin must be connected to V_{DD} if the same circuit board is used for both programmable and nonprogrammable devices.

X1, X2

These pins are the system clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

$\overline{\text{RESET}}$

This is the reset input, and it is active low.

V_{DD}

The system positive power supply pin.

V_{SS}

System ground.

Product Comparison

Item	μPD75104/104A	μPD75106	μPD75108/108A	μPD75P108	μPD75112	μPD75116	μPD75P116
Program memory	Mask ROM 000H–FFFH 4096 x 8 bits	Mask ROM 000H–177FH 6016 x 8 bits	Mask ROM 000H–1F7FH 8064 x 8 bits	EPROM/OTP 000H–1FFFH 8192 x 8 bits	Mask ROM 000H–2F7FH 12160 x 8 bits	Mask ROM 000H–3F7FH 16256 x 8 bits	OTP 000H–3FFFH 16384 x 8 bits
Data memory	320 x 4 bits Bank 0: 256 x 4 Bank 1: 64 x 4	320 x 4 bits Bank 0: 256 x 4 Bank 1: 64 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4	512 x 4 bits Bank 0: 256 x 4 Bank 1: 256 x 4
Instruction set	The BR !addr instruction is not provided in the μPD75104/104A.						
Port lines	CMOS I/O lines: 32 12 open-drain outputs with 12 V breakdown. These outputs can have pull-up resistors as a mask option, except for programmable parts. (Note 1) Lines which directly drive LEDs: 44 Total number of lines: 52 (44 I/O and 8 input-only)						

Product Comparison

Item	μPD75104/104A	μPD75106	μPD75108/108A	μPD75P108	μPD75112	μPD75116	μPD75P116
Power-on-reset circuit and power-on flag	Mask option	Mask option	Mask option	Internally provided	Mask option	Mask option	Not included
Operating voltage range	2.7 to 6.0 V	2.7 to 6.0 V	2.7 to 6.0 V	5 V ± 10%	2.7 to 6.0 V	2.7 to 6.0 V	5 V ± 10%
Package	See ordering information for a complete list of packages						

Notes:

(1) The μPD75104/104A have 24 additional I/O port lines and 4 more input-only lines with mask option programmable pull-up resistors.

ADDRESS SPACES AND MEMORY MAPS

The 75X architecture has two separate address spaces, one for program memory (ROM), and another for data memory (RAM).

Program Memory (ROM)

The ROM is addressed by the program counter. The size of the program counter is 12, 13, or 14 bits; its size depends on which member of the family is being used, as does the amount of ROM present. The ROM contains program object code, interrupt vector table, a GETI instruction reference table, and table data. Table data can be obtained using the table reference instruction, MOVT.

Figure 1 shows the addressing range which can be made using a branch instruction or subroutine call instruction. In addition, the BR PCDE and BR PCXA instructions can be used for a branch where only the low 8 bits of the PC are changed. The program memory addresses are,

μPD75104/104A: 000H to FFFH
μPD75106: 0000H to 177FH
μPD75108/108A: 0000H to 1F7FH
μPD75P108: 0000H to 1FFFH
μPD75112: 0000H to 2F7FH
μPD75116: 0000H to 3F7FH
μPD75P116: 0000H to 3FFFH

All locations of ROM except 000H and 0001H can be used as program memory. However, if interrupts or GETI instructions are used, the locations corresponding to those functions cannot be used. Addresses are normally reserved as follows:

0000H to 0001H:	This address area contains the program start address when a RESET is applied, and is also used for setting the values of RBE and MBE. Program execution can be started from any address after a RESET.
002H to 000BH:	This area is used for interrupt vector addresses and for setting the value of RBE and MBE. Interrupts can start from any location except where noted.
0020H to 007FH:	This is the table area for GETI instructions. The GETI instruction is used to access 1, 2 or 3-byte instructions using one byte of program memory. This is useful in compacting code.

Program Counter (PC)

This is a 12/13/14-bit binary counter that contains the address of the current program memory location. The μPD75104/104A contain a 12-bit PC, the μPD75106/108/108A/P108 have a 13-bit PC, and the μPD75112/116/P116 have a 14-bit PC.

When an instruction is executed, the PC is automatically incremented by the number of bytes of the current instruction. When a branch instruction (BR, BRCB) is executed, the contents of the immediate data or register pair indicating the new address are loaded into some or all the bits of the PC. When a subroutine call instruction (CALL, CALLF) is executed or an interrupt is generated, the PC is incremented to point to the next instruction, and this information is saved on the stack. During an interrupt, the program status word (PSW) is also automatically saved on the stack. The address to be jumped to by the CALL or interrupt is then loaded into the PC.

When a return instruction (RET, RETS, or RETI) is executed, the contents of the stack are restored to the PC.

Data Memory (RAM)

The data memory contains three memory banks, 0, 1, and 15. The RAM memory map is shown in figure 2. The memory consists of general purpose static RAM, general purpose registers, and peripheral control registers. Memory banks are accessed by using the MBE (memory bank enable) bit and by programming the BS (bank select) register. If MBE = 0, the lower 128 nibbles of memory bank 0 and the upper 128 nibbles of memory bank 15 are accessed. If MBE = 1, the upper four bits in the BS register will specify the memory bank. The values are 0H for memory bank 0, 1H for memory bank 1, and FH for memory bank 15. Memory bank 0 contains 256 nibbles, while memory bank 1 contains either 64 or 256 nibbles depending on which member of the μPD751XX/P1XX family is being used. Although the memory is organized in nibbles, the 75X architecture allows the data to be manipulated in bytes, nibbles and individual bits.

The data memory is used for storing processed data, general purpose registers, and as a stack for subroutine or interrupt service. Because of its static nature, the RAM will retain its data when the chip is in the STOP mode, provided V_{DD} is at least 2 volts.

The on-chip peripheral control registers and ports reside in the upper 128 nibbles of bank 15. Bank 15 addresses which are not assigned to a register are not available as random memory except for the 16-bit sequential buffer. Also, the lower 128 nibbles of bank 15 do not contain RAM.

There are four general-purpose register banks in RAM Bank 0, beginning at address 00H. Each bank contains eight 4-bit registers, (B, C, D, E, H, L, X, A), which may be used together to form four 8-bit registers. Register bank selection is accomplished by using the two low-order bits of the BS register and the RBE (register bank enable) bit. A register bank is selected by setting RBE to 1 and programming BS to be 0H-3H for register banks 0-3, respectively. If RBE = 0, the chip defaults to bank 0. Registers which are not used for any other purpose may be used as general purpose RAM.

Each register can be used either in a 4-bit configuration or in a 8-bit configuration when paired with one of the others (BC, DE, HL, XA). There is also a "DL" pair available. DL and pairs DE and HL can be used as data pointers. For 8-bit manipulation, besides BC, DE, HL, and XA, register pairs BC', DE', HL', and XA' are provided. If memory bank 0 is selected and BC' is referenced, BC' is register BC in memory bank 1. If bank 1 is selected and BC' is referenced, BC' is register BC in bank 0.The same concept is true for register banks 2 and 3

Figure 1. Program Memory Map

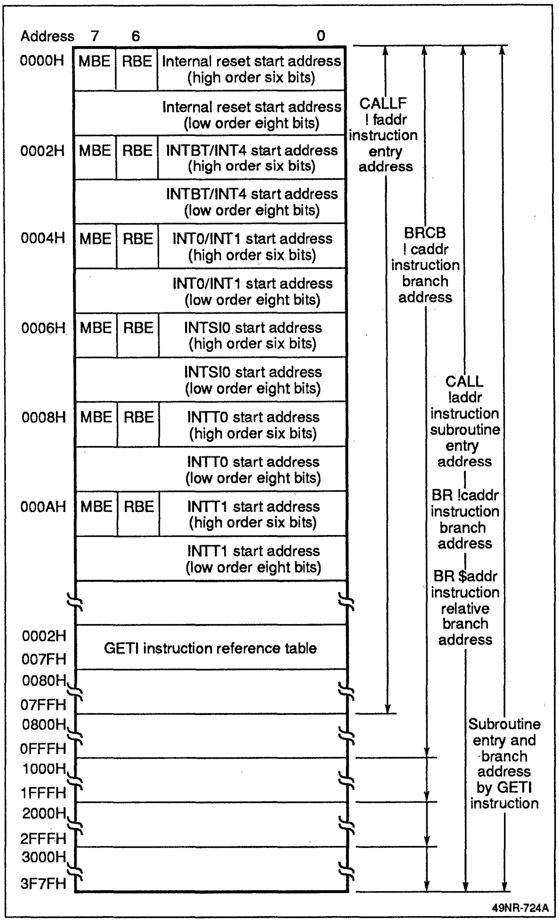

Address 7 6 0
0000H
MBE
RBE
Internal reset start address (high order six bits)
Internal reset start address (low order eight bits)
0002H
MBE
RBE
INTBT/INT4 start address (high order six bits)
INTBT/INT4 start address (low order eight bits)
0004H
MBE
RBE
INT0/INT1 start address (high order six bits)
INT0/INT1 start address (low order eight bits)
0006H
MBE
RBE
INTSI0 start address (high order six bits)
INTSI0 start address (low order eight bits)
0008H
MBE
RBE
INTT0 start address (high order six bits)
INTT0 start address (low order eight bits)
000AH
MBE
RBE
INTT1 start address (high order six bits)
INTT1 start address (low order eight bits)
0002H
007FH
GETI instruction reference table
0080H
07FFH
0800H
0FFFH
1000H
1FFFH
2000H
2FFFH
3000H
3F7FH
CALLF ! faddr instruction entry address
BRCB ! caddr instruction branch address
CALL !addr instruction subroutine entry address
BR !caddr instruction branch address
BR $addr instruction relative branch address
Subroutine entry and branch address by GETI instruction
49NR-724A

Figure 2. Data Memory Map (μPD75104/104A/106)

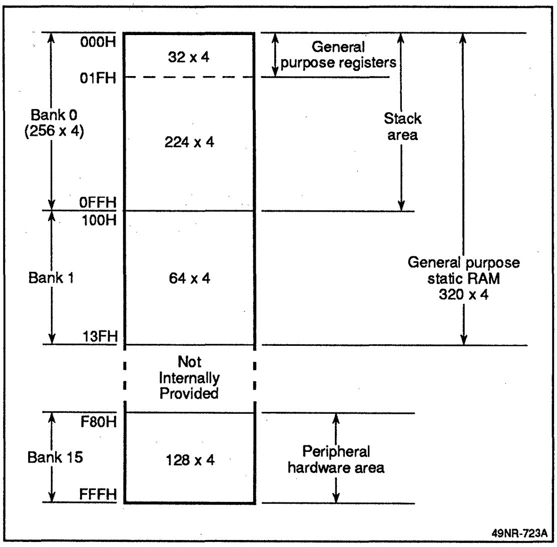


Figure 2a. Data Memory Map (μPD75108 to μPD75116)

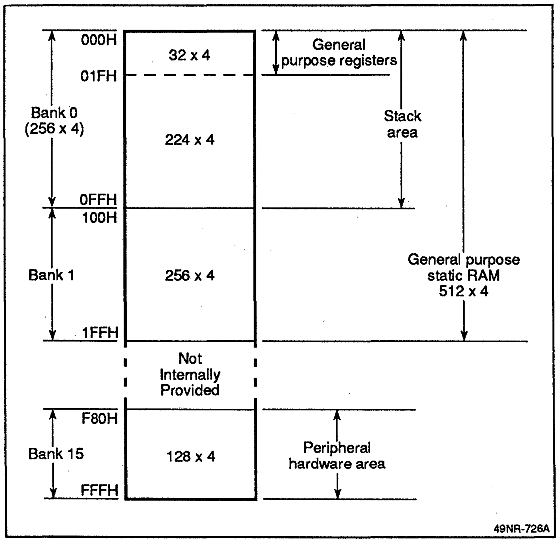
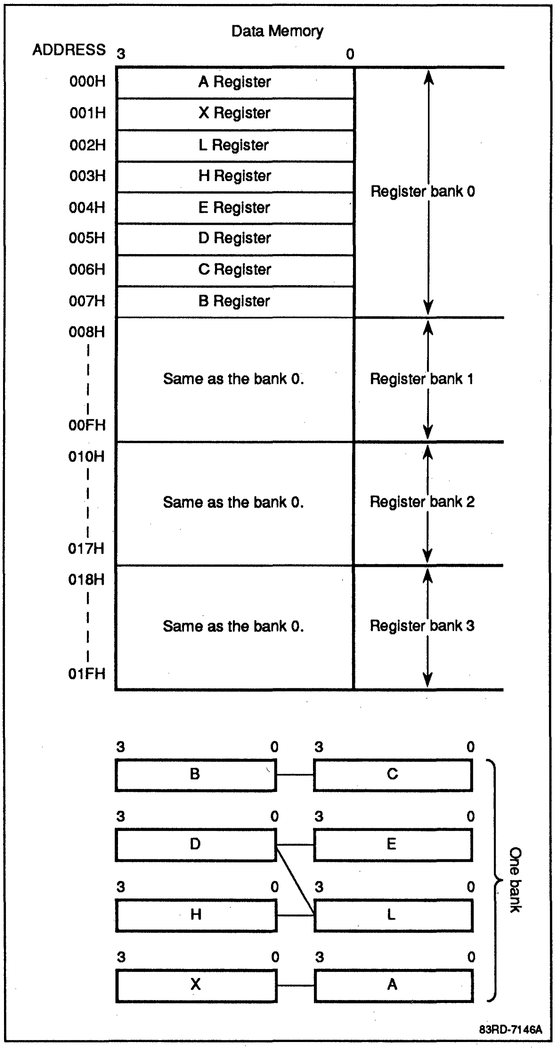


Figure 3. General Purpose Register Configurations



Addressing Modes

The μPD751xx/P1xx is able to address data memory and ports as individual bits, nibbles, or bytes. The addressing modes are as follows:

1-bit direct data memory
4-bit direct data memory
4-bit register indirect (@rpa)
8-bit direct data memory
8-bit register indirect (@HL)

See table 1 for data memory addressing and table 2 for peripheral control register addressing.

Table 1. Data Memory Addressing Modes

Addressing Mode	Representation Format	How the Address Is Created
1-bit direct addressing	mem.bit	If MBE = 0, the memory bank is Bank 0 for addresses 00H-7FH, and Bank 15 for addresses 80H-FFH. If MBE = 1, the memory bank is selected by the four bits of the MBS. The bit to be manipulated is specified in mem.bit
4-bit direct addressing	mem	If MBE = 0, the memory bank is Bank 0 for addresses 00H-7FH, and Bank 15 for addresses 80H-FFH. If MBE = 1, the memory bank is selected by the four bits of the MBS. The bit to be manipulated is specified in mem.
8-bit direct addressing	mem (must be an even address)	If MBE = 0, the memory bank is Bank 0 for addresses 00H-7FH, and Bank 15 for addresses 80H-FFH. If MBE = 1, the memory bank is selected by the four bits of the MBS. The bit to be manipulated is specified in mem.
4-bit register indirect addressing	@HL, @HL+, @HL–	The memory bank is selected by the four bits of the MBS, and the location within the memory bank is contained in register HL. @HL+ : After addressing the L register automatically increments. @HL – : After addressing, the L register automatically decrements.
	@DE	The memory bank is always Bank 0, and the location within the memory bank is contained in register DE
	@DL	The memory bank is always Bank 0, and the location within the memory bank is contained in register DL
8-bit register indirect addressing	@HL (must be an even address)	The memory bank is selected by the four bits of the MBS, and the location within the memory bank is contained in register HL.

Bit manipulation addressing	fmem.bit	The memory bank is Bank 15, and the location is fmem, where
		fmem = FB0H- FBFH for interrupts
		fmem = FF0H-FFFH I/O ports
		The actual bit is specified in fmem.bit

	pmem.@L (where pmem= FC0H to FFFH)	The memory location is independent of MBE and MBS. The upper 10 address bits of the location are contained in the ten high order bits of pmem and the two lower address bits are contained in the two upper bits of register L. The bit to be manipulated is specified by the two LSBs of register L.
	@H + mem.bit	The memory bank is selected by the four bits of the MBS, and the location is determined by the following: The four upper bits are the contents of register H The four lower bits are mem. The actual bit is specified in mem.bit.
Stack addressing		The memory bank is always Bank 0, and the location is indicated by the stack pointer (SP)

MBE: memory bank enable bit
MB: memory bank
MBS: memory bank select register
mem: a location within a memory bank
mem.bit: a bit at a specified memory location.
fmem and pmem are specialized cases of mem.

Table 2. Addressing Modes During Peripheral Hardware Operation

Manipulation	Addressing Mode	Applicable Hardware
1-bit	With MBE = 0 (or MBE = 1 and MBS = 15) direct addressing (address in mem.bit)	All hardware where bit manipulation can be performed
	Direct addressing regardless of how MBE and MBS are set. (address in fmem.bit)	IST0, IST1, MBE, RBE IExxx, IRQxxx, PORTn (n=0 to 3)
	Indirect addressing regardless of how MBE and MBS are set. (address in pmem. @L)	BSBn.x PORTn
4-bit	With MBE = 0 (or MBE = 1 and MBS = 15) direct addressing (address in mem.bit)	All hardware where 4-bit manipulation can be performed
	With MBE = 1 and MBS = 15, register indirect addressing (address in @HL)	
8-bit	With MBE=0 (or MBE = 1 and MBS = 15) direct addressing (address in mem); mem must be an even address	All hardware where 8-bit manipulation can be performed
	With MBE = 1 and MBS = 15, register indirect addressing (address in @HL); L register must contain an even number	

INSTRUCTIONS

The μPD751xx/P1xx provides a powerful set of 136 instructions.

Instruction Timing

The minimum instruction execution time is 0.95 μs with a 4.19 MHz clock. The PCC register can be used to program the CPU's minimum instruction cycle time to 0.95, 1.91, or 15.3 μs; all three speeds presuppose a 4.19 MHz crystal. Reducing the CPU clock speed will reduce the microprocessor's power consumption.

Instruction Set

The instruction set contains the following features:

- Versatile bit manipulation instructions
- Efficient 4-bit manipulation instructions
- 8-bit instructions
- GETI instruction to reduce program size
- Vertically stored instructions and base correction instructions
- Table reference instructions
- 1-byte relative branch instructions

The instruction set is unusually powerful for a 4-bit microcomputer. It consists of the full 75X instruction set. It contains instructions that operate on 1-bit, 4-bit, and 8-bit data. It contains 8-bit instructions generically equivalent to virtually every 4-bit instruction type. Specifically, the instruction set contains the following 8-bit instruction types:

- Arithmetic: ADD W/CARRY, ADD W/SKIP, SUB W/BORROW, SUB W/SKIP
- Logical: AND, OR, XOR
- Comparison: SKE (skip if equal)
- Transfer: MOV, MOVT, XCH, IN, OUT, PUSH, POP, BR, CALL
- Manipulation: INC W/SKIP, DEC W/SKIP

In addition, some of the 4-bit ports may be paired together to function as one 8-bit port. The combination of 8-bit ports and 8-bit instructions allows IN and OUT instructions to move full bytes of data at a time.

Organization. Tables 3 and 4 define the instruction set symbols and operand formats, found in the instruction set.

Clock Cycles. One machine cycle equals one CPU Clock Cycle ϕ. The PCC selects one of four available CPU cycle speeds.

Skip Cycles. S equals the number of extra machine cycles required for skip operation when executing a skip instruction:

- S = 0; no skip
- S = 1; one- or two-byte instruction or GETI instruction is skipped
- S = 2; three-byte instruction is skipped (BR !addr, CALL !addr)

Table 3. Instruction Set Symbols

The devices use the following symbol definitions:

Symbol	Definition
A	A register; 4-bit accumulator
B	B register; 4-bit register
C	C register; 4-bit register
D	D register; 4-bit register
E	E register; 4-bit register
H	H register; 4-bit register
L	L register; 4-bit register
X	X register; 4-bit register
XA	XA register pair; 8-bit accumulator
BC	BC register pair; 8-bit register
DE	DE register pair; 8-bit register
DL	DL register pair; 8-bit register
HL	HL register pair; 8-bit register
XA′	XA′ register pair; 8-bit register
BC′	BC′ register pair; 8-bit register
DE′	DE′ register pair; 8-bit register
HL′	HL′ register pair; 8-bit register
PC	Program counter
SP	Stack pointer
CY	Carry flag; bit accumulator
PSW	Program status word
MBE	Memory bank enable flag
RBE	Register bank enable flag
PORTn	Port n (n = 0-9, 12-14)
IME	Interrupt master enable
IPS	Interrupt priority selection register
IExxx	Interrupt enable flag
RBS	Register bank selection register
MBS	Memory bank selection register
PCC	Clock processor control register
.	Separation between address and bit
(xx)	The contents addressed by xx
xxH	Hexadecimal data

Operation Representation Format and Description Method

An operand is entered in the operand field of each instruction according to the format of the instruction (see assembler specifications). When two or more en tries are indicated in the description method, one should be selected. Capital letters and symbols must be entered exactly as shown. For immediate data, a proper numeric value or label should be entered as shown in table 4.

Table 4. Operand Formats

Symbol	Description
reg reg1	X, A, B, C, D, E, H, L X, B, C, D, E, H, L
rp rp1 rp2 rp′ rp′1	XA, BC, DE, HL BC, DE, HL BC, DE XA, BC, DE, HL, XA′, BC′ DE′, HL′ BC, DE, HL, XA′, BC′ DE′, HL′
rpa rpa1	HL, HL+, HL–, DE, DL DE, DL
n4 n8	4-bit immediate data or label 8-bit immediate data or label
mem (Note 1) bit	8-bit immediate data or label 2-bit immediate data or label
fmem pmem	FB0H-FBFH, FF0H-FFFH immediate data or label FC0H-FFFH immediate data or label
addr, caddr	μPD75104: 000H-FFFH immediate data or label
	μPD75106: 000H-177FH immediate data or label
	μPD75108: 000H-1F7FH immediate data or label μPD75P108: 000H-1FFFH immediate data or label
	μPD75112: 000H-2F7FH immediate data or label
	μPD75116: 000H-3F7F immediate data or label μPD75P116: 000H-3FFFH immediate data or label.
faddr	11-bit immediate data or label
taddr	20H-7EH immediate data (where bit 0 = 0) or label
PORTn IExxx RBn MBn	Port 0-Port 9, Port 12-Port 14 IEBT, IESIO, IET0, IET1, IE0-IE4 RB0-RB3 MB0, MB1, MB15

Notes:

(1) Memory address must be an even number in 8-bit processing.

String Instructions

The μPD751xx/P1xx family has the following two types of string effect instructions:

(1) MOV A, #n4 or MOV XA, #n8
(2) MOV HL, #n8

String effect means to place the same type instructions in consecutive addresses. For example;

A0: MOV A, #0
A1: MOV A, #1
XA7: MOV XA, #07

If the first execution address is A0, the two subsequent instructions are treated as NOP instructions during program execution; if the first execution address is A1, the instruction that follows is treated as an NOP instruction during program execution. This means that only the first string instruction is valid, with the following string instructions being treated as NOP instructions during program execution.

The string instructions increase efficiency when setting constants into an accumulator (A register or XA register-pair) or into a data pointer (HL register pair).

Instruction Set

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Transfer					
MOV	A, #n4	1	1	A ← n4	String A
	reg1, #n4	2	2	reg1 ← n4	
	XA, #n8	2	2	XA ← n8	String A
	HL, #n8	2	2	HL ← n8	String B
	rp2, #n8	2	2	rp2 ← n8	
	A, @HL	1	1	A ← (HL)	
	A, @HL+	1	2+S	A ← (HL), then L←L+1	L = 0
	A, @HL–	1	2+S	A ← (HL), then L←L–1	L = FH
	A, @rpa1	1	1	A ← (rpa1)	
	XA, @HL	2	2	XA ← (HL)	
	@HL, A	1	1	(HL) ← A	
	@HL, XA	2	2	(HL) ← XA	
	A, mem	2	2	A ← (mem)	
	XA, mem	2	2	XA ← (mem)	
	mem, A	2	2	(mem) ← A	
	mem, XA	2	2	(mem) ← XA	
	A, reg1	2	2	A ← (reg1)	
	XA, rp'	2	2	XA ← rp'	
	reg1, A	2	2	reg1 ← A	
	rp'1, XA	2	2	rp'1 ← XA	
XCH	A, @HL	1	1	A ↔ (HL)	
	A, @HL+	1	2+S	A ↔ (HL), then L←L+1	L = 0
	A, @HL–	1	2+S	A ↔ (HL), then L←L–1	L = FH
	A, @rpa1	1	1	A ↔ (rpa1)	
	XA, @HL	2	2	XA ↔ (HL)	
	A, mem	2	2	A ↔ (mem)	
	XA, mem	2	2	XA ↔ (mem)	
	A, reg1	1	1	A ↔ (reg1)	
	XA, rp'	2	2	XA ↔ rp'	
MOVT	XA, @PCDE	1	3	XA ← $(PC_{13-8} + DE)_{ROM}$	
	XA, @PCXA	1	3	XA ← $(PC_{13-8} + XA)_{ROM}$	

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Transfer (cont)					
MOV1	CY, fmem.bit	2	2	CY ← (fmem.bit)	
	CY, pmem.@L	2	2	CY ← (pmem$_{7-2}$+ L$_{3-2}$.bit(L$_{1-0}$))	
	CY, @H+ mem.bit	2	2	CY ← (H+ mem$_{3-0}$.bit)	
	fmem.bit, CY	2	2	(fmem.bit) ← CY	
	pmem.@L, CY	2	2	(pmem$_{7-2}$+ L$_{3-2}$.bit(L$_{1-0}$)) ← CY	
	@H+ mem.bit, CY	2	2	(H+ mem$_{3-0}$.bit) ← CY	
Arithmetic					
ADDS	A, #n4	1	1+S	A ← A+ n4	Carry
	XA, #n8	2	2+S	XA ← XA+ n8	Carry
	A, @HL	1	1+S	A ← A+ (HL)	Carry
	XA, rp′	2	2+S	XA ← XA+ rp′	Carry
	rp′1, XA	2	2+S	rp′1 ← rp′1+XA	Carry
ADDC	A, @HL	1	1	A, CY ← A+ (HL)+ CY	
	XA, rp′	2	2	XA, CY ← XA+ rp′+ CY	
	rp′1, XA	2	2	rp′1, CY ← rp′1+XA+ CY	
SUBS	A, @HL	1	1+S	A ← A–(HL)	Borrow
	XA, rp′	2	2+S	XA ← XA–rp′	Borrow
	rp′1, XA	2	2+S	rp′1 ← rp′1–XA	Borrow
SUBC	A, @HL	1	1	A, CY ← A – (HL) – CY	
	XA, rp′	2	2	XA, CY ← XA–rp′–CY	
	rp′1, XA	2	2	rp′1, CY ← rp′1–XA–CY	
AND	A, #n4	2	2	A←A $\wedge$ n4	
	A, @HL	1	1	A←A $\wedge$ (HL)	
	XA, rp′	2	2	XA ← XA $\wedge$ rp′	
	rp′1, XA	2	2	rp′1 ← rp′1 $\wedge$ XA	
OR	A, #n4	2	2	A ← A $\vee$ n4	
	A, @HL	1	1	A ←A $\vee$ (HL)	
	XA, rp′	2	2	XA ← XA $\vee$ rp′	
	rp′1, XA	2	2	rp′1 ← rp′1 $\vee$ XA	
XOR	A, #n4	2	2	A ← A **XOR** n4	
	A, @HL	1	1	A ← A **XOR** (HL)	
	XA, rp′	2	2	XA ← XA **XOR** rp′	
	rp′1, XA	2	2	rp′1 ← rp′1 **XOR** XA	
Accumulator Manipulation					
RORC	A	1	1	CY ← A$_0$, A$_3$ ← CY, A$_{n-1}$ ← A$_n$	
NOT	A	2	2	A ← $\overline{A}$	

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Increment/decrement					
INCS	reg	1	1+S	reg ← reg+1	reg = 0
	rp1	1	1+S	rp1 ← rp1+1	rp1 = 00H
	@HL	2	2+S	(HL) ← (HL)+1	(HL) = 0
	mem	2	2+S	(mem) ← (mem)+1	(mem) = 0
DECS	reg	1	1+S	reg ← reg–1	reg = FH
	rp′	2	2+S	rp′ ← rp′–1	rp′ = FFH
Comparison					
SKE	reg, #n4	2	2+S	skip if reg = n4	reg = n4
	@HL, #n4	2	2+S	skip if (HL) = n4	(HL) = n4
	A, @HL	1	1+S	skip if A = (HL)	A = (HL)
	XA, @HL	2	2+S	skip if XA = (HL)	XA = (HL)
	A, reg	2	2+S	skip if A = reg	A = reg
	XA, rp′	2	2+S	skip if XA = rp′	XA = rp′
Carry Flag Manipulation					
SET1	CY	1	1	CY ← 1	
CLR1	CY	1	1	CY ← 0	
SKT	CY	1	1+S	skip if CY = 1	CY = 1
NOT1	CY	1	1	CY ← $\overline{CY}$	
Memory Bit Manipulation					
SET1	mem.bit	2	2	(mem.bit) ← 1	
	fmem.bit	2	2	(fmem.bit) ← 1	
	pmem.@L	2	2	(pmem$_{7-2}$+ L$_{3-2}$.bit(L$_{1-0}$)) ← 1	
	@H+ mem.bit	2	2	(H + mem$_{3-0}$.bit) ← 1	
CLR1	mem.bit	2	2	(mem.bit) ← 0	
	fmem.bit	2	2	(fmem.bit) ← 0	
	pmem.@L	2	2	(pmem$_{7-2}$+ L$_{3-2}$.bit(L$_{1-0}$)) ← 0	
	@H+ mem.bit	2	2	(H + mem$_{3-0}$.bit) ← 0	
SKT	mem.bit	2	2+S	skip if (mem.bit) = 1	(mem.bit) = 1
	fmem.bit	2	2+S	skip if (fmem.bit) = 1	(fmem.bit) = 1
	pmem.@L	2	2+S	skip if (pmem$_{7-2}$+ L$_{3-2}$.bit(L$_{1-0}$)) = 1	(pmem.@L = 1)
	@H+ mem.bit	2	2+S	skip if (H+ mem$_{3-0}$.bit) = 1	(@H+ mem.bit) = 1
SKF	mem.bit	2	2+S	skip if (mem.bit) = 0	(mem.bit) = 0
	fmem.bit	2	2+S	skip if (fmem.bit) = 0	(fmem.bit) = 0
	pmem.@L	2	2+S	skip if (pmem$_{7-2}$+ L$_{3-2}$.bit(L$_{1-0}$)) = 0	(pmem.@L = 0)
	@H+ mem.bit	2	2+S	skip if (H+ mem$_{3-0}$.bit) = 0	(@H+ mem.bit) = 0

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Memory Bit Manipulation (cont)					
SKTCLR	fmem.bit	2	2+S	skip if (fmem.bit) = 1 and clear	(fmem.bit) = 1
	pmem.@L	2	2+S	skip if ($pmem_{7-2}$+L_{3-2}.bit(L_{1-0})) = 1 and clear	(pmem.@L = 1)
	@H+mem.bit	2	2+S	skip if (H+mem_{3-0}.bit) = 1 and clear	(@H+mem.bit) = 1
AND1	CY, fmem.bit	2	2	CY ← CY ∧ (fmem.bit)	
	CY, pmem.@L	2	2	CY ← CY ∧ ($pmem_{7-2}$+L_{3-2}.bit(L_{1-0}))	
	CY, @H+mem.bit	2	2	CY ← CY ∧ (H+mem_{3-0}.bit)	
OR1	CY, fmem.bit	2	2	CY ← CY ∨ (fmem.bit)	
	CY, pmem.@L	2	2	CY ← CY ∨ ($pmem_{7-2}$+L_{3-2}.bit(L_{1-0}))	
	CY, @H+mem.bit	2	2	CY ← CY ∨ (H+mem_{3-0}.bit)	
XOR1	CY, fmem.bit	2	2	CY ← CY **XOR** (fmem.bit)	
	CY, pmem.@L	2	2	CY ← CY **XOR** ($pmem_{7-2}$+L_{3-2}.bit(L_{1-0}))	
	CY, @H+mem.bit	2	2	CY ← CY **XOR** (H+mem_{3-0}.bit)	
Branch					
BR (Note 1)	addr	–	–	PC_{13-0} ← addr	
	!addr (Note 1)	3	3	PC_{13-0} ← addr	
	$addr	1	2	PC_{13-0} ← addr	
BRCB	!caddr	2	2	PC_{13-0} ← $PC_{13,12}$+ $caddr_{11-0}$	
BR	PCDE	2	3	PC_{13-0} ← PC_{13-8}+DE	
	PCXA	2	3	PC_{13-0} ← PC_{13-8}+XA	
Subroutine Stack Control					
CALL	!addr	3	3	(SP−4)(SP−1)(SP−2) ← PC_{11-0} (SP−3) ← (MBE, RBE, $PC_{13,12}$) PC_{13-0} ← addr, SP ← (SP−4)	
CALLF	!faddr	2	2	(SP−4)(SP−1)(SP−2) ← PC_{11-0} (SP−3) ← (MBE, RBE, $PC_{13,12}$) PC_{13-0} ← 00, faddr, SP ← (SP−4)	
RET		1	3	(MBE, RBE, $PC_{13,12}$) ← (SP+1) PC_{11-0} ← (SP)(SP+3)(SP+2) SP ← (SP+4)	
RETS		1	3+S	(MBE, RBE, $PC_{13,12}$) ← (SP+1) PC_{11-0} ← (SP)(SP+3)(SP+2) SP ← (SP+4), then skip unconditionally	Unconditional
RETI		1	3	($PC_{13,12}$) ← (SP+1) PC_{11-0} ← (SP)(SP+3)(SP+2) PSW ← (SP+4)(SP+5), SP ← (SP+6)	
PUSH	rp	1	1	(SP−1)(SP−2) ← rp, SP ← (SP−2)	
	BS	2	2	(SP−1) ← MBS, (SP−2) ← RBS, SP ← (SP−2)	
POP	rp	1	1	rp ← (SP+1)(SP), SP ← (SP+2)	
	BS	2	2	MBS ← (SP+1), RBS ← (SP), SP ← (SP+2)	

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Interrupt Control					
EI		2	2	IME ← 1	
	IExxx	2	2	IExxx ← 1	
DI		2	2	IME ← 0	
	IExxx	2	2	IExxx ← 0	
Input/Output (Note 2)					
IN	A, PORT$_n$	2	2	A ← PORT$_n$; (n = 0 to 9, 12-14)	
	XA, PORT$_n$	2	2	XA ← PORT$_{n+1}$, PORT$_n$; (n = 4, 6, 8, 12)	
OUT	PORT$_n$, A	2	2	PORT$_n$ ← A; (n = 2 to 9, 12-14)	
	PORT$_n$, XA	2	2	PORT$_{n+1}$, PORT$_n$ ← XA; (n = 4, 6, 8, 12)	
CPU Control					
HALT		2	2	Set HALT mode (PCC.2 ← 1)	
STOP		2	2	Set STOP mode (PCC.3 ← 1)	
NOP		1	1	No operation	
Special					
SEL	RB$_n$	2	2	RBS ← n; (n = 0-3)	
	MB$_n$	2	2	MBS ← n; (n = 0, 1, 15)	
GETI	taddr	1	3	When the table is specified by the TBR instruction, PC$_{11-0}$ ← (taddr)$_{3-0}$ + (taddr+1) When the table is specified by the TCALL instruction (SP−4)(SP−1)(SP−2) ← PC$_{11-0}$; (SP−3) ← (MBE, RBE, PC$_{13,12}$); PC$_{13-0}$ ← (taddr)$_{5-0}$ + (taddr+1); SP ← SP−4 When the table is specified by any other instructions, the (taddr), (taddr+1) instructions are executed. (Note 3)	Depends on the referenced instruction

Notes:

(1) Appropriate instructions are selected from BR !addr, BRCB !caddr, and BR $addr by the assembler. (BR !addr is not available on the μPD75104/104A)

(2) When executing the IN/OUT instruction, either MBE must be reset to 0, or MBE and MBS must be set to 1 and 15, respectively.

(3) TBR and TCALL are pseudoinstructions used only to specify these tables.

Table 5. Digital Port Features

Port Number	Type	Operational Features	Comments
Port 0	4-bit input	Can be read or tested at any time regardless of the functional mode of the shared pins.	Pins are also used for SI, SO, $\overline{SCK}$, and INT4.
Port 1	4-bit input	Can be read or tested at any time regardless of the functional mode of the shared pins.	Pins also used for INT0-INT3. On the μPD75104A/108A, internal pull-up resistors are available for each line as a mask option.
Port 3 Port 6	4-bit I/O	Can be set for input or output mode in 1-bit units.	On the μPD75104A/108A, internal pull-up resistors are available for each line of port 6 as a mask option.
Port 2 (Note 1)	4-bit I/O	Can be set for input or output mode in 4-bit units.	Pins are also used for PTO0, PTO1, and PCL.
Port 4 Port 5 Port 7 Port 8 Port 9 (Note 1)	4-bit I/O	Can be set for input or output mode in 4-bit units. Ports 4-5, 6-7, and 8-9 can be paired together to enable 8-bit data transfers.	On the μPD75104A/108A, internal pull-up resistors are available for each line as a mask option.
Port 12 Port 13 Port 14 (Notes 1, 2)	4-bit I/O	Can be set for input or output mode in 4-bit units; Ports 12 and 13 can be paired to form a single 8-bit I/O port.	Except for μPD75P108/P116, internal pull-up resistors are available for each line as a mask option.

Notes:

(1) Ports 2-9 and 12-14 can directly drive LEDs. Total current must not exceed 200 mA (peak).

(2) The output stage of ports 12-14 contains an N-channel open-drain transistor capable of withstanding 12 V.

Input/Output Ports

There are thirteen 4-bit ports; some are I/O ports and some are input only. Figure 4 shows the structure of the ports and table 5 lists the features. Figure 4 also shows the structure of inputs and outputs of the other pins.

Clock Generator

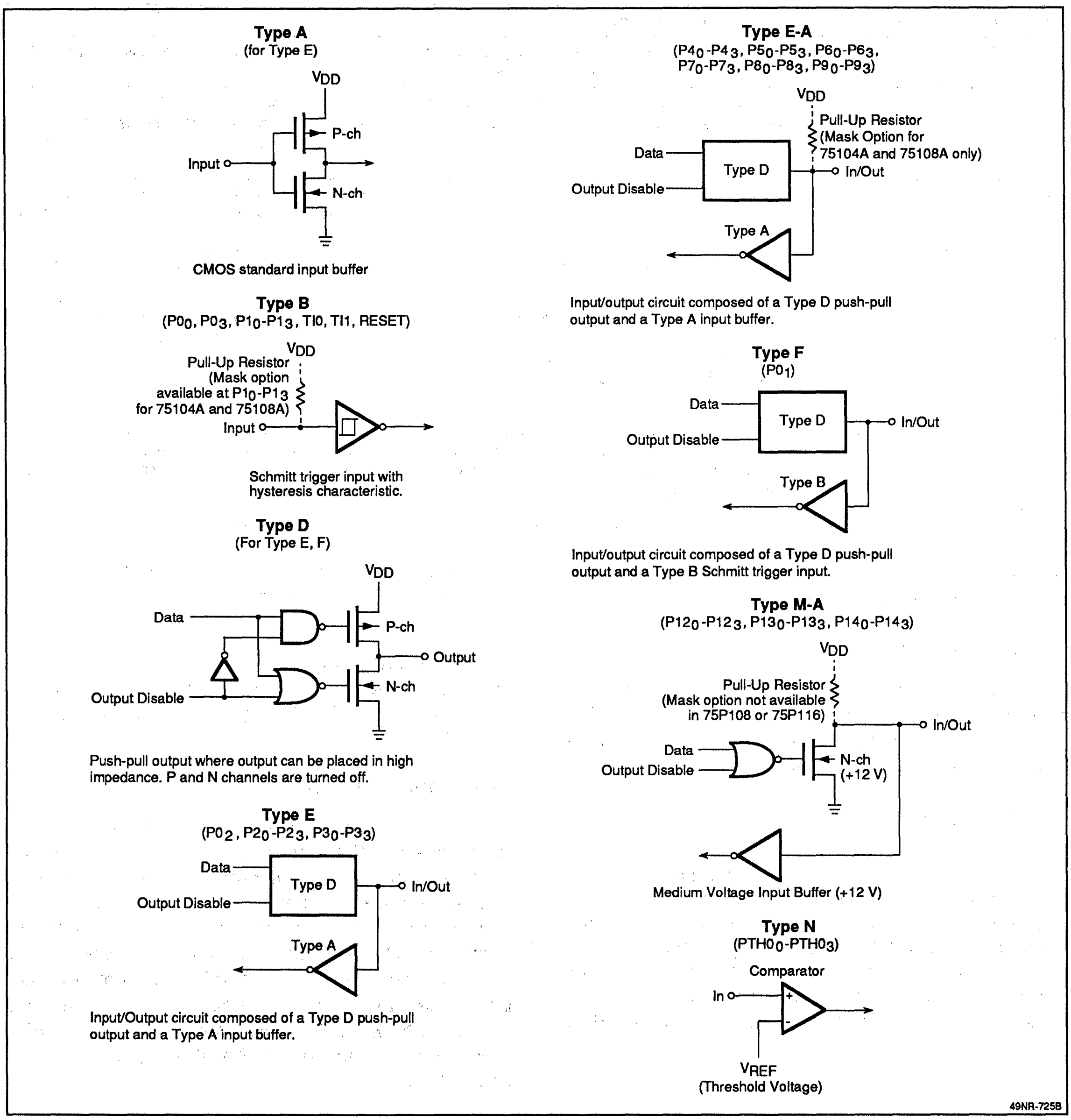
The clock generator (figure 5) uses the crystal inputs X1 and X2 as a time base to provide clocks for the μPD751xx/P1xx . The generator consists of an oscillator, frequency dividers, multiplexers, and two control registers, (PCC and CLOM). By programming PCC and CLOM, frequencies derived from the crystal are supplied to the CPU, the interval timer, the timer/event counter, the serial interface, and the output pin, PCL.

The PCC register controls the HALT and STOP logic and can also be used to set the CPU to operate at one of three speeds. The CLOM register controls the output clock PCL.

Basic Interval Timer

The basic interval timer (figure 6) is used to provide continuous real-time interrupts. It consists of a multiplexer, an 8-bit free-running counter, and a 4-bit control register (BTM). Each time the counter reaches FFH it causes an interrupt, overflows to 00H and continues to count. The BTM register is used to select one of four clock inputs to the counter as well as clear the counter and its interrupt request. The counter can generate 250 ms interrupts with a 4.19 MHz crystal and also provides oscillator stabilization time when the chip comes out of the STOP mode.

Figure 4. I/O Circuits



Type A
(for Type E)

CMOS standard input buffer

Type B
($P0_0$, $P0_3$, $P1_0$-$P1_3$, TI0, TI1, RESET)

Schmitt trigger input with hysteresis characteristic.

Type D
(For Type E, F)

Push-pull output where output can be placed in high impedance. P and N channels are turned off.

Type E
($P0_2$, $P2_0$-$P2_3$, $P3_0$-$P3_3$)

Input/Output circuit composed of a Type D push-pull output and a Type A input buffer.

Type E-A
($P4_0$-$P4_3$, $P5_0$-$P5_3$, $P6_0$-$P6_3$, $P7_0$-$P7_3$, $P8_0$-$P8_3$, $P9_0$-$P9_3$)

Input/output circuit composed of a Type D push-pull output and a Type A input buffer.

Type F
($P0_1$)

Input/output circuit composed of a Type D push-pull output and a Type B Schmitt trigger input.

Type M-A
($P12_0$-$P12_3$, $P13_0$-$P13_3$, $P14_0$-$P14_3$)

Medium Voltage Input Buffer (+12 V)

Type N
($PTH0_0$-$PTH0_3$)

Figure 5. Clock Generator

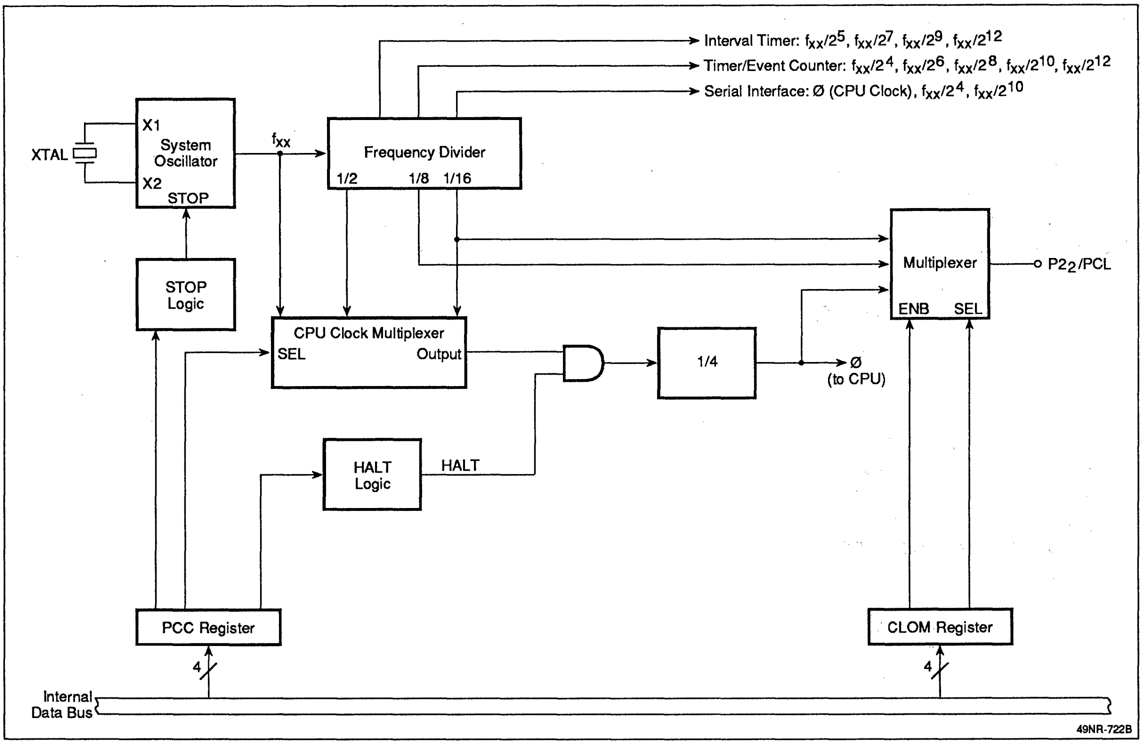
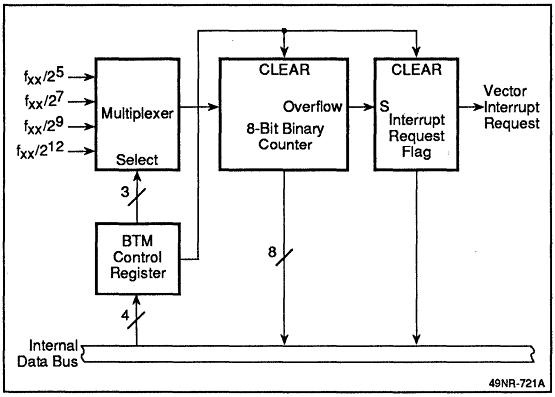

Interval Timer: $f_{xx}/2^5$, $f_{xx}/2^7$, $f_{xx}/2^9$, $f_{xx}/2^{12}$
Timer/Event Counter: $f_{xx}/2^4$, $f_{xx}/2^6$, $f_{xx}/2^8$, $f_{xx}/2^{10}$, $f_{xx}/2^{12}$
Serial Interface: Ø (CPU Clock), $f_{xx}/2^4$, $f_{xx}/2^{10}$
XTAL
X1
System Oscillator
X2
STOP
f_{xx}
Frequency Divider
1/2
1/8
1/16
Multiplexer
ENB
SEL
$P2_2$/PCL
STOP Logic
CPU Clock Multiplexer
SEL
Output
1/4
Ø
(to CPU)
HALT Logic
HALT
PCC Register
CLOM Register
4
4
Internal Data Bus
49NR-722B

Figure 6. Basic Interval Timer



Timer/Event Counters

Each of the two timer/event counters (figure 7) consists of an 8-bit modulo register, 8-bit comparator, 8-bit count register, clock multiplexer, mode control register, and a TOUT flip flop. There is also some control logic so that the timer's TOUT flip flop can be sent to port 2.

The two timers differ only by the clock selection to the count register. Timer 0 has an $f_{XX}/16$ clock input, and Timer 1 has an $f_{XX}/4096$ clock input. TI0 and TI1 can also be used as external clock inputs to count events.

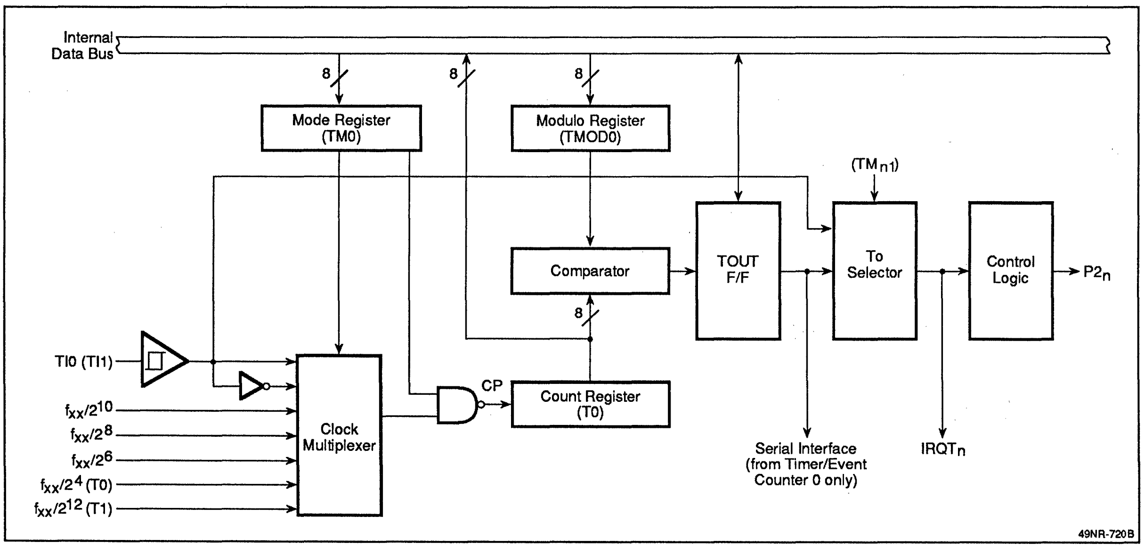
An 8-bit value is loaded into the modulo register, and a count register clock is selected by the clock multiplexer, via control register TM0 (or TM1 for counter 1). The count register is incremented each time it receives a CP pulse. When the value in the count register is equal to the count in the modulo register, the comparator generates a signal which toggles the TOUT flip flop and causes the count register to be reset to 00H. The count register will continue to count up unless stopped. Each time TOUT changes state it causes an interrupt. This signal can also be used as a clock for the serial interface.

Serial Interface

The 8-bit serial interface (figure 8) allows the μPD751xx/P1xx to communicate with other NEC or NEC-like serial interfaces. It consists of an 8-bit shift register, 3-bit counter, clock multiplexer, and control register SIOM. The three-wire interface consists of the serial data in (SI), serial data out (SO), and serial shift clock ($\overline{SCK}$).

The 8-bit shift register is loaded with a byte of data, and when bit 3 of SIOM is set, 8 clock pulses are generated. These pulses shift data out the SO line and data in from the SI line, thus, communicating in full duplex. Each time bit 3 of SIOM is set, a burst of eight clock pulses is generated and eight bits of data will be sent. Data may be sent either LSB or MSB first. The interface may also be set to receive data only; in this case SO is in the high-impedance state. One of four internal clocks or an external clock may be used to clock the data.

Figure 7. Timer/Event Counter



Comparator Port

The four-input comparator port (figure 9) contains a resistor ladder with 4-bit resolution, a 4-1 multiplexer, a comparator, a 1-4 demultiplexer, and an input result register, PTH0. This port is controlled by the 8-bit PTHM register and operates in a sequential manner. When bit 7 of the PTHM starts the comparator, the comparator reads and converts input PTH3, then the others in order, ending with PTH0. Then the PTH0 register may be read to get the results.

The user may select a slow or fast conversion time. With a 4.19 MHz crystal, total time required to convert all four inputs is 258 μs and 32.3 μs, respectively.

Bit Sequential Buffer

The bit sequential buffer is 16 bits of general-purpose RAM located in the upper half of memory bank 15, and is the only general-purpose RAM in this area. All other locations in this bank contain either the on-chip peripheral control registers or are unused addresses. A typical application of this buffer might be to store data for the next serial output or to store data from a serial input. It could also be used to store data which is to be sent from a port. This area can be bit, nibble, or byte manipulated.

Interrupts

The μPD751xx/P1xx family interrupts (figure 10) are all vectored; there are five external and four internal interrupts. Table 4 gives a summary of the interrupts. The hardware provides two levels of interrupt nesting; interrupt priorities can be changed via register IPS. Inputs INT2 and INT3 will detect rising edges and generate an interrupt request flag which is testable. Neither INT2 nor INT3 will cause an interrupt, but they can be used to release the STANDBY mode.

Figure 8. Serial Interface Block Diagram

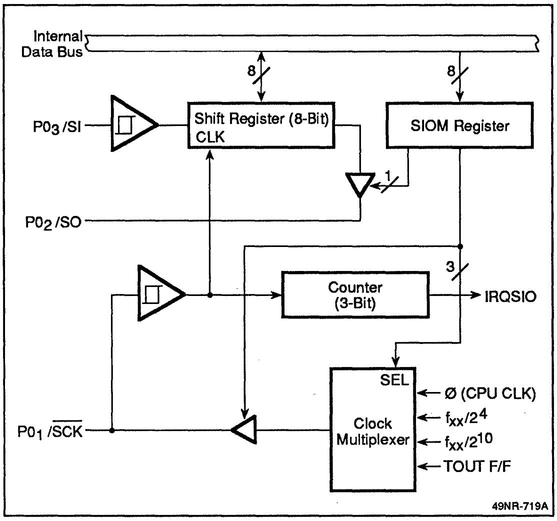


Figure 9. Comparator Port

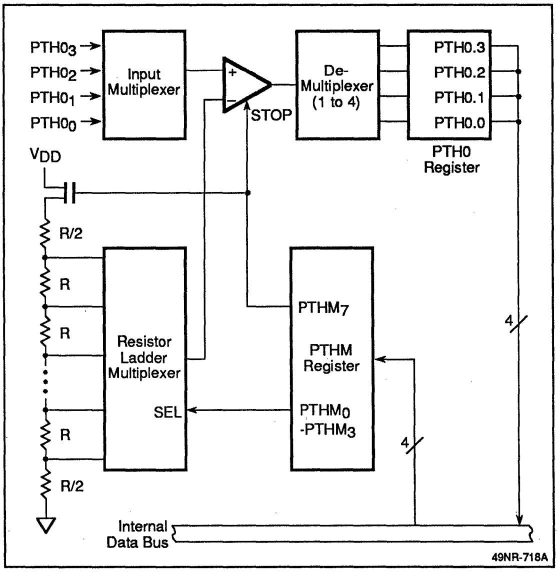
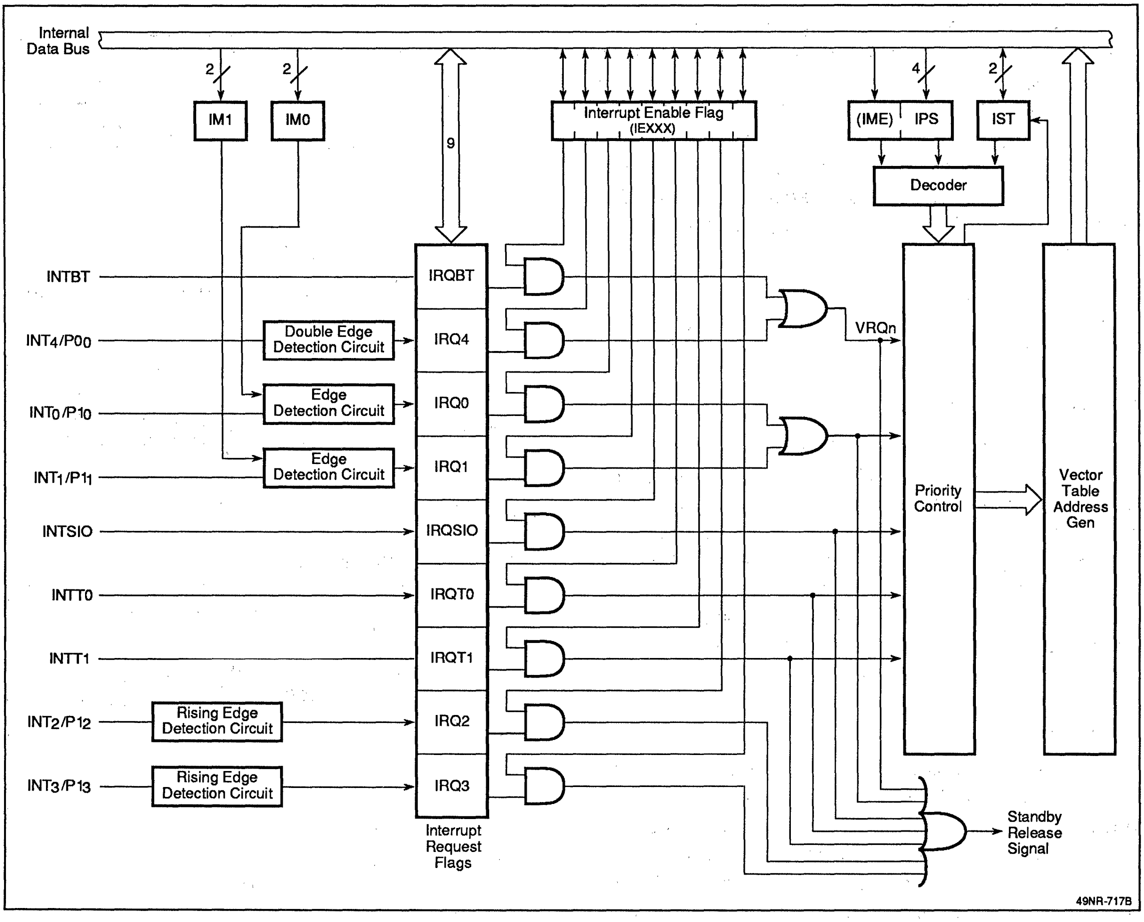


Figure 10. Interrupt Controller Block Diagram



Standby Modes

The standby mode is summarized in table 7 and consists of three submodes, HALT, STOP, and Data Retention.

HALT mode. The HALT mode is entered by executing the HALT instruction. In this mode, the clock to the CPU is shut off (thus stopping the CPU), while all other parts of the chip remain fully functional.

STOP mode. The STOP mode is entered by executing the STOP instruction. In this mode, the chip's main system oscillator is shut off, thereby stopping all portions of the chip.

The HALT and STOP modes are released by a $\overline{\text{RESET}}$ or by any interrupt request.

Data Retention mode. This mode may be entered after entering the STOP mode. Here, supply voltage V_{DD} may be lowered to 2 volts to further reduce power consumption. The contents of the RAM and registers are retained. This mode is released by first raising V_{DD} to the proper operating range, then releasing the STOP mode.

Table 6. Interrupt Sources

Interrupt Source	Operation and Source	Internal/ External	Interrupt Priority	Vectored Interrupt Request Signal (Vector Table Address)
INTBT	Reference time interval signal from basic interval timer	Internal	1	VRQ1 (002H)
INT4	Both rising and falling edge detection	External		
INT0	Selection of rising or falling edge detection	External	2	VRQ2 (0004H)
INT1	Selection of rising or falling edge detection	External		
INTSIO	Serial data transfer completion signal	Internal	3	VRQ3 (0006H)
INTT0	Coincidence signal between timer/counter 0, or edge detection of TI0 input	Internal/External	4	VRQ4 (0008H)
INTT1	Coincidence signal between timer/counter 1, or edge detection of TI1 input	Internal/External	5	VRQ5 (000AH)
INT2	Rising edge detection	External	Testable input signals (IRQ2 and IRQ3 are set)	
INT3	Rising edge detection	External		

Table 7. Standby Mode Operation

Item	STOP Mode	HALT Mode
Setting the mode	STOP instruction	HALT instruction
Clock oscillator	The main system clock oscillator is stopped	Only CPU clock ϕ is stopped (oscillation continues)
Basic interval timer	Operation stopped	Can Operate (IRQBT is set by reference time interval)
Serial interface	Can operate only when external $\overline{\text{SCK}}$ input is selected for serial clock. (Note 1)	Can operate if other than CPU clock ϕ is specified as serial clock
Timer/event counter	Can operate only when TIn (n = 0, 1) pin input is selected for count clock	Can operate
Clock output circuit	Stops operation	Can operate if other than CPU clock ϕ is specified
CPU	Operation stopped	Operation stopped
Retained data	Contents of all registers (general registers, flags, mode registers, and output latches) and contents of data memory retained	
Release signal	Interrupt request signal (enabled with interrupt enable flag) from operating hardware or $\overline{\text{RESET}}$	

Notes:

(1) Can also operate with TI0 selected as the serial clock, but only when Timer/Event Counter 0 is operated with an external TI0 input.

RESET and the Reset Generator

The power-on-reset (POR) generator (figure 11) is always present in the µPD75P108, not present in the µPD75P116, and available by mask option in the mask ROM devices.

The POR circuit generates a one-shot pulse by detecting the supply voltage rising edge. Use of this pulse is determined by mask option:

(1) Both SWA and SWB are ON.
When the power supply rising edge is detected, the internal reset signal (RES) is generated and the power-on flag (PONF) is set at the same time.

(2) SWA only is ON.
When the power supply rising edge is detected, PONF is set. (RES is not generated automatically.)

(3) Both SWA and SWB are OFF.
The power-on reset generator and and power-on flag are disabled. RES is generated only by the RESET input.

Figure 11. Power-On-Reset Signal Generator and PONF

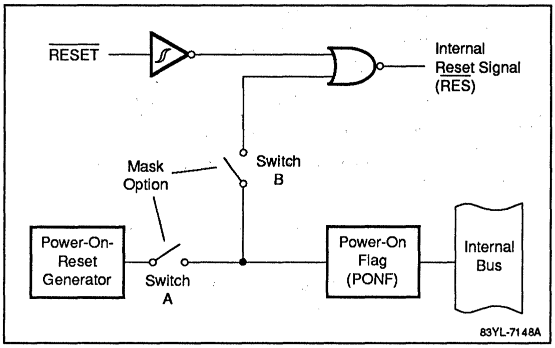


Table 8. State of the Device after Reset

Hardware		RESET Inputted During Standby Mode	RESET Inputted During Normal Operation or Power-on
Program counter (PC)		The six low-order bits of program memory address 000H are loaded into PC13–PC8. The contents of address 0001H are loaded into PC7–PC0.	
PSW	Carry flag (CY)	Held	Undefined
	Skip flags (SK0, SK1, SK2)	0	0
	Interrupt status flags (IST0, IST1)	0	0
	Bank enable flags (MBE, RBE)	Bit 6 of program memory address 000H is loaded into RBE and bit 7 into MBE.	
Stack pointer (SP)		Undefined	Undefined
Data memory (RAM)		Held (Note 1)	Undefined
General purpose registers (X, A, H, L, D, E, B, C)		Held	Undefined
Bank selection registers (MBS, RBS)		0, 0	0, 0
Basic interval timer	Counter (BT)	Undefined	Undefined
	Mode register (BTM)	0	0
Timer/event counter (n=0, 1)	Counter (Tn)	0	0
	Modulo register (TMODn)	FFH	FFH
	Mode register (TMn)	0	0
	TOEn, TOFn	0, 0	0, 0
Serial interface	Shift register (SIO)	Held	Undefined
	Mode register (SIOM)	0	0

Table 8. State of the Device after Reset (cont)

Hardware		$\overline{\text{RESET}}$ Inputted During Standby Mode	$\overline{\text{RESET}}$ Inputted During Normal Operation or Power-on
Clock generator and clock output circuit	Processor clock control register (PCC)	0	0
	Clock output mode register (CLOM)	0	0
Interrupt function	Interrupt request flags (IRQxxx)	Reset to 0	Reset to 0
	Interrupt enable flags (IExxx)	0	0
	Priority selection register (IPS)	0	0
	INT0, and INT1 mode registers (IM0, IM1)	0, 0	0, 0
Digital ports	Output buffers	Off	Off
	Output latches	Cleared (to 0)	Cleared (to 0)
	Input/output mode registers (PMGA, PMGB, PMGC)	0	0
Bit sequential buffer		0	0
Analog port	PTH00-03 input latches	Undefined	Undefined
	Mode register (PTHM)	0	0
Power-on flag (PONF)		Undefined	1, Undefined (Note 2)
Bit sequential buffer (BSB0-3)		0	0

Notes:

(1) Addresses 0F8H to 0FDH are undefined after RESET.

(2) This value is 1 upon power-on-reset and undefined during normal operation.

EPROM Write/Verify

The μPD75P108 contains 8192 bytes of EPROM, while the μPD75P116 has 16256 bytes. Table 9 shows the pin functions during the Write/Verify cycles. Note that it is not necessary to enter an address, since the address is updated by pulsing the clock pins. When V_{DD} = 6 V and V_{PP} = 21 V in the μPD75P108 (or V_{PP} = 12.5 V in the μPD75P116) are applied, the EPROM is placed in the write/verify mode. The operation is selected by the MD0–MD3 pins, as shown in table 10.

Table 9. EPROM Write/Verify Pin Functions

Pin Name	Function
X1, X2	After a write/verify write, the X1, and X2 clock pins are pulsed. (Note that these pins are also pulsed during a read.)
MD0–MD3	These are the operation mode selection pins.
$P4_0$–$P4_3$ (four low-order bits) $P5_0$–$P5_3$ (four high-order bits)	8-bit data input/output pins for write/verify
V_{DD}	Supply voltage. Normally 5 volts; 6 volts is applied during write/verify
V_{PP}	Normally 5 volts; V_{PP} = 21 V in the μPD75P108 (or V_{PP} = 12.5 V in the μPD75P116) during write/verify

Notes:

(1) A cover should be placed over the UV erase window. The OTP devices do not have windows, thus the EPROM contents cannot be erased.

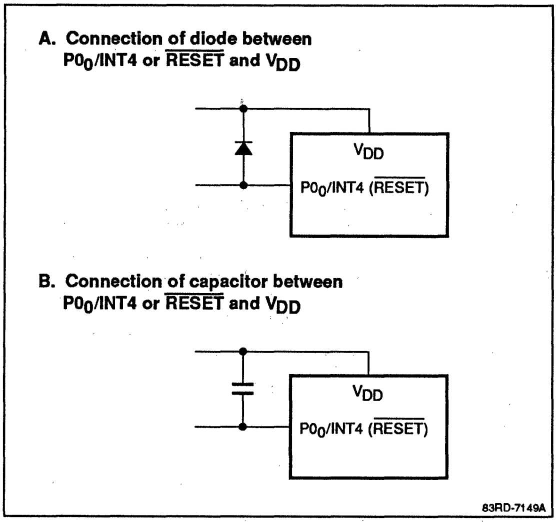
Caution

Apart from their normal functions, The $P0_0$/INT4 and $\overline{\text{RESET}}$ pins are used to test the internal operation of the programmable devices. The test mode is entered by applying a voltage greater than V_{DD} to either of these pins.

For this reason, care must be taken to limit the voltage applied to these two pins. For example, it is conceivable that even during normal operation enough spurious noise may be present to set the chip into the test mode. If this happens, further normal operation is impossible. Consequently, it is important that interwiring noise be suppressed as much as possible. If this is inconvenient, anti-noise measures, like those shown in figure 12, should be implemented.

The write/verify mode is entered by applying 6 volts to V_{DD} and V_{PP} = 21 V in the μPD75P108 (or V_{PP} = 12.5 V in the μPD75P116). Mode is determined by the setting of the MD0-MD3 pins; all other pins are tied to ground by pulldown resistors.

Figure 12. Noise Reduction Techniques



A. Connection of diode between $P0_0$/INT4 or $\overline{\text{RESET}}$ and V_{DD}

B. Connection of capacitor between $P0_0$/INT4 or $\overline{\text{RESET}}$ and V_{DD}

Table 10. Write/Verify Operation

V_{PP} = 21 V in the μPD75P108, V_{PP} = 12.5 in the μPD75P116, V_{DD} = +6.0 V

Operation Mode Specification				
MD0	**MD1**	**MD2**	**MD3**	**Operation Mode**
1	0	1	0	Clear program memory address
0	1	1	1	Write mode
0	0	1	1	Verify mode
1	X	1	1	Program inhibit

Notes:

(1) X = Don't care.

EPROM Write/Verify Procedure

EPROMs can be written at high speed using the follow ing procedure:

(1) Pull unused pins to V_{SS} through resistors. Set the X1 pin low.

(2) Supply 5 volts to the V_{DD} and V_{PP} pins.

(3) Wait for 10 μs.

(4) Select the *clear program memory address* mode.

(5) For the μPD75P108, supply 6 volts to V_{DD} and 21.0 volts to V_{PP}. For the μPD75P116, supply 6 volts to V_{DD}, and 12.5 volts to V_{PP}.

(6) Select the *program inhibit* mode.

(7) Write data in the 1 ms write mode.

(8) Select the *program inhibit* mode.

(9) Select the *verify* mode. If the data is correct, proceed to step 10. If not, repeat steps 7, 8, and 9.

(10) Perform one additional write with an MD0 pulse width equal in ms to the number of *writes* performed in step 7, times 1 ms.

(11) Select the *program inhibit* mode.

(12) Apply four pulses to the X1 pin to increment the program memory address by one.

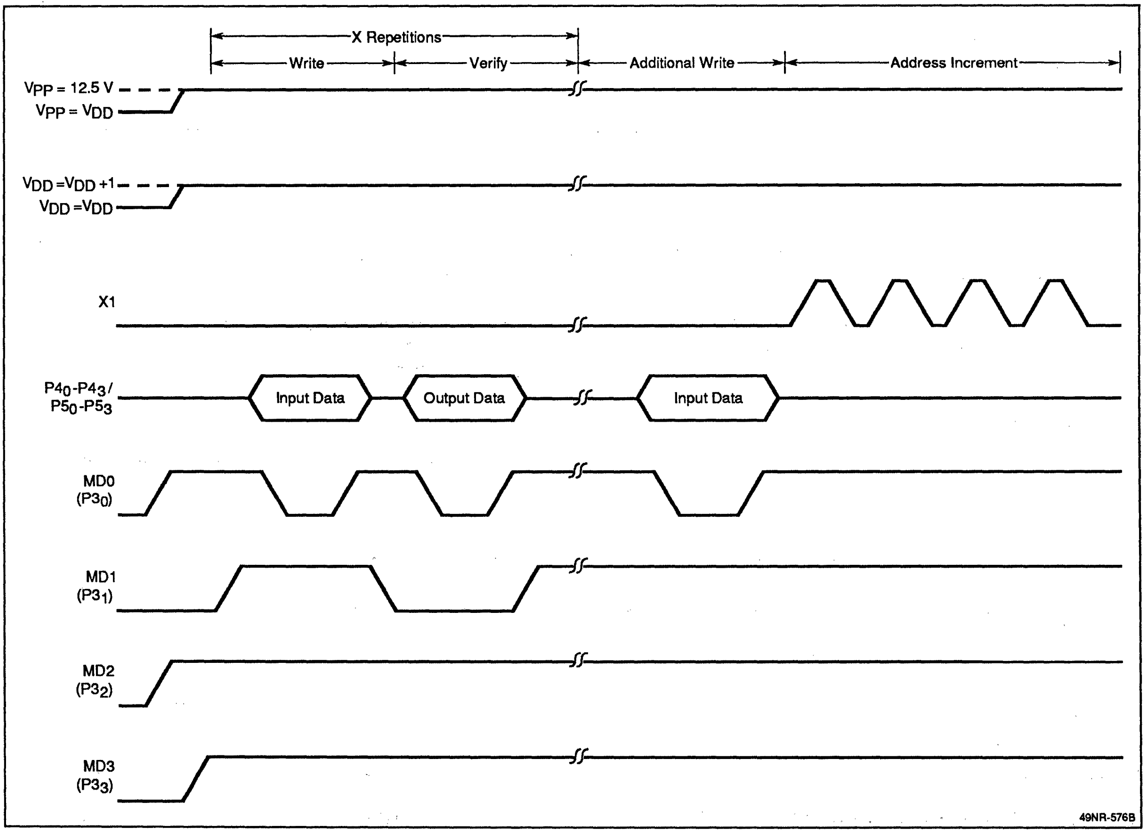
(13) Repeat steps 7-12 until the end address is reached.

(14) Select the *clear program memory address* mode.

(15) Return the V_{DD} and V_{PP} pins back to + 5 volts.

(16) Turn off the power.

Figure 13. EPROM Write/Verify Cycle Timing



EPROM Read Procedure

The EPROM contents can be read by using the following procedure:

(1) Pull unused pins to V_{SS} through resistors. Set the X1 pin low.

(2) Supply 5 volts to the V_{DD} and V_{PP} pins.

(3) Wait for 10 μs.

(4) Select the *clear program memory address* mode.

(5) For the μPD75P108, supply 6 volts to V_{DD} and 21.0 volts to V_{PP}. For the μPD75P116, supply 6 volts to V_{DD}, and 12.5 volts to V_{PP}.

(6) Select the *program inhibit* mode.

(7) Select the *verify* mode. Apply four pulses to the X1 pin. Every four clock pulses will output the data stored in one address.

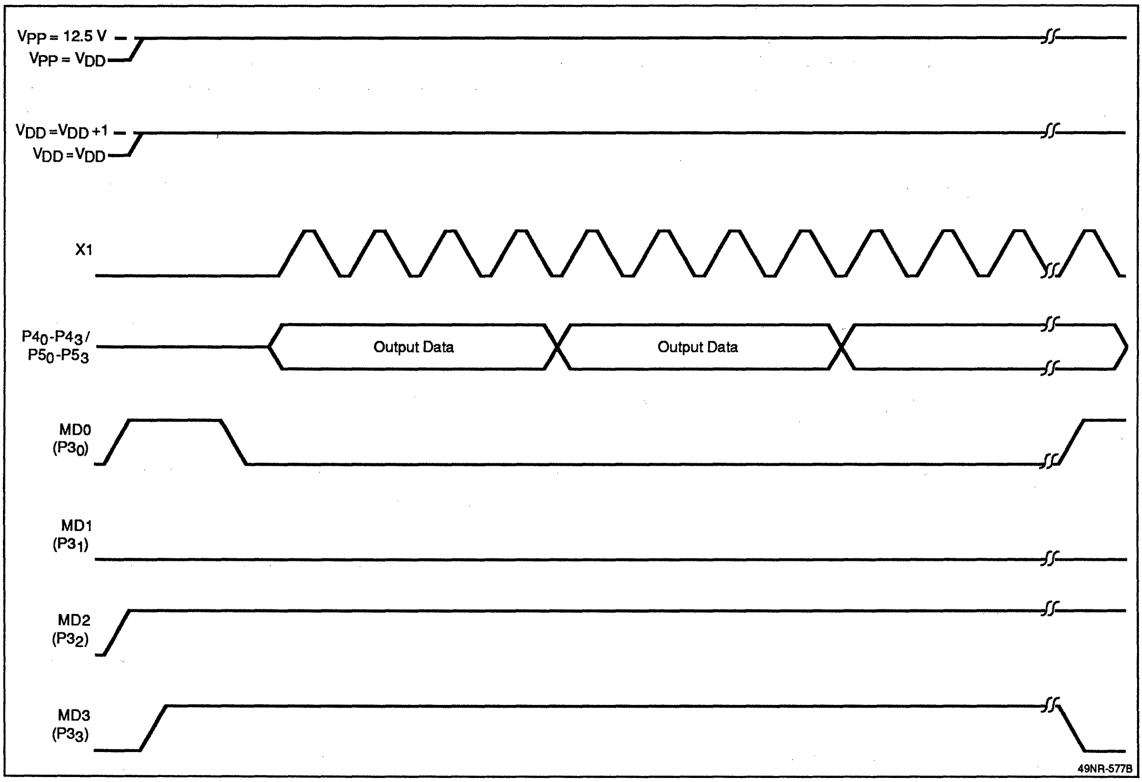
(8) Select the *program inhibit* mode.

(9) Select the *clear program memory address* mode.

(10) Return the V_{DD} and V_{PP} pins to + 5 volts.

(11) Turn off the power.

Figure 14. EPROM Read Cycle Timing



Program Memory Erase (μPD75P108DW only)

The μPD75P108DW allows the programmed data contents to be erased by light rays whose wavelength is shorter than about 400 nm. The programmed data contents may also be erased if the uncovered window is exposed to direct sunlight or a fluorescent light for several hours. Thus, to protect the data contents, cover the window with an opaque film. NEC provides quality-tested shading film with each UV EPROM shipment.

For normal EPROM erase, place the device under an ultraviolet light source (254 nm). The minimum amount of radiation exposure required to erase the μPD75P108DW completely is 15 Ws/cm^2 (ultraviolet ray strength times erase time). This corresponds to about 15 to 20 minutes when using a UV lamp of 12 V_{PP} μW/cm^2. However, the erase time may be prolonged if the UV lamp is old or if the device window is dirty. The distance between the light source and the window should be 2.5 cm or less.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (μPD751xx)

T_A = 25°C

Supply voltage, V_{DD}	–0.3 to +7.0 V
Input voltage, V_{I1} (ports 12-14)	–0.3 to V_{DD} + 0.3 V
Input voltage, V_{I2} (ports 12-14; internal pull-up resistor)	–0.3 to V_{DD} + 0.3 V
Input voltage, V_{I2} (ports 12-14; open drain)	–0.3 to +13 V (Note 1)
Output voltage, V_O	–0.3 to V_{DD} + 0.3 V
High-level output current, I_{OH} (Single pin)	–15 mA
High-level output current, I_{OH} (Total of all pins)	–30 mA
Low-level output current, I_{OL} (Single pin)	30 mA pk
	15 mA rms (Note 2)
Low-level output current, I_{OL} (Total of ports 0, 2–4, 12–14)	100 mA pk
	60 mA rms (Note 2)
Low-level output current, I_{OL} (Total of ports 5–9)	100 mA pk
	60 mA rms (Note 2)

Operating temperature, t_{OPT}	–40 to + 85°C
Storage temperature, t_{STG}	–65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Notes:

(1) When applying more than 10 V to ports 12, 13, or 14, the external pull-up resistor must be at least 50 kΩ.

(2) rms value = pk x (duty cycle)$^{1/2}$.

Capacitance (μPD751xx)

V_{DD} = 0 V; T_A = 25°C

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_{IN}	15	pF	f = 1 MHz; all unmeasured pins returned to ground
Output capacitance	C_{OUT}	15	pF	
I/O capacitance	C_{IO}	15	pF	

Oscillator Characteristics (All devices)

μPD751xx: T_A = –40 to +85°C; V_{DD} = 2.7 to 6.0 V
μPD75P108: T_A = –10 to +85°C; V_{DD} = 4.5 to 5.5 V
μPD75P116: T_A = –40 to +85°C; V_{DD} = 4.5 to 5.5 V

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 15A)	Oscillation frequency (Note 1)	f_{XX}	2.0		5.0	MHz	
	Oscillation stabilization time (Note 2)				4 (Note 3)	ms	After V_{DD} reaches oscillation voltage
Crystal resonator (Figure 15A)	Oscillation frequency (Note 1)	f_{XX}	2.0		5.0	MHz	
	Oscillation stabilization time (Note 2)				10 (Note 3)	ms	V_{DD} = 4.5 to 6.0 V
					30 (Note 3)	ms	
External clock (Figure 15B)	X1 input frequency (Note 1)	f_{XX}	2.0		5.0	MHz	
	X1 input high/low level width	t_{XH}, t_{XL}	100		250	ns	

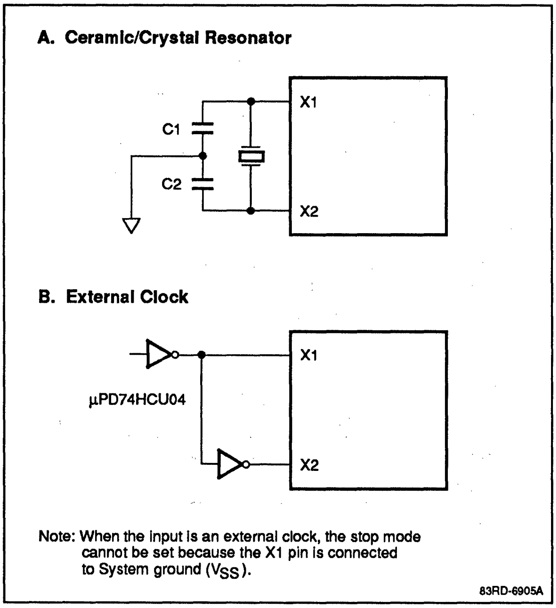
Notes:

(1) The oscillation frequency and X1 input frequency are included only to show the characteristics of the oscillators. Refer to the AC Characteristics table for actual instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillator to stabilize after voltage is applied or the STOP mode is released.

(3) Values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's spec sheets.

Figure 15. System Clock Configurations



Note: When the input is an external clock, the stop mode cannot be set because the X1 pin is connected to System ground (V_{SS}).

Recommended Ceramic Resonators (μPD751xx)

Manufacturer	Part Number	C1 (pF)	C2 (pF)	Remarks
Murata	CSA 2.00MG	30	30	V_{DD} = 2.7 to 6.0 V
	CSA 4.19MG	30	30	V_{DD} = 3.0 to 6.0 V
	CSA 4.19MGU	30	30	V_{DD} = 2.7 to 6.0 V
	CST 4.19T (Note 1)	—	—	V_{DD} = 3.0 to 6.0 V
Kyocera	KBR-2.0MS	100	100	V_{DD} = 3.0 to 6.0 V
	KBR-4.0MS	33	33	
	KBR-4.19MS	33	33	
	KBR-4.9152M	33	33	

Notes:

(3) C1 and C2 are contained in the oscillator.

Recommended Crystal Resonator (μPD751xx)

Manufacturer	Frequency (MHz)	Part Number (note 1)	C1 (pF)	C2 (pF)	Remarks
Kinseki	4.19	HC-49/U	22	22	V_{DD} = 2.7 to 6.0 V

Notes:

(1) Equivalent series resistance of crystal must be less than 80 Ω

Comparator Characteristics (All devices)

μPD751xx: V_{DD} = 4.5 to 6.0 V; T_A = −40 to +85°C
μPD75P108: T_A = −10 to +85°C; V_{DD} = 4.5 to 5.5 V
μPD75P116: T_A = −40 to +85°C; V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Comparison accuracy	V_{ACOMP}			±100	mV	
Threshold voltage	V_{TH}	0		V_{DD}	V	
PTH input voltage	V_{IPTH}	0		V_{DD}	V	
Comparator consumption current	I_{COMP}		1		mA	Set PTHM7 to 1

DC Characteristics (μPD751xx)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	$0.7V_{DD}$		V_{DD}	V	Except ports 0, 1, 12-14, TI0, TI1, $\overline{\text{RESET}}$, X1, X2
	V_{IH2}	$0.8V_{DD}$		V_{DD}	V	Ports 0, 1, TI0, TI1 and $\overline{\text{RESET}}$
	V_{IH3}	$0.7V_{DD}$		V_{DD}	V	Ports 12-14; built-in pull-up resistor
		$0.7V_{DD}$		12	V	Ports 12-14; open drain
	V_{IH4}	V_{DD}−0.5		V_{DD}	V	X1, X2
Low-level input voltage	V_{IL1}	0		$0.3V_{DD}$	V	Except ports 0, 1, TI0, TI1, $\overline{\text{RESET}}$, X1, X2
	V_{IL2}	0		$0.2V_{DD}$	V	Ports 0, 1, TI0, TI1 and $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2
High-level output voltage	V_{OH}	V_{DD}−1.0			V	V_{DD} = 4.5 to 6.0 V; I_{OH} = −1 mA
		V_{DD}−0.5			V	V_{DD} = 2.7 to 6.0 V; I_{OH} = −100 μA
Low-level output voltage	V_{OL}		0.35	2.0	V	Ports 0, 2-9; V_{DD} = 4.5 to 6.0 V; I_{OL}= 15 mA
			0.35	2.0	V	Ports 12-14; V_{DD} = 4.5 to 6.0 V; I_{OL}= 10 mA
				0.4	V	V_{DD} = 4.5 to 6.0 V; I_{OL}= 1.6 mA
				0.5	V	I_{OL}= 400 μA
High-level input leakage current	I_{LIH1}			3	μA	All except X1, X2, and ports 12-14; V_{IN} = V_{DD}
	I_{LIH2}			20	μA	X1, X2; V_{IN} = V_{DD}
	I_{LIH3}			20	μA	Ports 12-14 (with open drain); V_{IN} = 12 V
Low-level input leakage current	I_{LIL1}			−3	μA	All except X1, X2; V_{IN} = 0 V
	I_{LIL2}			−20	μA	X1, X2; V_{IN} = 0 V
High-level output leakage current	I_{LOH1}			3	μA	Other than Ports 12-14; V_{OUT} = V_{DD}
	I_{LOH2}			20	μA	Ports 12-14 (open drain); V_{OUT} = 12 V
Low-level output leakage current	I_{LOL}			−3	μA	V_{OUT} = 0 V
Internal pull-up resistor	R_L	15	40	70	kΩ	Ports 12-14; V_{DD} = 5.0 V ± 10%
		10		80	kΩ	Ports 12-14
Supply current (Note 1)	I_{DD1}		3	9	mA	V_{DD} = 5 V ± 10% (Notes 2, 3)
			0.55	1.5	mA	V_{DD} = 3 V ± 10% (Notes 3, 4)
	I_{DD2}		600	1800	μA	HALT mode; V_{DD} = 5 V ± 10% (Note 3)
			200	600	μA	HALT mode; V_{DD} = 3 V ± 10% (Note 3)
	I_{DD3}		0.1	10	μA	STOP mode; V_{DD} = 3 V ± 10%

Notes:

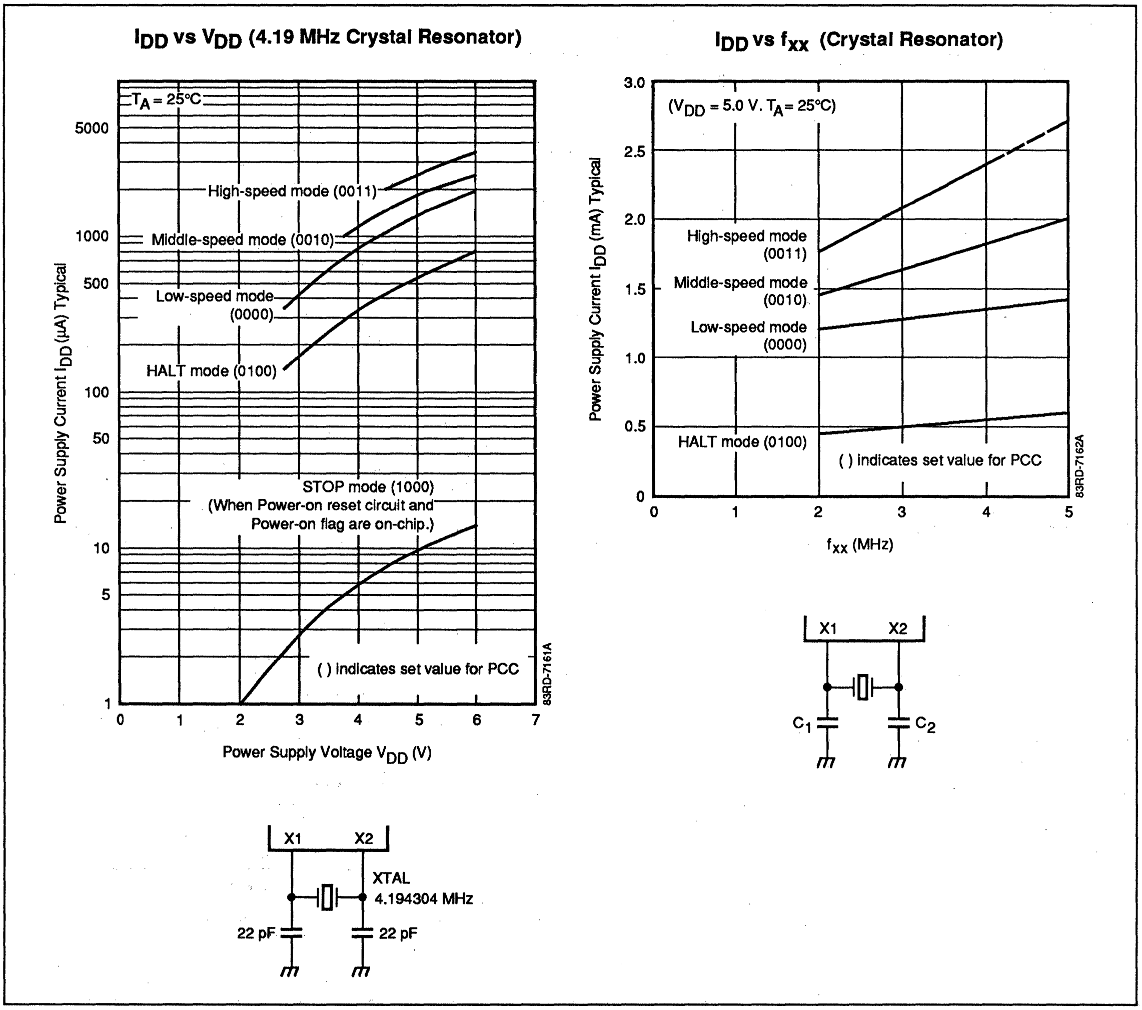
(1) Does not include pull-up resistor current, current through the power-on-reset circuit, or comparator current.

(2) When operated in the high-speed mode with the processor clock control register (PCC) set to 0011.

(3) f_{xx}= 4.19 MHz; C1 = C2 = 22 pF.

(4) When operated in the low-speed mode with the PCC set to 0000.

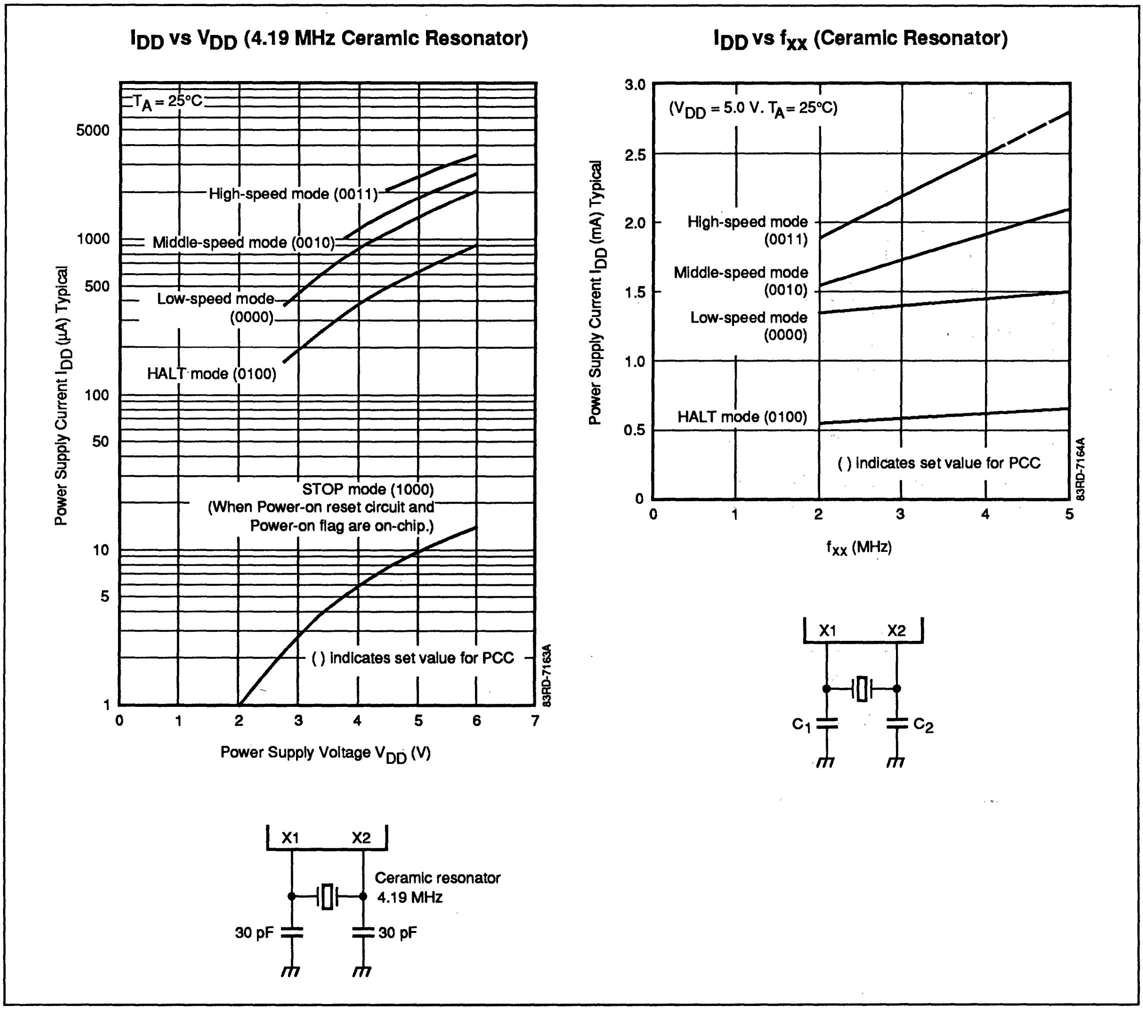
Figure 16. DC Characteristics (μPD751xx)



I_{DD} vs V_{DD} (4.19 MHz Crystal Resonator)

I_{DD} vs f_{XX} (Crystal Resonator)

Figure 16. DC Characteristics (μPD751xx) (cont)



I_{DD} vs V_{DD} (4.19 MHz Ceramic Resonator)

I_{DD} vs f_{XX} (Ceramic Resonator)

Figure 16. DC Characteristics (µPD751xx) (cont)

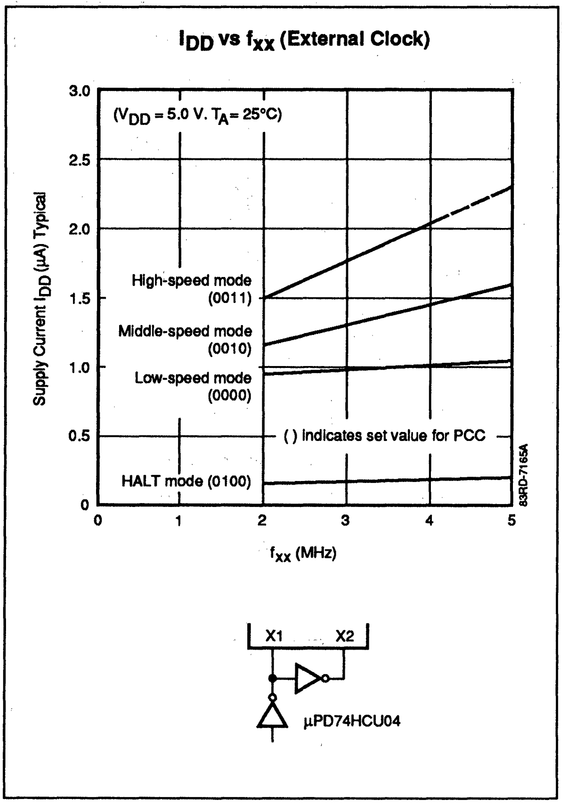
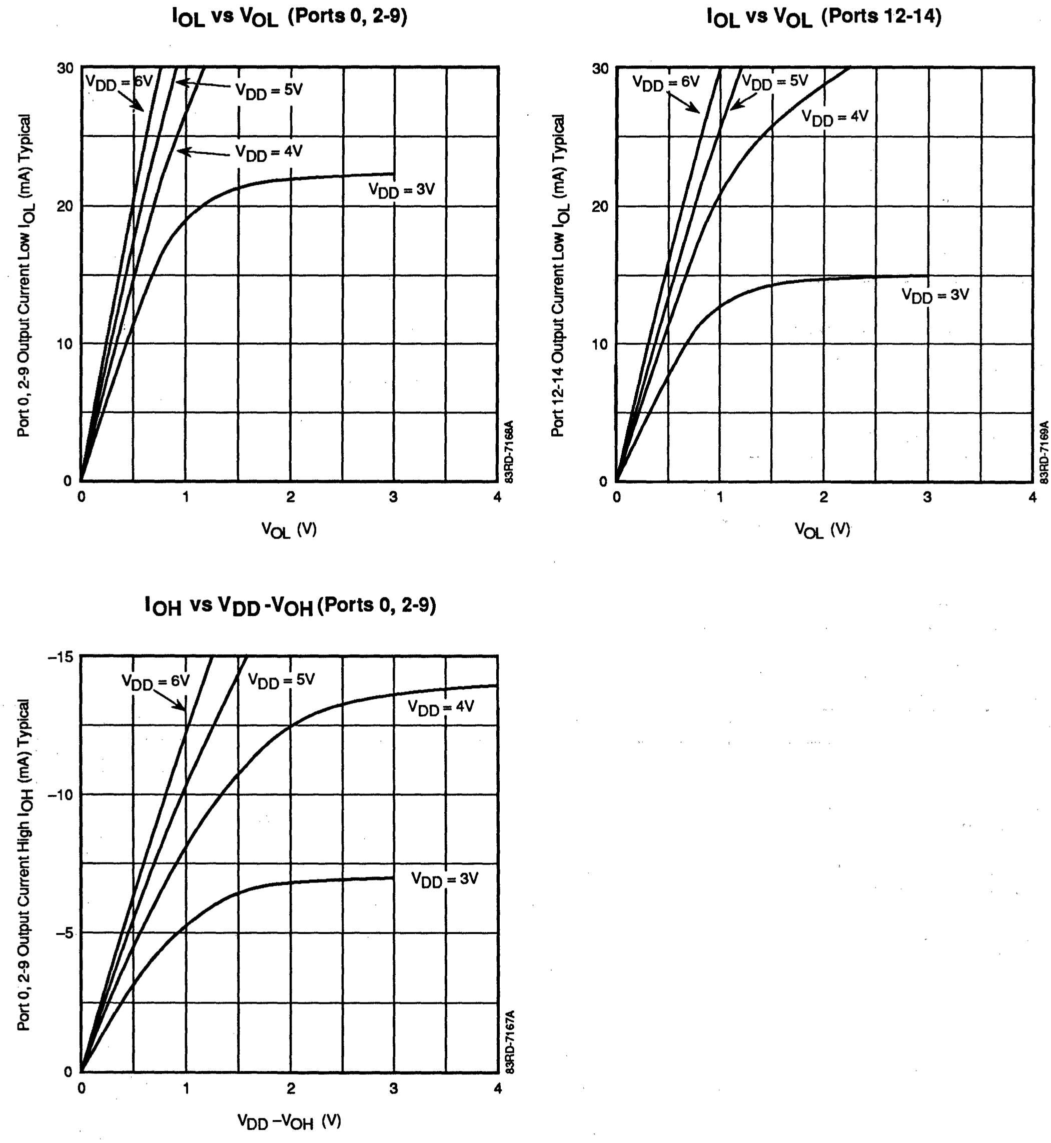


Figure 16. DC Characteristics (µPD751xx) (cont)



I_{OL} vs V_{OL} (Ports 0, 2-9)

I_{OL} vs V_{OL} (Ports 12-14)

I_{OH} vs V_{DD}-V_{OH} (Ports 0, 2-9)

Power-on-Reset Circuit Characteristics (All devices exept μPD75P116...notes 3, 4)

μPD751xx: T_A = −40 to +85°C
μPD75P108: T_A = −10 to +85°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Power-on reset voltage, high	V_{DDH}	4.5		V_{DD} max	V	μPD751xx: V_{DD} max = 6.0 V μPD75P108: V_{DD} max = 5.5 V
Power-on reset voltage, low	V_{DDL}	0		0.2	V	
Power supply voltage rise time	t_r	10		(Note 1)	μs	
Power supply voltage off time	t_{off}	1			s	
POR circuit consumption circuit ; μPD75108 only (Note 2)	I_{DDPR}		10	100	μA	V_{DD} = 5 V ± 10%
			2	20	μA	V_{DD} = 2.5 V

Notes:

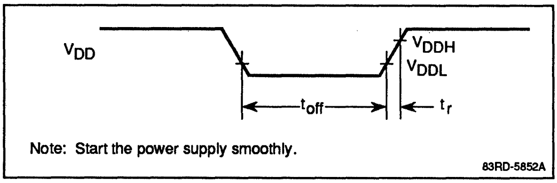
(1) $2^{17}/f_{xx}$ (31.3 ms at f_{xx} = 4.19 MHz)

(2) Current consumed when POR circuit or power-on flag is provided internally.

(3) Power supply voltage must be raised smoothly. See "Power-On-Reset" timing diagram.

(4) Power-on-reset circuit is available as a mask option on on all μPD751xx devices, is always provided with the μPD75P108, and not available on the μPD75P116.

Power-On-Reset Timing



AC Characteristics (μPD751xx)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Note 1)	t_{CY}	0.95		32	μs	V_{DD} = 4.5 to 6.0 V
		3.8		32	μs	V_{DD} = 2.7 to 6.0 V
TI0, TI1 input frequency	f_{TI}	0		1	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	V_{DD} = 2.7 to 6.0 V
TI0, TI1 input high- and low-level width	t_{TIH}, t_{TIL}	0.48			μs	V_{DD} = 4.5 to 6.0 V
		1.8			μs	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ cycle time	t_{KCY}	0.8			μs	Input; V_{DD} = 4.5 to 6.0 V
		0.95			μs	Output; V_{DD} = 4.5 to 6.0 V
		3.2			μs	Input; V_{DD} = 2.7 to 6.0 V
		3.8			μs	Output; V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ high and low level width	t_{KH}, t_{KL}	0.4			μs	Input; V_{DD} = 4.5 to 6.0 V
		0.5 t_{KCY}−50			ns	Output; V_{DD} = 4.5 to 6.0 V
		1.6			μs	Input; V_{DD} = 2.7 to 6.0 V
		0.5 t_{KCY}−150			ns	Output; V_{DD} = 2.7 to 6.0 V

AC Characteristics (μPD751xx) (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
SI vs $\overline{SCK}$ ↑ setup time	t_{SIK}	100			ns	
SI vs $\overline{SCK}$ ↑ hold time	t_{KSI}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO}			300	ns	V_{DD} = 4.5 to 6.0 V
				1000	ns	V_{DD} = 2.7 to 6.0 V
INT0-INT4 high- and low-level width	t_{INTH}, t_{INTL}	5			μs	
$\overline{RESET}$ low level width	t_{RSL}	5			μs	

Notes:

(1) Cycle time (minimum instruction execution time is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See figure 17.

Figure 17. Guaranteed Operating Range (μPD751xx)

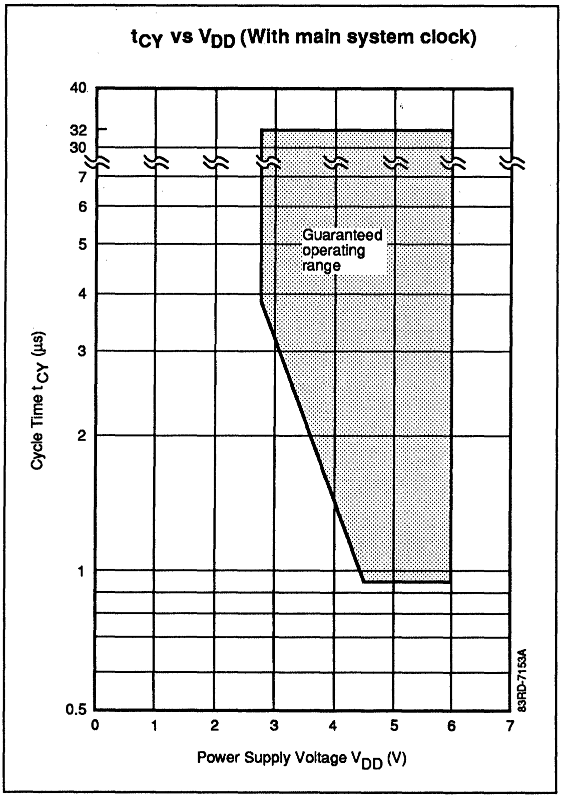


Figure 18. AC Timing Measurement Points (except Ports 0, 1, TI0, TI1, X1, X2, and $\overline{RESET}$)

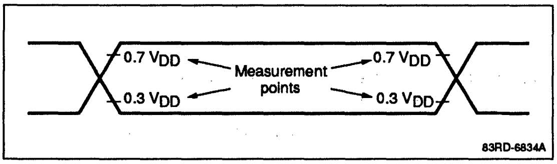


Figure 18A. Clock Timing Measurement Points

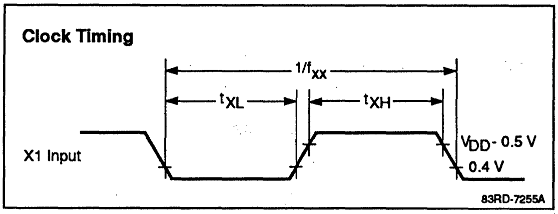


Figure 18B. TI Timing

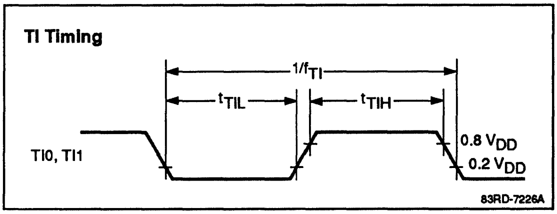


Figure 18C. Serial Transfer Timing

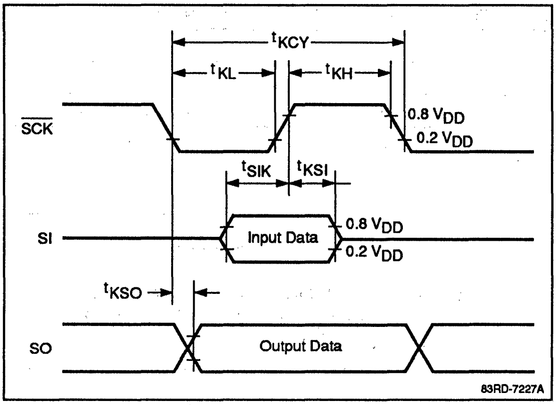


Figure 18D. Interrupt Input Timing

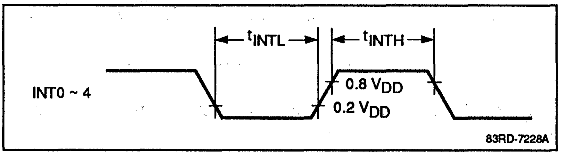


Figure 18E. RESET Input Timing

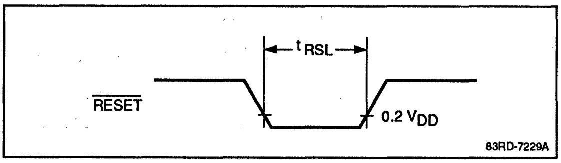
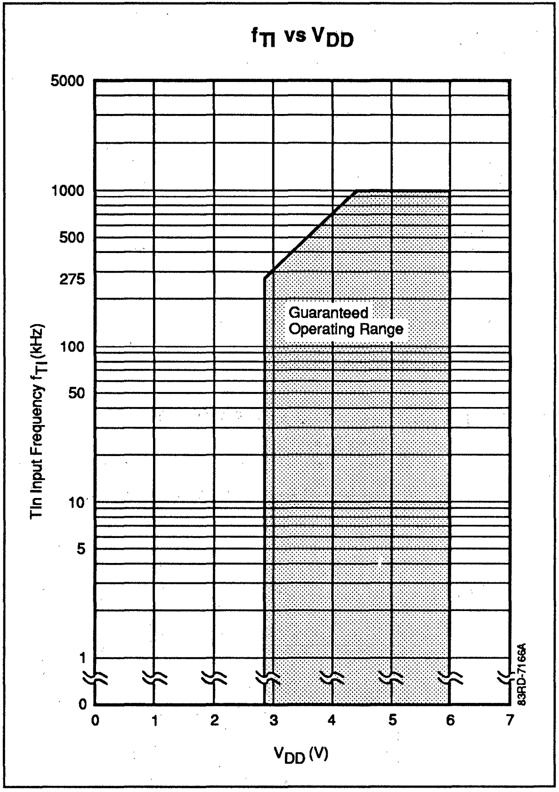


Figure 18F. f_{TI} vs V_{DD}



Data Memory STOP Mode, Low Voltage Data Retention Characteristics (µPD751xx)

T_A = −40 to +85°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		6.0	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	µA	V_{DDDR} = 2.0 V
Release signal set time	t_{SREL}	0			µs	
Oscillation stabilization time (Note 2)	t_{WAIT}		$2^{17}/f_{XX}$		s	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

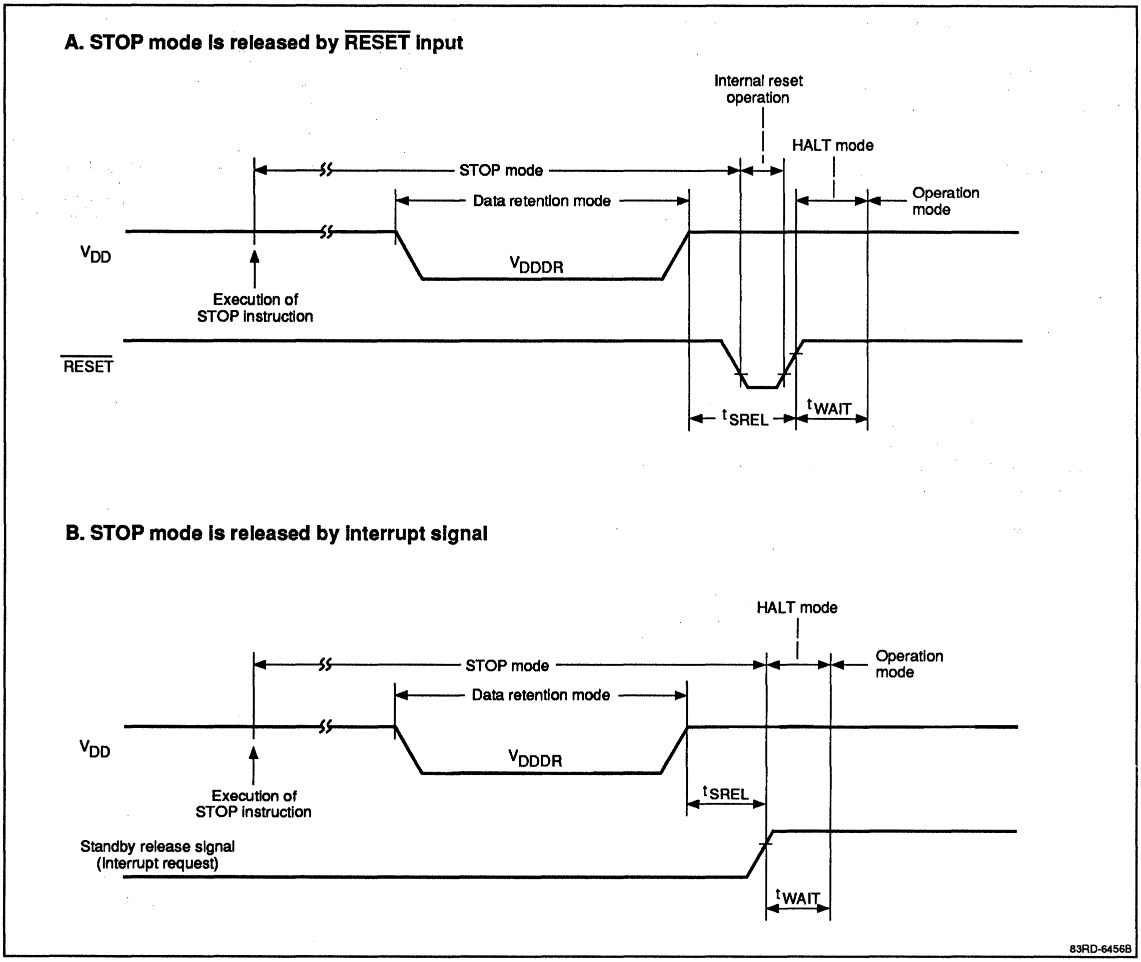
(1) Excludes current in the pull-up resistors, power-on-reset circuit, and comparator.

(2) Consult the manufacturer's resonator or crystal spec sheet for this value

(3) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillation is started. The interval timer can be used to delay the CPU from executing instructions using the the basic interval timer mode register (BTM) according to the following table: .

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{XX} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{XX}$ (Approx 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (Approx 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (Approx 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (Approx 1.95 ms)

Figure 19. Data Retention Timing


A. STOP mode is released by $\overline{\text{RESET}}$ input
Internal reset operation
HALT mode
STOP mode
Data retention mode
Operation mode
V_{DD}
V_{DDDR}
Execution of STOP instruction
$\overline{\text{RESET}}$
t_{SREL}
t_{WAIT}
B. STOP mode is released by interrupt signal
HALT mode
STOP mode
Operation mode
Data retention mode
V_{DD}
V_{DDDR}
Execution of STOP instruction
t_{SREL}
Standby release signal (interrupt request)
t_{WAIT}
83RD-6456B

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (μPD75P1xx)

T_A = 25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Operating voltage, V_{pp} (μPD75P108)	−0.3 to +22 V
Operating voltage, V_{pp} (μPD75P116)	−0.3 to +13.5 V
Input voltage, V_{I1} (other than ports 12-14)	−0.3 to V_{DD} + 0.3 V
Input voltage, V_{I2} (ports 12-14)	−0.3 to +13 V (Note 1)
Output voltage, V_O	−0.3 to V_{DD} + 0.3 V
High-level output current, I_{OH} (Single pin)	−15 mA
High-level output current, I_{OH} (Total of all pins)	−30 mA
Low-level output current, I_{OL} (Single pin)	30 mA pk
	15 mA rms (Note 2)
Low-level output current, I_{OL} (Total of ports 0, 2–4, 12–14)	100 mA pk
	36 mA rms (Note 2)
Low-level output current, I_{OL} (Total of ports 5–9)	100 mA pk
	36 mA rms (Note 2)
Operating temperature, t_{OPT}: μPD75P108	−10 to + 85°C
Operating temperature, t_{OPT}: μPD75P116	−40 to + 85°C
Storage temperature, t_{STG}	−65 to +125°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Notes:

(1) When applying more than 10 V to ports 12, 13, or 14, the external pull-up resistor must be at least 50 kΩ.

(2) rms value = pk x (duty cycle)½.

Capacitance (μPD75P1xx)

V_{DD} = 0 V; T_A = 25°C

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_{IN}	15	pF	f = 1 MHz; all unmeasured pins returned to ground
Output capacitance	C_{OUT}	15	pF	
I/O capacitance	C_{IO}	15	pF	

DC Characteristics (μPD75P1xx)

μPD75P108: T_A = −10 to +85°C; V_{DD} = 4.5 to 5.5 V
μPD75P116: T_A = −40 to +85°C; V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Except ports 0, 1, 12-14, TI0, TI1, X1, X2, and $\overline{\text{RESET}}$
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	Ports 0, 1, TI0, TI1 and $\overline{\text{RESET}}$
	V_{IH3}	0.7 V_{DD}		12	V	Ports 12-14; open drain
	V_{IH4}	V_{DD}−0.5		V_{DD}	V	X1, X2
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Except ports 0, 1, TI0, TI1, X1, X2, and $\overline{\text{RESET}}$
	V_{IL2}	0		0.2 V_{DD}	V	Ports 0, 1, TI0, TI1 and $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2
High-level output voltage	V_{OH}	V_{DD}−1.0			V	I_{OH} = −1 mA
Low-level output voltage	V_{OL}		0.55	2.0	V	Ports 0, 2-9; I_{OL}= 15 mA
			0.35	2.0	V	Ports 12-14; I_{OL}= 10 mA
				0.4	V	I_{OL} = 1.6 mA
High-level input leakage current	I_{LIH1}			3	μA	All except X1, X2, and ports 12-14; V_{IN} = V_{DD}
	ILIH2			20	μA	X1, X2; V_{IN} = V_{DD}
	ILIH3			20	μA	Ports 12-14; V_{IN} = 12 V
Low-level input leakage current	I_{LIL1}			−3	μA	All except X1, X2; V_{IN} = 0 V
	I_{LIL2}			−20	μA	X1, X2; V_{IN} = 0 V

DC Characteristics (μPD75P1xx) (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level output leakage current	I_{LOH1}			3	μA	All output pins except ports 12-14; $V_{OUT} = V_{DD}$
	I_{LOH2}			20	μA	Ports 12-14; V_{OUT} = 12 V
Low-level output leakage current	I_{LOL}			−3	μA	V_{OUT} = 0 V
Supply current (Note 1)	I_{DD1}		5	10	mA	V_{DD} = 5 V ± 10%; (Notes 2, 3)
	I_{DD2}		500	1500	μA	HALT Mode (Notes 2, 4); V_{DD} = 5V ±5%
	I_{DD3}		0.5	20	μA	μPD75P116: STOP Mode; V_{DD} = 5V ±5% (Note 5)
			30	100	μA	μPD75P108: STOP Mode ; V_{DD} = 5 V ±5% (Note 5)

Notes:

(1) Does not include comparator and includes current in the power-on-reset circuit.

(2) 4.19 MHz crystal oscillator; C1 = C2 = 22 pF.

(3) Value during high-speed operation and the processor control clock (PCC) is set to 0011.

(4) Value when the processor control clock (PCC) is set to 0100, and CPU is in HALT mode.

(5) I_{DD3} is less for the μPD75P116 because it does not contain the power-on-reset and power-on flag circuitry.

AC Characteristics (μPD75P1xx)

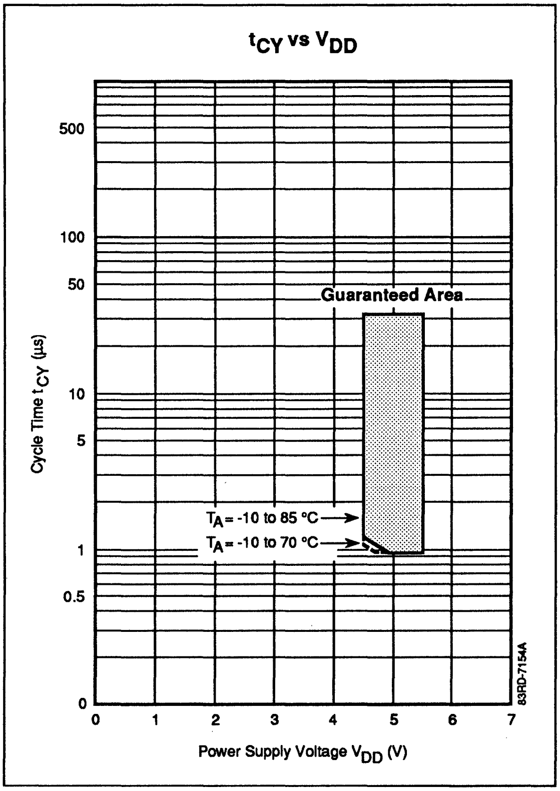
μPD75P108: T_A = −10 to +85°C; V_{DD} = 4.5 to 5.5 V
μPD75P116: T_A = −40 to +85°C; V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Note 1)	t_{CY}	0.95		32	μs	V_{DD} = 5 V ± 5%
		1.1		32	μs	V_{DD} = 5 V ± 10%
TI0, TI1 input frequency	f_{TI}	0		1	MHz	
TI0, TI1 input high-and low-level width	t_{TIH}, t_{TIL}	0.48			μs	
$\overline{SCK}$ cycle time	t_{KCY}	0.8			μs	Input
		0.95			μs	Output
$\overline{SCK}$ high-and low-level width	t_{KH}, t_{KL}	0.4			μs	Input
		0.5 t_{KCY}−50			ns	Output
SI vs. $\overline{SCK}$ ↑ setup time	t_{SIK}	100			ns	
SI vs. $\overline{SCK}$ ↑ hold time	t_{KSI}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO}			300	ns	
INT0-4 high- and low-level width	t_{INTH}, t_{INTL}	5			μs	
$\overline{RESET}$ low level width	t_{RSL}	5			μs	

Notes:

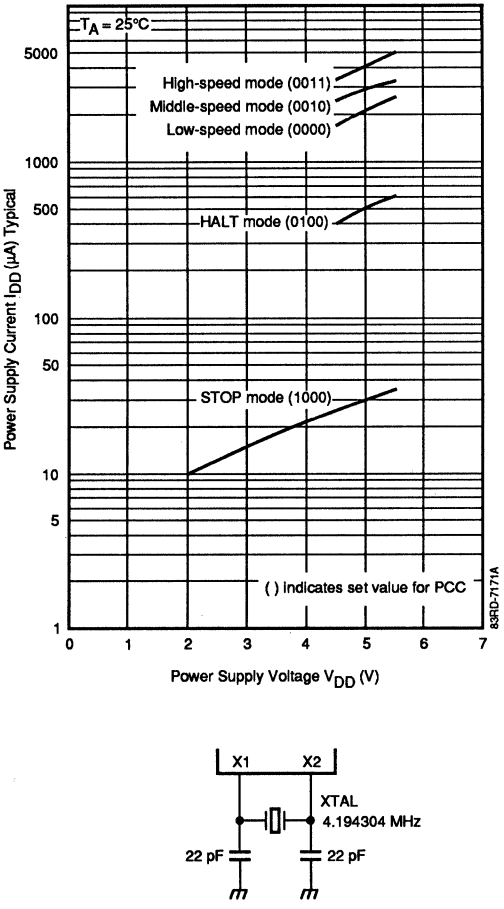
(1) Cycle time is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See figure 20.

Figure 20. Guaranteed Operating Range (μPD75P1xx)



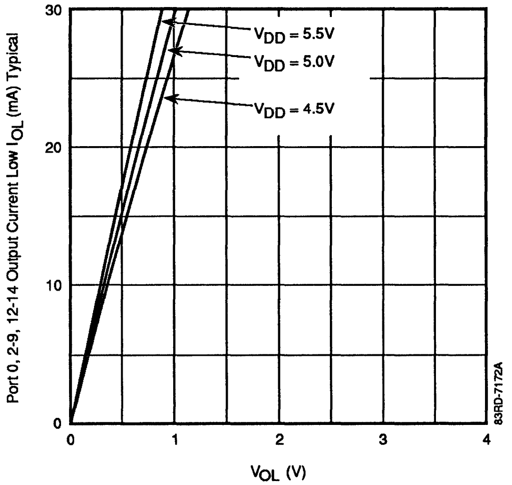
DC Characteristics (μPD75PIxx)

I_{DD} vs V_{DD} (4.19 MHz Crystal Resonator)

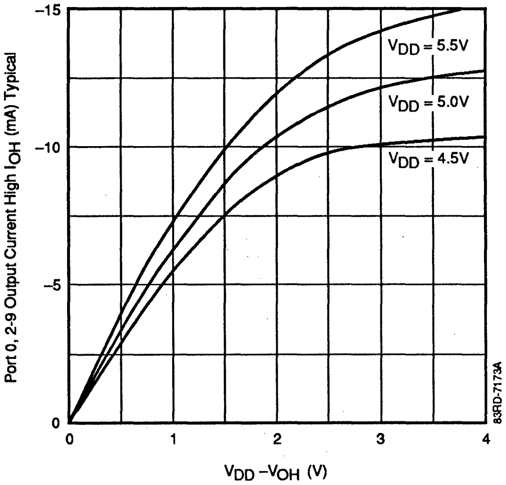


DC Characteristics (μPD75PIxx) (cont)

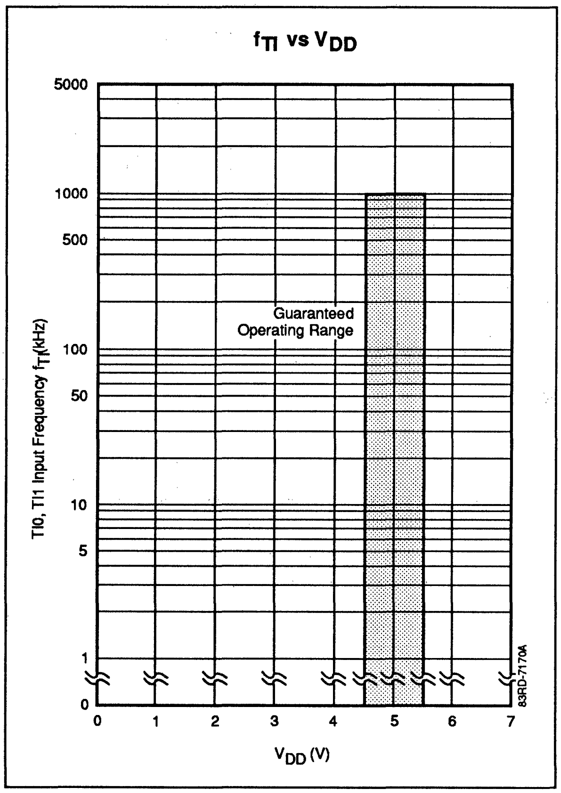
I_{OL} vs V_{OL} (Ports 0, 2-9, 12-14)



I_{OH} vs V_{DD}-V_{OH} (Ports 0, 2-9)



DC Characteristics (µPD75PIxx) (cont)



Data Memory STOP Mode Low Voltage Data Retention Characteristics (μPD75P1xx)

μPD75P108: T_A = −10 to +85°C; V_{DD} = 4.5 to 5.5 V
μPD75P116: T_A = −40 to +85°C; V_{DD} = 4.5 to 5.5 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		5.5	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	μA	μPD75P116; V_{DDDR} = 2.0 V (Note 4)
			15	40	μA	μPD75P108; V_{DDDR} = 2.0 V (Note 4)
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization time (Note 2)	t_{WAIT}		$2^{17}/f_{xx}$		s	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

(1) Includes current in the power-on-reset circuit, but excludes current in the comparator circuit.

(2) Consult the manufacturer's resonator or crystal specification for this value.

(3) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillation is started. The interval timer can be used to delay the CPU from executing instructions using the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{xx} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{xx}$ (Approx 250 ms)
–	0	1	1	$2^{17}/f_{xx}$ (Approx 31.3 ms)
–	1	0	1	$2^{15}/f_{xx}$ (Approx 7.82 ms)
–	1	1	1	$2^{13}/f_{xx}$ (Approx 1.95 ms)

(4) I_{DDDR} is less for the μPD75P116 because it does not contain the power-on-reset or power-on flag circuity

DC Programming Characteristics (μPD75P1xx)

μPD75P108: V_{DD} = 6.0 ± 0.25 V; V_{PP} = 21.0 ± 0.5 V; V_{SS} = 0 V; T_A = 25°C
μPD75P116: V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V; T_A = 25°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	$0.7V_{DD}$		V_{DD}	V	All except X1, X2
	V_{IH2}	$V_{DD}-0.5$		V_{DD}	V	X1, X2
Low-level input voltage	V_{IL1}	0		$0.3V_{DD}$	V	All except X1, X2
	V_{IL2}	0		0.4	V	X1, X2
Input leakage current	I_{LI}			10	μA	$V_{IN} = V_{IL}$ or V_{IH}
High-level output voltage	V_{OH}	$V_{DD}-1.0$			V	I_{OH} = −1 mA
Low-level output voltage	V_{OL}			0.4	V	I_{OL} = 1.6 mA
V_{DD} supply current	I_{DD}			30	mA	
V_{PP} supply current	I_{PP}			30	mA	MD0 = V_{IL}; MD1 = V_{IH}

Notes:

(1) V_{PP} must not exceed +22.0 V (μPD75P108) or +13.5 V (μPD75P116), including overshoot.

(2) V_{DD} must be applied before V_{PP}, and should be removed after V_{PP} is removed.

AC Programming Characteristics (μPD75P1xx)

μPD75P108: V_{DD} = 6.0 ± 0.25 V; V_{PP} = 21.0 ± 0.5 V; V_{SS} = 0 V; T_A = 25°C
μPD75P116: V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V; T_A = 25°C

Parameter	Symbol	(Note 1)	Min	Max	Unit	Conditions
Address setup time to MD0 ↓ (Note 2)	t_{AS}	t_{AS}	2		μs	
MD1 to MD0 ↓ setup	t_{M1S}	t_{OES}	2		μs	
Data to MD0 ↓ setup	t_{DS}	t_{DS}	2		μs	
Address hold from MD0 ↑ (Note 2)	t_{AH}	t_{AH}	2		μs	
Data hold from MD0 ↑	t_{DH}	t_{DH}	2		μs	
Data output float from MD0 ↑ delay	t_{DF}	t_{DF}	0	130	ns	
V_{PP} Setup to MD3 ↑	t_{VPS}	t_{VPS}	2		μs	
V_{DD} Setup to MD3 ↑	t_{VDS}	t_{VCS}	2		μs	
Initialized program pulse width	t_{PW}	t_{PW}	0.95	1.05	ms	
Additional program pulse width	t_{OPW}	t_{OPW}	0.95	21	ms	
MD0 setup to MD1 ↑	t_{MOS}	t_{CES}	2		μs	
Data output from MD0 ↓ delay	t_{DV}	t_{DV}		1	μs	MD0 = MD1 = V_{IL}
MD1 hold to MD0 ↑	t_{M1H}	t_{OEH}	2		μs	$t_{M1H} + t_{M1R} \geq 50$ μs
MD1 recovery from MD0 ↓	t_{M1R}	t_{OR}	2		μs	$t_{M1H} + t_{M1R} \geq 50$ μs
Program counter reset	t_{PCR}	–	10		μs	
X1 input high/low level width	t_{XH}, t_{XL}	–	0.125		μs	
X1 input frequency	f_{XX}	–		4.19	MHz	
Initial mode set	t_I	–	2		μs	
MD3 setup to MD1 ↑	t_{M3S}	–	2		μs	
MD3 hold to MD1 ↓	t_{M3H}	–	2		μs	
MD3 setup to MD0 ↓	t_{M3SR}	–	2		μs	During program read cycle
Address to data output delay time (Note 2)	t_{DAD}	t_{ACC}	2		μs	
Address to data output hold time (Note 2)	t_{HAD}	t_{OH}	0	130	ns	
MD3 output hold from MD0 ↑	t_{M3HR}	–	2		μs	
Data output from MD3 ↓ float delay time	t_{DFR}	–	2		μs	

Notes:

(1) These symbols correspond to these of the μPD27C256 EPROM.

(2) The internal address signal is incremented by one by the rising edge of the fourth X1 pulse.The address is not connected to an external pin.

Figure 21. EPROM Program Memory Write/Verify Timing

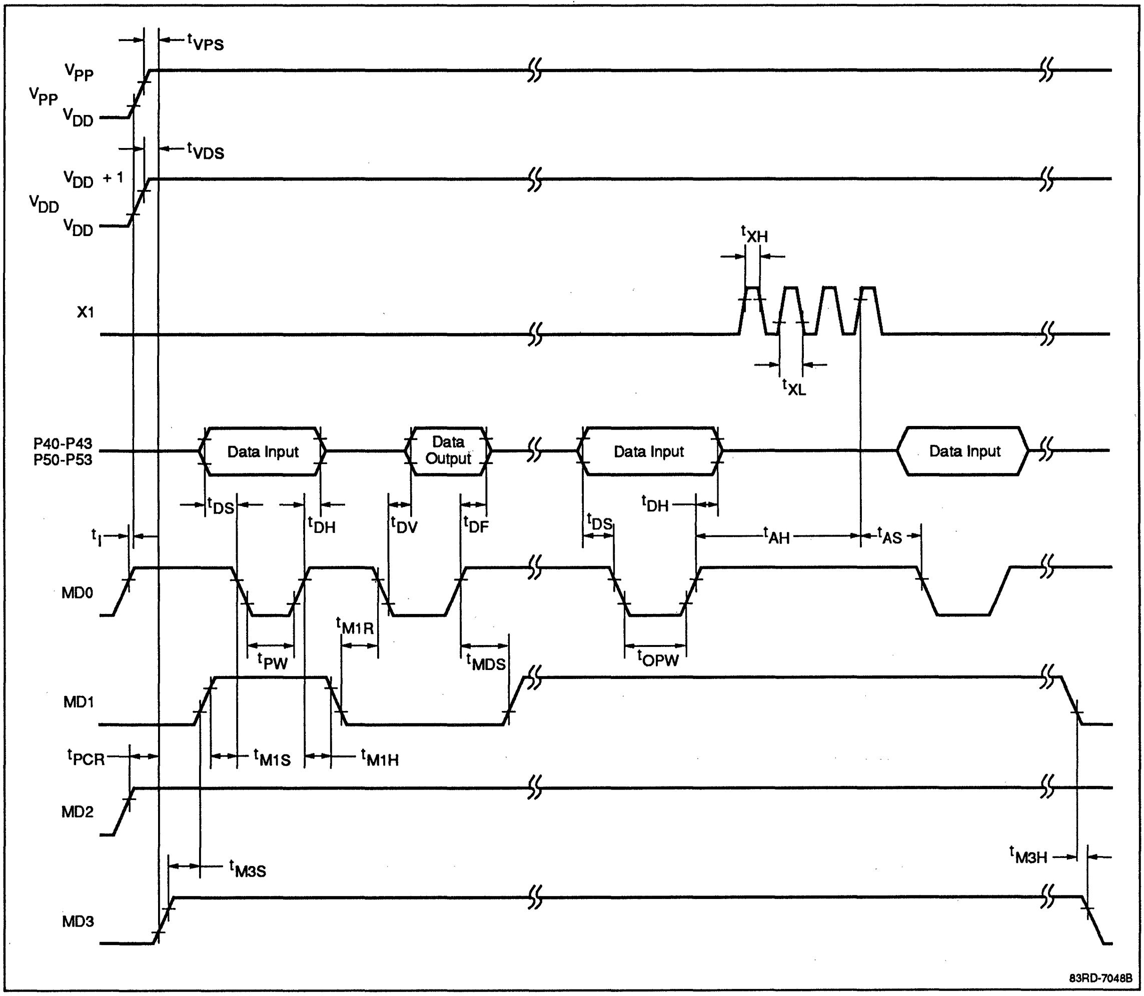
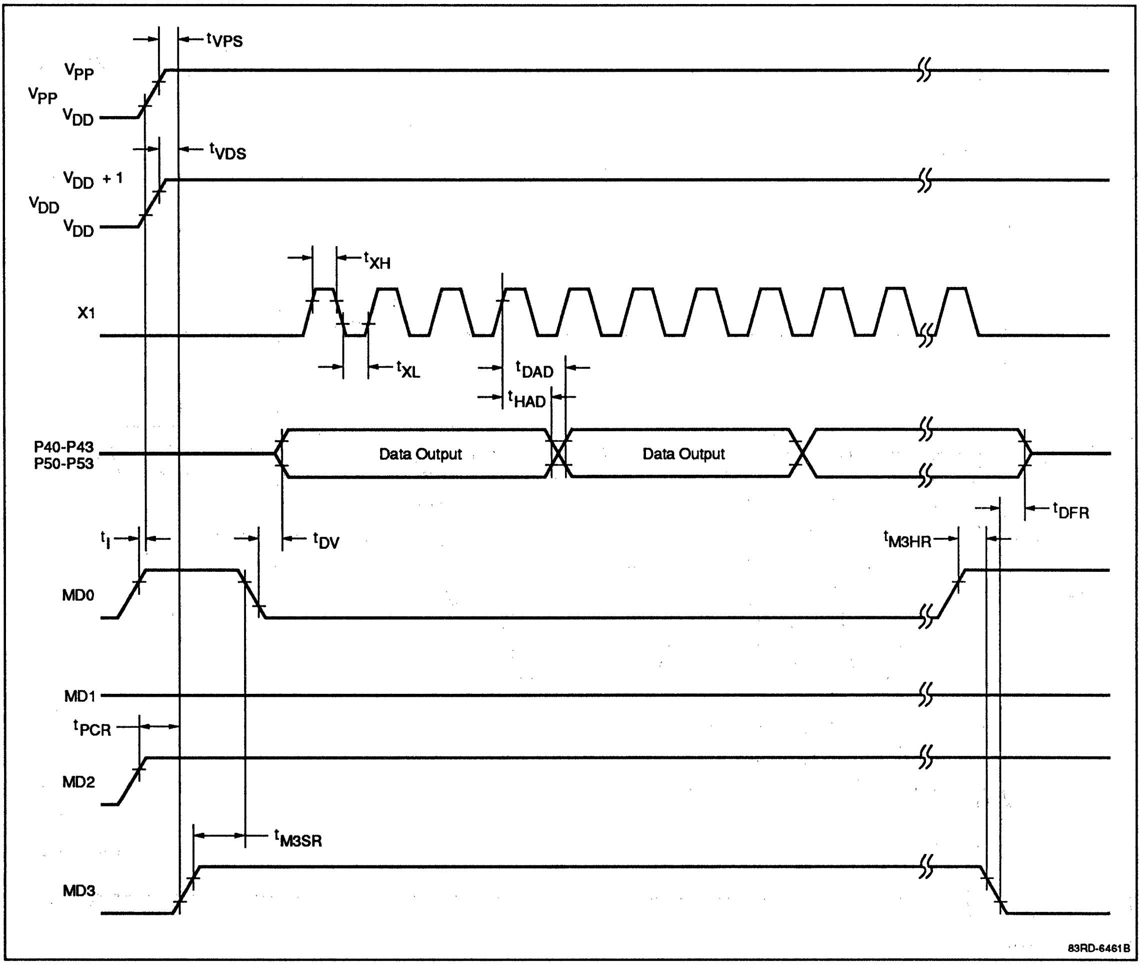


Figure 22. EPROM Program Memory Read Timing



μPD7520x/7521x/75CG2xx/75P216A 4-Bit Microcomputers With FIP (VF) Controller/Driver

Description

The μPD7520x/7521x is a family of single-chip CMOS microcomputers containing CPU, ROM, RAM, I/O ports, several timer/counters, vectored interrupts, a FIP® controller/driver, subsystem clock, and serial interface.

The instruction set allows the user to manipulate RAM data and I/O ports in 1-, 4-, and 8-bit units. The devices are ideally suited for controlling VCRs, microwave ovens, electronic stoves, washing machines, electronic cash registers, audio equipment, and meters.

Both EPROM and OTP versions are available. Refer to the ordering information.

Features

- □ 136 instructions
  - — Bit manipulation
  - — 4-bit and 8-bit transfer, arithmetic, logical comparison, and increment/decrement instructions
  - — 1-byte relative branch
  - — GETI instruction, to convert one 2-byte or 3-byte or two 1-byte instructions into a single 1-byte instruction
- □ Fast execution time (@ 4.19 MHz)
  - — High-speed cycle: 0.95 μs
  - — Lower-voltage cycles: 1.91 and 15.3 μs
- □ Program ROM
  - — μPD75206: 6016 bytes
  - — μPD75208/CG208A: 8064 bytes
  - — μPD75212A: 12160 bytes
  - — μPD75216A/CG216A/P216A: 16256 bytes
- □ Data memory (RAM)
  - — μPD75206: 369 x 4 bits
  - — μPD75208/CG208A: 497 x 4 bits
  - — μPD75212A/216A/CG216A/P216A: 512 x 4 bits
  - — Allows operation on 1, 4, or 8 bits
- □ Four banks of eight 4-bit registers
- □ Accumulators
  - — 1-bit (CY)
  - — 4-bit (A)
  - — 8-bit (XA)
- □ 28 port lines
  - — 20 I/O lines; 8 outputs directly drive LEDs (I_{sink} = 15 mA rms)
  - — 8 input-only lines
- □ One external event input
- □ Four timers
  - — 8-bit basic interval timer
  - — 8-bit timer/event counter
  - — 14-bit watch timer with buzzer output
  - — 14-bit PWM timer
- □ Programmable FIP controller/driver with memory area
  - — Up to 16 segments
  - — Up to 16 digits
  - — Eight dimming levels
  - — Key scan interrupt generation
- □ 8-bit serial interface
  - — Data transfer can be full duplex or receive only, and can be MSB or LSB first
- □ Vectored interrupts
  - — Two-level nesting
  - — Three external interrupts
  - — Five internal interrupts
  - — One input which generates an interrupt request
- □ Standby modes
  - — HALT mode: stops CPU only
  - — STOP mode: stops main system clock
- □ Operates with crystal or ceramic resonator
- □ CMOS operation with V_{DD} from 2.7 to 6.0 V
- □ Low current (V_{DD} = 5 V; f_{xx} = 4.19 MHz)
  - — Normal operation: 3.0 mA typical
  - — HALT mode: 0.6 mA typical
  - — STOP mode: 0.1 μA typical
- □ Mask options
  - — Power-on reset circuit and power-on flag (always in μPD75CG208A, μPD75CG216A, μPD75P216A)
  - — Port 6 input pull-down resistor
  - — FIP output pins have pull-down resistor
- □ Programmable versions
  - — Piggyback ROM: μPD75CG208A/CG216A
  - — OTP: μPD75P216A
- □ Available in 64-pin SDIP or QFP

FIP is a registered trademark of NEC Corporation

4

Ordering Information

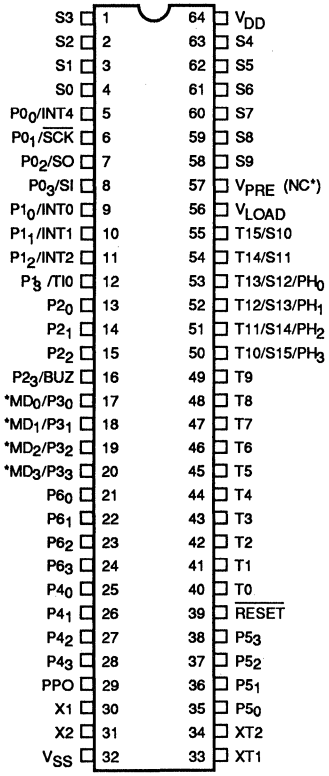
Part Number	Package Type	ROM Type
μPD75206CW-xxx	64-pin plastic SDIP	Mask ROM
μPD75206G-xxx-1B	64-pin plastic QFP (resin thickness 2.05 mm)	
μPD75206GF-xxx-3BE	64-pin plastic QFP (resin thickness 2.7 mm)	
μPD75208CW-xxx	64-pin plastic SDIP	Mask ROM
μPD75208G-xxx-1B	64-pin plastic QFP (resin thickness 2.05 mm)	
μPD75208GF-xxx-3BE	64-pin plastic QFP (resin thickness 2.7 mm)	
μPD75CG208E	64-pin ceramic SDIP	Piggyback EPROM
μPD75CG208EA	64-pin ceramic QFP	
μPD75212ACW-xxx	64-pin plastic SDIP	Mask ROM
μPD75212AGF-xxx-3BE	64-pin plastic QFP	
μPD75216ACW-xxx	64-pin plastic SDIP	
μPD75216AGF-xxx-3BE	64-pin plastic QFP	
μPD75CG216AE	64-pin ceramic SDIP	Piggyback EPROM
μPD75CG216AEA	64-pin ceramic QFP	
μPD75P216ACW	64-pin plastic SDIP	OTP

Notes:

(1) xxx indicates ROM code suffix

Pin Configurations

64-Pin Plastic SDIP

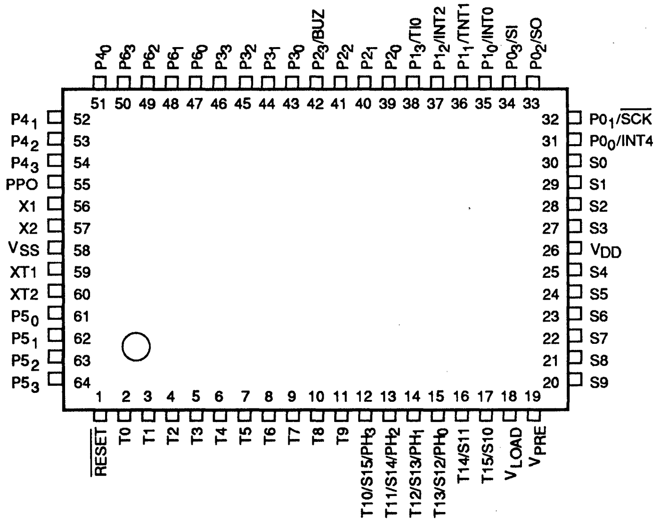


* Pin 57 is not connected on μPD75P216A.
MD$_0$ – MD$_3$ are used for programming the μPD75P216A.

83RD-6829A

Pin Configurations (cont)

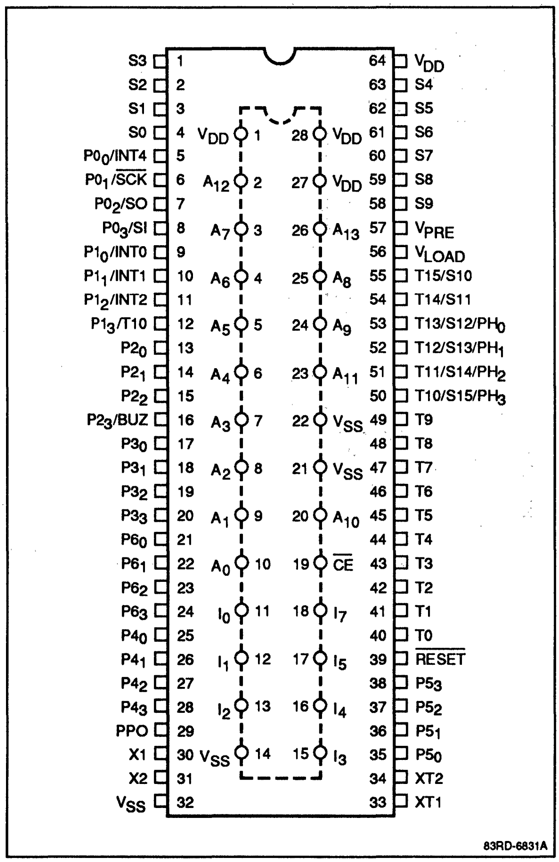
64-Pin Plastic QFP



83RD-6830B

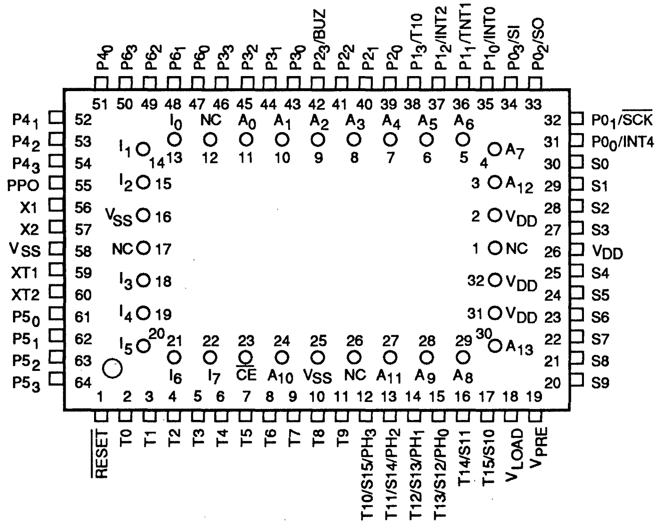
Pin Configurations (cont)

64-Pin Ceramic Piggyback SDIP



Pin Configurations (cont)

64-Pin Ceramic Piggyback QFP



83RD-6832B

Pin Identification

Symbol	Function
$P0_0$/INT4	Port 0 input; interrupt 4
$P0_1$/$\overline{SCK}$	Port 0 input; serial clock
$P0_2$/SO	Port 0 input; serial out
$P0_3$/SI	Port 0 input; serial in
$P1_0$/INT0	Port 1 input; interrupt 0
$P1_1$/INT1	Port 1 input; interrupt 1
$P1_2$/INT2	Port 1 input; interrupt 2
$P1_3$/TI0	Port 1 input; timer 0 input
$P2_0$-$P2_2$	Port 2 I/O
$P2_3$/BUZ	Port 2 I/O; buzzer output
$P3_0$-$P3_3$/MD0-MD3	Port 3 I/O; OTP operation mode (μPD75P216A)
$P4_0$-$P4_3$	Port 4 I/O
$P5_0$-$P5_3$	Port 5 I/O
$P6_0$-$P6_3$	Port 6 I/O
PH_0/T13/S12	Port H output; digit select line; segment line
PH_1/T12/S13	Port H output; digit select line; segment line
PH_2/T11/S14	Port H output; digit select line; segment line
PH_3/T10/S15	Port H output; digit select line; segment line
PPO	Pulse output
$\overline{RESET}$	Reset input
S0-S9	FIP segment outputs
T0-T9	FIP digit select outputs
T14/S11 T15/S10	Digit selects T14 and T15; segment lines S10 and S11
V_{DD}	Positive power supply
V_{LOAD}	FIP high-voltage negative supply voltage
V_{PRE}	FIP predriver negative supply voltage
V_{SS}	Ground
X1, X2	Main clock inputs
XT1, XT2	Subsystem clock inputs

PIN FUNCTIONS

$P0_0$-$P0_3$, INT4, $\overline{SCK}$, SO, SI (Port 0, Interrupt 4, Serial Clock, Serial In/Out)

These pins can be used as 4-bit input port 0. $P0_0$ can also be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. $P0_1$-$P0_3$ may also be used for the serial interface under the control of the SIOM register. SI is the serial input, SO is the serial output, and $\overline{SCK}$ is the serial clock. Reset causes these pins to default to the port 0 input mode.

$P1_0$-$P1_3$, INT0-INT2, TI0 (Port 1, Interrupts, Timer Input)

These pins can be used as 4-bit input port 1. $P1_0$ and $P1_1$ can also be used for edge-triggered interrupts INT0 and INT1. $P1_2$ can be used for INT2, which is also an edge-triggered input, but one which generates an interrupt request and does not cause an interrupt. $P1_3$ can be used as an input clock to the timer/event counter to count external events. Reset causes these pins to default to the port 1 input mode.

$P2_0$-$P2_3$, BUZ (Port 2, Buzzer Output)

These pins can be used as 4-bit I/O port 2. When used as an output the data is latched. When used as an input port the port outputs are three-state. $P2_3$ can also be used to output square waves for a buzzer. Reset causes these pins to default to the port 2 input mode.

$P3_0$-$P3_3$ (MD0-MD3) (Port 3)

These pins are used for input/output port 3. Each bit in this port can be independently programmed to be either an input or output. This port has latched outputs. MD0 through MD3 are used for the μPD75P216A OTP program memory write and verify mode to select the operation mode. A reset causes this port to be in the input mode.

$P4_0$-$P4_3$ (Port 4)

These pins are used for input/output port 4; this port has latched outputs. Port 4 outputs can directly drive an LED. Ports 4 and 5 can be paired together to function as one 8-bit port. A reset causes this port to be in the input mode.

$P5_0$-$P5_3$ (Port 5)

These pins are used for input/output port 5; this port has latched outputs and its outputs can directly drive an LED. Ports 4 and 5 can be paired together to function as one 8-bit port. A reset causes this port to be in the input mode.

$P6_0$-$P6_3$ (Port 6)

Port 6 is a 4-bit I/O port. Outputs are latched, and each bit can be independently programmed to be either an input or an output. Port 6 can have pull-down resistors added as a mask option. A reset signal causes this port to default to the input mode.

PH_0-PH_3, T10-T13, S12-S15 (Port H, Digit Select, Segment Lines)

Port H is a 4-bit output-only port, with P-channel open-drain outputs capable of directly driving LEDs. Pull-down resistors can be selected as a mask-option. Alternatively, these pins can be used as high voltage digit/segment outputs. A reset signal causes this port to default to the high-impedance state; if mask-option resistors are present the output goes low.

S0-S9 (Segment Lines)

These are high-voltage outputs used as FIP controller segment lines. Pull-down resistors can be selected as a mask-option. A reset signal sets these pins to the high-impedance state; if mask-option resistors are present the outputs go low.

T0-T9 (FIP Digit Select)

These are high-voltage outputs used as FIP controller digit select timing signals. Pull-down resistors can be selected as a mask-option. A reset signal sets these pins to the high-impedance state; if mask-option resistors are present the outputs go low.

T14/S11, T15/S10 (Digit Select/Segment Lines)

These two pins provide additional digit select or segment lines. When not used for the display they can be used as static outputs. Internal pull-down resistors are available as a mask option.

PPO (Timer/Pulse Generator Output)

This is an output signal from the timer/pulse generator, and can be either PWM (Pulse Width Modulated) or a square wave. This pin can also be used as a 1-bit output port. Pin assumes a high impedance state upon reset.

X1, X2 (System Clock Inputs)

These pins are the main system clock inputs. The clock can be either a ceramic or crystal resonator; an external logic signal may also be used as a clock source.

XT1, XT2 (Subsystem Clock Inputs)

These pins are the subsystem clock inputs. The clock can be either a ceramic or crystal resonator; an external logic signal may also be used as a clock source.

$\overline{\text{RESET}}$ (Reset)

This is the reset input, and it is active low.

V_{PRE} (Predriver Power)

This is the power supply for the predrivers of the FIP controller/driver.

V_{LOAD} (FIP Power Supply)

This pin is used to supply power to the output drivers for the segment lines and digit select pins of the FIP controller/driver.

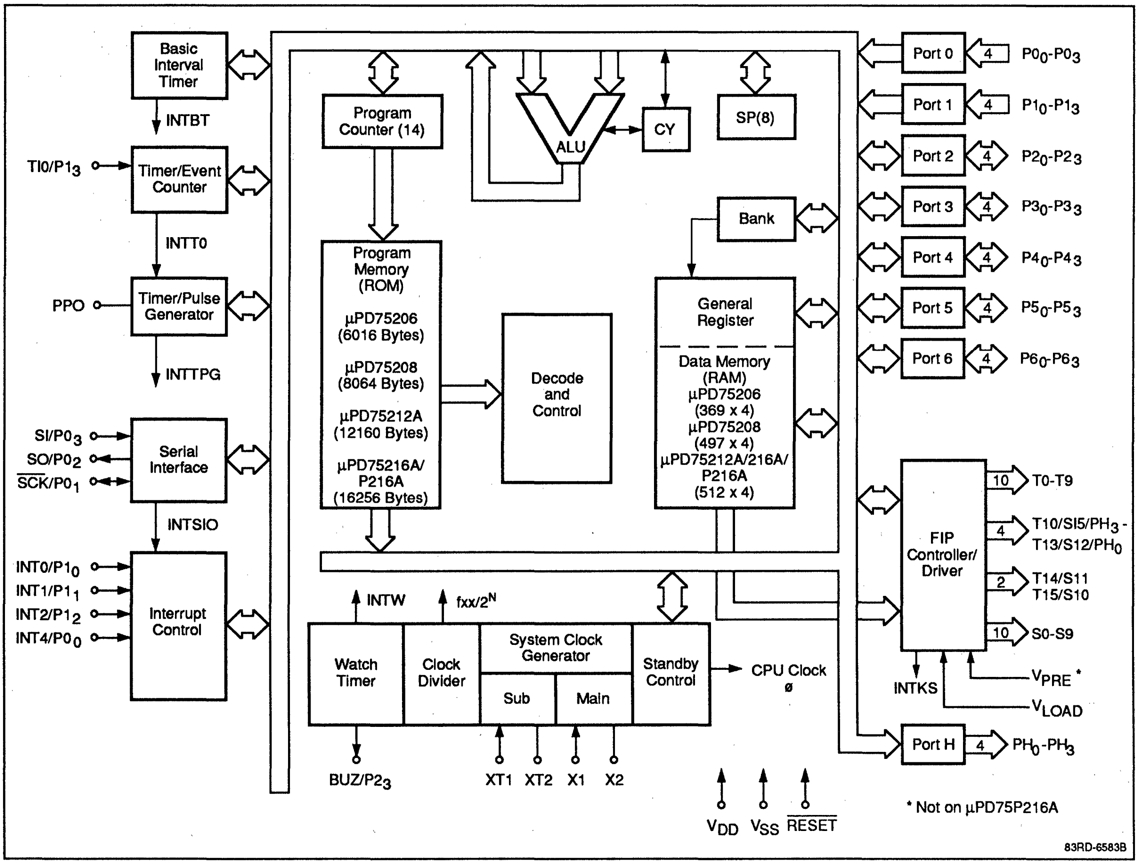
V_{DD} (Power Supply)

The system positive power supply pin.

V_{SS} (Ground)

System ground.

Block Diagram



Product Comparison

Item	µPD75CG208	µPD75CG216A	µPD75206	µPD75208	µPD75212A	µPD75216A	µPD75P216A
Program memory (ROM)	Piggyback EPROM 0000H–1FFFH 8192 x 8 bits	Piggyback EPROM 0000H–3FFFH 16384 x 8 bits	Mask ROM 0000H–177FH 6016 x 8 bits	Mask ROM 0000H–1F7FH 8064 x 8 bits	Mask ROM 0000H–2F7FH 12160 x 8 bits	Mask ROM 0000H–3F7FH 16256 x 8 bits	OTP 000H-3F7FH 16256 x 8 bits
Data memory (RAM)	497 x 4 bits	512 x 4 bits	369 x 4 bits	497 x 4 bits	512 x 4 bits	512 x 4 bits	512 x 4 bits
Port 6 pull-down resistor	None	None	Mask option (each bit)	Mask option (each bit)	Mask option (each bit)	Mask option (each bit)	None
S0-S8, T0-T9	On-chip pull-down resistor		Each bit can be mask programmed either for a pull-down resistor or as an open drain output				On-chip pull-down resistor
S9, T10-T15	Open drain	Open drain	Each bit can be mask programmed either for a pull-down resistor or as an open drain output				Open drain
Number of FIP segments	19 - 12	9 - 16	9 - 12	9 - 12	9 - 16	9 - 16	9-16
Power-on reset circuitry	On-chip	On-chip	Mask option	Mask option	Mask option	Mask option	None
Low-power data retention	Not provided	Not provided	2 volts	2 volts	2 volts	2 volts	Not provided
Operating voltage range	5 V ± 10%	5 V ± 10 %	2.7 to 6.0 V	2.7 to 6.0 V	2.7 to 6.0 V	2.7 to 6.0 V	5 V ±10%
Package	64-pin piggyback ceramic shrink DIP with window. 64-pin piggyback ceramic QFP with window.		64-pin plastic shrink DIP 64-pin plastic QFP				64-pin plastic shrink DIP

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (All Parts)

T_A = 25°C

Supply voltage, V_{DD}	–0.3 to +7.0 V
Supply voltage, V_{LOAD}	V_{DD}–40 to V_{DD}+0.3 V
Supply voltage, V_{PRE} (Note 1)	V_{DD}–12 to V_{DD}+0.3 V
Supply voltage, V_{PP} (Note 2)	–0.3 to +13.5 V
Input voltage, V_I	–0.3 to V_{DD} + 0.3 V
Output voltage, V_O (other than display)	–0.3 to V_{DD} + 0.3 V
Output voltage, V_{OD} (display pins)	V_{DD}–40 to V_{DD}+0.3 V
High-level output current, I_{OH} (single pin; other than display)	–15 mA
High-level output current, I_{OH} (single pin; S0-S9)	–15 mA
High-level output current, I_{OH} (single pin; T0-T15)	–30 mA
High-level output current, I_{OH} (total of all pins other than display)	–20 mA
High-level output current, I_{OH} (total of all display outputs)	–120 mA
Low-level output current, I_{OL} (single pin)	17 mA
Low-level output current, I_{OL} (total of all pins)	60 mA
Power dissipation, P_T (Plastic QFP)	450 mW
Power dissipation, P_T (Plastic SDIP)	600 mW
Storage temperature, t_{STG}	–65 to + 150°C
Operating temperature, t_{OPT} (Note 3)	–40 to +85°C
Operating temperature, t_{OPT} (Note 4)	–10 to +70°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Notes:

(1) Does not apply to μPD75P216A.

(2) For μPD75P216A only.

(3) For mask ROM parts.

(4) For μPD75CG208/CG216A/P216A.

Capacitance (All Parts)

V_{DD} = 0 V; T_A = 25°C

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_{IN}	15	pF	f = 1 MHz; all unmeasured pins returned to ground
Output capacitance; other than display	C_{OUT1}	15	pF	
Output capacitance; display only	C_{OUT2}	35	pF	
I/O capacitance	C_{IO}	15	pF	

Operating Supply Voltage

Mask ROM parts: T_A = –40 to +85°C
Programmable parts: T_A = –10 to +70°C

Parameter	Min	Max	Unit	Conditions
CPU (Note 2)	(Note 3)	6.0	V	(Note 4)
	4.5	5.5	V	μPD75CG208/CG216A and μPD75P216A only
Display controller	4.5	6.0	V	(Note 4)
	4.5	5.5	V	μPD75CG208/CG216A, μPD75P216A only
Timer/pulse generator	4.5	6.0	V	(Note 4)
	4.5	5.5	V	μPD75CG208/CG216A and μPD75P216A only
Other hardware (Note 2)	2.7	6.0	V	(Note 4)
	4.5	5.5	V	μPD75CG208/CG216A and μPD75P216A only

Notes:

(1) Care must be taken when designing the microcomputer that the total power dissipation does not exceed the maximum allowable. Power is dissipated in three areas:

a. At the CPU. PD is calculated by the product of V_{DD} (max) and I_{DD1} (max).

b. By the output pins. Total power dissipation is the sum of the values for each pin when maximum current is applied.

c. By the pull-down resistors.

(2) The CPU does not include the system clock oscillator, the display controller, or the timer/pulse generator.

(3) Varies according to the cycle time. See AC Characteristics.

(4) Mask ROM parts only.

Main System Clock Oscillator Characteristics

Mask ROM parts: $T_A = -40$ to $+85°C$; $V_{DD} = 2.7$ to $6.0\,V$
Programmable parts: $T_A = -10$ to $+70°C$; $V_{DD} = 4.5$ to $5.5\,V$

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 1A)	Oscillation frequency (Note 1)	f_{XX}	2.0		5.0	MHz	(Note 5)
	Oscillation stabilization time (Note 2)				4 (Note 3)	ms	After V_{DD} reaches the minimum oscillation voltage
Crystal resonator (Figure 1A)	Oscillation frequency (Note 1)	f_{XX}	2.0	4.19	5.0	MHz	(Note 5)
	Oscillation stabilization time (Note 2)				10 (Note 3)	ms	(Note 4)
					30 (Note 3)	ms	(Note 5)
External clock (Figure 1B)	X1 input frequency (Note 1)	f_{XX}	2.0		5.0	MHz	(Note 5)
	X1 input low- and high-level width	t_{XH}, t_{XL}	100		250	ns	

Notes:

(1) The oscillation frequency and X1 input frequency are included only to show the characteristics of the oscillators. Refer to the AC Characteristics table for actual instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillator to stabilize after voltage V_{DD} is applied and reaches the V_{DD} spec or the STOP mode is released.

(3) Values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's specification sheets.

(4) $V_{DD} = 4.5$ to $6.0\,V$ for mask ROM parts and $V_{DD} = 4.5$ to $5.5\,V$ for programmable parts.

(5) $V_{DD} = 2.7$ to $6.0\,V$ for mask ROM parts and $V_{DD} = 4.5$ to $5.5\,V$ for programmable parts.

Subsystem Clock Oscillator Characteristics

Mask ROM parts: $T_A = -40$ to $+85°C$; $V_{DD} = 2.7$ to $6.0\,V$
Programmable parts: $T_A = -10$ to $+70°C$; $V_{DD} = 4.5$ to $5.5\,V$

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 2A)	Oscillation frequency	f_{XT}	32	32.768	35	kHz	(Note 1)
	Oscillation stabilization time (Note 2)			1.0	2	s	(Note 3)
					10	s	(Note 4)
External clock (Figure 2B)	XT1 input frequency	f_{XT}	32		100	kHz	(Note 4)
	XT1 input high/low level width	t_{XTH}, t_{XTL}	10		32	μs	

Notes:

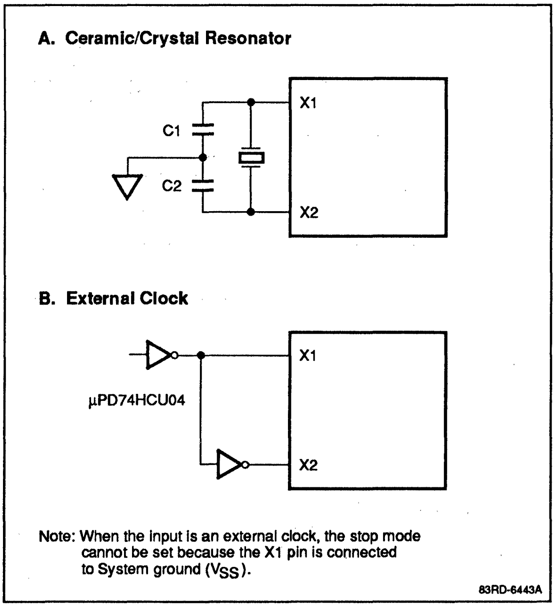
(1) The oscillator frequency and input frequency indicates only the oscillator characteristics. Refer to the AC Characteristics for the instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillation to stabilize after V_{DD} is applied and reaches the V_{DD} spec or after STOP mode is released.

(3) $V_{DD} = 4.5$ to $6.0\,V$ for mask ROM parts and 4.5 to $5.5\,V$ for programmable parts.

(4) $V_{DD} = 2.7$ to $6.0\,V$ for mask ROM parts and 4.5 to $5.5\,V$ for programmable parts.

Figure 1. Main System Clock Configurations



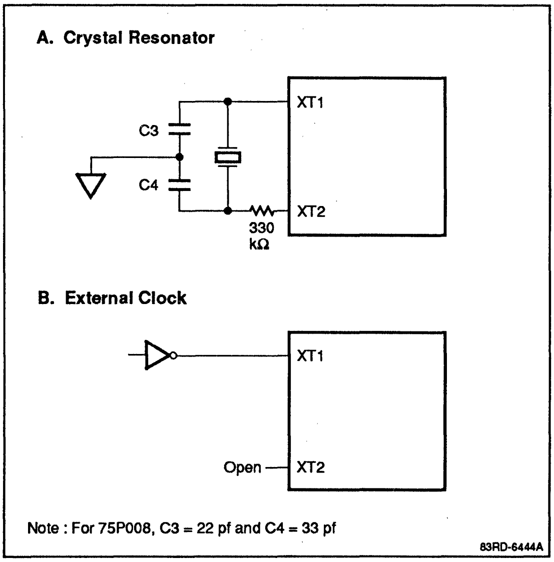
A. Ceramic/Crystal Resonator

B. External Clock

Note: When the input is an external clock, the stop mode cannot be set because the X1 pin is connected to System ground (V_{SS}).

83RD-6443A

Figure 2. Subsystem Clock Configurations



A. Crystal Resonator

B. External Clock

Note : For 75P008, C3 = 22 pf and C4 = 33 pf

83RD-6444A

Recommended Main System Clock Oscillator Circuit Constants (Mask ROM Parts and μPD75CG216A)

Main system clock = Ceramic; T_A = −40 to +85°C (for mask ROM parts) and −10 to +70°C (for μPD75CG216A)

Manufacturer	Product name (Note 1)	C1 (pF)	C2 (pF)	Remarks
Murata	CSA 2.00MG	30	30	
	CSA 4.19MG	30	30	
	CSA 4.91MG	30	30	
	CAT 2.00MG	None	None	C on-chip type
	CST 4.19MG	None	None	C on-chip type
	CST 4.91MG	None	None	C on-chip type
Kyocera	KBR-2.0 MS	47	47	For mask ROM parts only
	KBR-4.0MS	33	33	
	KBR-4.19MS	33	33	
	KBR-4.91MS	33	33	
TDK	FCR-3.58M2	30	30	
	FCR-4.00M2	30	30	
	FCR-4.19M2	30	30	
	FCR-4.19MC	None	None	C on-chip type

Notes:

(1) Oscillation voltage range = 4.0 to 6.0 V for mask ROM parts and V_{DD} = 4.5 to 5.5 V for μPD75CG216A

Recommended Main System Clock Oscillator Circuit Constants (Mask ROM Parts and μPD75CG216A)

Main system clock = Crystal; T_A = −40 to +85°C (for mask ROM parts) and −10 to +70°C (for μPD75CG216A)

Manufacturer	Frequency (MHz)	Retainer	Load Capacitance C_L (pF)	C1 (pF)	C2 (pF)	Oscillator Voltage Range	
						Min (V)	Max (V)
Kinseki	2.00	HC-18/U	16	20	20	4.5	(Note 1)
	4.19	HC-49/U	16	20	20	4.5	(Note 1)
	4.91	HC-43/U	16	20	20	4.5	(Note 1)

Notes:

(1) Oscillation voltage range max = 6.0 V for mask ROM parts and 5.5 V for μPD75CG216A.

Recommended Subsystem Clock Oscillator Circuit Constants (Mask ROM Parts and μPD75CG216A)

Subsystem clock = Crystal; T_A = −10 to +60°C (for the mask ROM parts) and −10 to +70°C (for the μPD75CG216A)

Manufacturer	Type	Load Capacitance C_L (pF)	C3 (pF)	C4 (pF)	R (k Ω)	Oscillator Voltage Range	
						Min (V)	Max (V)
Kinseki	P-3	12	22	22	330	(Note 1)	(Note 1)
Citizen	CFS-308	14	22	33	330	(Note 1)	(Note 1)

Notes:

(1) Oscillation voltage range is 2.7 to 6.0 V for the mask ROM parts and 4.5 to 5.5 V for the μPD75CG216A.

Recommended Main System Clock Ceramic Resonators (μPD75CG208)

		External Capacitors		V_{DD} Range	
Manufacturer	**Product name**	**C1 (pF)**	**C2 (pF)**	**Min (V)**	**Max (V)**
Murata	CSA 4.19 MG	30	60	4.5	5.5
Kyocera	KBR-2.09 MS	68	68	4.5	5.5
	KBR-3.58 MS	33	33	4.5	5.5
	KBR-4.19 MS	33	33	4.5	5.5
	KBR-4.9 M	33	33	4.5	5.5

Recommended Main System Clock Crystal Resonators (μPD75CG208)

		External Capacitors		V_{DD} Range	
Manufacturer (Note 1)	**Product name**	**C1 (pF) (Note 2)**	**C2 (pF)**	**Min (V)**	**Max (V)**
Kinseki	HC-49/U	15	15	4.5	5.5

Notes:

(1) Equivalent series resistance of a crystal must be lower than 80 Ω.

(2) Variable range of C1 for frequency trimming should be 10 to 33 (pF).

Power-on Reset Characteristics (Note 1)

T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 (Mask ROM parts); T_A = −10 to +70°C, V_{DD} = 5 V ±10% (μPD75CG208 and μPD75CG216A)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
POR high-level operating voltage	V_{DDH}	4.5		6.0	V	
POR low-level operating voltage	V_{DDL}	0		0.2	V	
Supply voltage rise time	t_r	10		(Note 2)	μs	
Supply voltage OFF time	t_{off}	1			s	
POR circuit current dissipation (Note 3)	I_{DDPR}		10	100	μA	V_{DD} = 5 V ± 10%; (μPD752xx only)
			10	200	μA	V_{DD} = 5 V ±10% (μPD75CG208/CG216A only)
			2	20	μA	V_{DD} = 2.7 V (μPD752xx only)

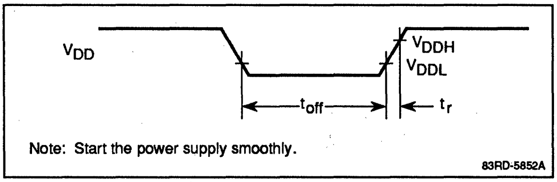
Notes:

(1) This circuit is present on the μPD75CG208 and μPD75CG216. It is a mask option on mask ROM parts and is not available on the μPD75P216A.

(2) $2^{17}/f_{XX}$ (31.3 ms at f_{XX} = 4.19 MHz).

(3) Current which flows when the internal reset circuit and power-on flag are used.

Figure 3. Power-on Reset Timing



Note: Start the power supply smoothly.

DC Characteristics

T_A = −40 to +85°C, V_{DD} = 2.7 to 6.0 V (Mask ROM parts); T_A = −10 to +70°C; V_{DD} = 4.5 to 5.5 V (Programmable Parts)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	All except ports 0, 1, 6; X1, X2, XT1, $\overline{\text{RESET}}$
	V_{IH2}	0.75 V_{DD}		V_{DD}	V	Ports 0 and 1; $\overline{\text{RESET}}$
	V_{IH3}	V_{DD}−0.4		V_{DD}	V	X1, X2, XT1
	V_{IH4}	0.65 V_{DD}		V_{DD}	V	Port 6; V_{DD} = 4.5 to 6.0 V (μPD752xx only); 5 V ±10% (programmable parts)
		0.7 V_{DD}		V_{DD}	V	Port 6; V_{DD} = 2.7 to 6.0 V (μPD752xx only)
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	All except ports 0, 1, 6; X1, X2, XT1, $\overline{\text{RESET}}$
	V_{IL2}	0		0.2 V_{DD}	V	Ports 0, 1, and 6; $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2, XT1
High-level output voltage	V_{OH}	V_{DD}−1.0			V	All outputs; I_{OH} = −1 mA (Note 10)
		V_{DD}−0.5			V	All outputs; I_{OH} = −100 μA (Note 11)
Low-level current	I_{IL}		−300	−800	μA	I_0-I_7; V_{IN} = 0 V; (μPD75CG208/G216A only)
Low-level output voltage	V_{OL}		0.4	2.0	V	Ports 4 and 5; I_{OL}= 15 mA (Note 10)
				0.4	V	All output pins; I_{OL}= 1.6 mA (Note 10)
				0.5	V	All output pins; I_{OL}= 400 μA (μPD752xx only)
High-level input leakage current	I_{LIH1}			3	μA	All except X1, X2, and XT1; V_{IN} = V_{DD}
	I_{LIH2}			20	μA	X1, X2, and XT1; V_{IN} = V_{DD}
Low-level input leakage current	I_{LIL1}			−3	μA	All except X1, X2, and XT1; V_{IN} = 0 V
	I_{LIL2}			−20	μA	X1, X2, and XT1; V_{IN} = 0 V
High-level output leakage current	I_{LOH}			3	μA	All output pins; V_{OUT} = V_{DD}
Low-level output leakage current	I_{LOL1}			−3	μA	All except display output pins; V_{OUT} = 0 V
	I_{LOL2}			−10	μA	Display output pins; V_{OUT} = V_{LOAD} = V_{DD}−35 V
Display output current	I_{OD}	−3	−5.5		mA	S0-S9 (Note 1) see Recommended External Circuit
		−3	−5.5		mA	S0-S9; V_{DD} = 4.5 to 6.0 V; V_{OD} =V_{DD} −2 V (μPD75P216A only)
		−1.5	−3.5		mA	S0-S9; All except μPD75P216A (Note 2)
		−15	−22		mA	T0-T15 (Note 1) see Recommended External Circuit
		−15	−22		mA	T0-T15; V_{DD} = 4.5 to 6.0 V; V_{OD} =V_{DD} −2 V (μPD75P216A only)
		−7	−15		mA	T0-T15; All except μPD75P216A (Note 2)
Internal pull-down resistor (mask option)	R_{P6}	30	80	200	kΩ	Port 6; V_{DD} = 4.5 to 6.0 V (μPD75206/208)
		20	80	200	kΩ	Port 6; V_{DD} = 4.5 to 6.0 V (μPD75212A/216A only); V_{IN} = V_{DD}
		30		1000	kΩ	Port 6; V_{DD} = 2.7 to 6.0 V (μPD75206/208)
		20		1000	kΩ	Port 6; V_{DD} = 2.7 to 6.0 V (μPD75212A/216A only); V_{IN} = V_{DD}
	R_L	25	70	135	kΩ	Display output pins; V_{DD}−V_{LOAD} = 35 V (μPD75212A/216A/P216A)
		40	70	120	kΩ	Display output pins; V_{DD}−V_{LOAD} = 35 V (μPD75206/208/CG208/CG216A)

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Supply current (Note 6)	I_{DD1} (Note 3)		3.0	9.0	mA	V_{DD} = 5 V ± 10% (Note 4)
			0.55	1.5	mA	V_{DD} = 3 V ± 10% (Note 5; µPD752xx only)
	I_{DD2} (Note 3)		600	1800	µA	HALT mode; V_{DD} = 5 V ± 10% (Note 12)
			200	600	µA	HALT mode; V_{DD} = 3 V ± 10% (µPD752xx only)
	I_{DD3}		40	120	µA	V_{DD} = 3 V ± 10% (Notes 7, 8; µPD752xx only)
			100	300	µA	(Note 7; µPD75CG208/CG216A and µPD75P216A only)
	I_{DD4}		5	15	µA	HALT mode; V_{DD} = 3 V ± 10% (Notes 7, 8; µPD752xx only)
			40	100	µA	HALT mode (Notes 7, 8; µPD75CG208/CG216A and µPD75P216A only)
	I_{DD5}		0.5	20	µA	STOP mode; XT1 = 0 V; V_{DD} = 5 V ± 10 %; µPD752xx only (Note 6)
			0.1	10	µA	STOP mode; XT1 = 0 V; V_{DD} = 3 V ± 10 %; µPD752xx only (Note 6)
			10	200	µA	STOP mode; XT1 = 0 V; (Note 9; µPD75CG208 and µPD75CG216A only)
			0.5	200	µA	STOP mode; XT1 = 0 V (µPD75P216A only)

Notes:

(1) V_{DD} = 4.5 to 6.0 V for mask ROM parts and V_{DD} = 4.5 to 5.5 V for programmable parts; V_{OD} = V_{DD} −2 V; V_{PRE} = V_{DD} −9±1 V.

(2) V_{DD} = 4.5 to 6.0 V for mask ROM parts and V_{DD} = 4.5 to 5.5 V for programmable parts; V_{OD} = V_{DD} −2 V; V_{PRE} = 0 V.

(3) 4.19 MHz crystal oscillator; C1 = C2 = 15 pF.

(4) Value during high-speed operation and the processor control clock (PCC) is set to 0011.

(5) Value during low-speed operation and the processor control clock (PCC) is set to 0000.

(6) Does not include pull-down resistor current for S0-S8 and T0-T9. In the mask ROM parts, the current for the power-on reset circuit (mask option) is not included. In the µPD75CG208/CG216A, the current for the piggyback EPROM and the current in the on-chip pull-up resistors for I0-I7 is not included.

(7) 32 kHz crystal oscillator.

(8) Value when the system clock control register (SCC) is set to 1001, generation of the main system clock pulse is stopped, and the CPU is operated by the subsystem clock pulse.

(9) With the $\overline{CE}$ or $\overline{OE}$ of the piggybacked EPROM set high.

(10) V_{DD} = 4.5 to 6.0 V for mask ROM parts and V_{DD} = 4.5 to 5.5 V for programmable parts.

(11) V_{DD} = 2.7 to 6.0 V for mask ROM parts and V_{DD} = 4.5 to 5.5 V for programmable parts.

(12) For the µPD75CG208/CG216A, the $\overline{CE}$ or $\overline{OE}$ pin of the piggyback EPROM is set to a high level.

(13) No subsystem clock.

AC Characteristics

Mask ROM parts: T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V
Programmable parts: T_A = −10 to +70°C; V_{DD} = 5 V ±10%

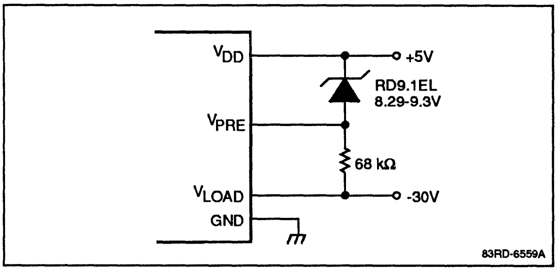
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time: minimum instruction execution time (Note 1)	t_{CY}	0.95		32	μs	Main system clock; V_{DD} = 4.5 to V_{DD} max
		3.8		32	μs	Main system clock; V_{DD} = 2.7 to 6.0 V; (μPD752xx only)
		114	122	125	μs	Subsystem clock
TI0 input frequency	f_{TI}	0		0.6	MHz	V_{DD} = 4.5 to V_{DD} max (μPD752xx/P216A)
		0		165	kHz	V_{DD} = 2.7 to 6.0 V (μPD752xx only)
		0		1	MHz	(μPD75CG208/CG216A only)
TI0 input low- and high-level width	t_{IL}, t_{IH}	0.83			μs	V_{DD} = 4.5 to V_{DD} max (μPD752xx and μPD75P216A only)
		3			μs	V_{DD} = 2.7 to 6.0 V (μPD752xx only)
		0.48			μs	(μPD75CG208/CG216A only)
$\overline{SCK}$ cycle time	t_{KCY}	0.8			μs	Input; V_{DD} = 4.5 to V_{DD} max
		0.95			μs	Output; V_{DD} = 4.5 to V_{DD} max
		3.2			μs	Input; V_{DD} = 2.7 to 6.0 V (μPD752xx only)
		3.8			μs	Output; V_{DD} = 2.7 to 6.0 V (μPD752xx only)
$\overline{SCK}$ low- and high-level width	t_{KL}, t_{KH}	0.4			μs	Input; V_{DD} = 4.5 to V_{DD} max
		0.5 t_{KCY} − 50			ns	Output; V_{DD} = 4.5 to V_{DD} max
		1.6			μs	Input; V_{DD} = 2.7 to 6.0 V (μPD752xx only)
		0.5 t_{KCY} − 150			ns	Output; V_{DD} = 2.7 V to 6.0 V (μPD752xx only)
SI vs. $\overline{SCK}$ ↑ setup time	t_{SIK}	100			ns	
SI vs. $\overline{SCK}$ ↑ hold time	t_{KSI}	400			ns	
$\overline{SCK}$ ↓ → SO output delay time	t_{KSO}			300	ns	V_{DD} = 4.5 to V_{DD} max
				1000	ns	V_{DD} = 2.7 to 6.0 V (μPD752xx only)
Interrupt inputs low- and high-level width	t_{INTL}, t_{INTH}	(Note 2)			μs	INT0
		2 t_{CY}			μs	INT1
		10			μs	INT2, INT4
$\overline{RESET}$ low-level width	t_{RSL}	10			μs	

Notes:

(1) Cycle time is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), V_{DD}, and the processor clock control (PCC). See the graph depicting the supply voltage vs. The cycle time when the microcomputer is operating on the main system clock.

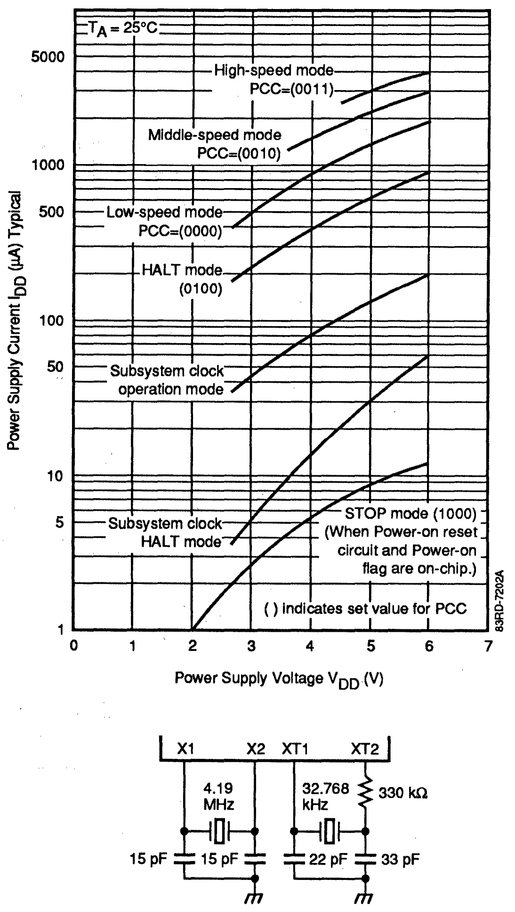
(2) 2t_{CY} or 128/f_{XX}, depending on the setting of the interrupt mode register (IM0).

Recommended External Circuit

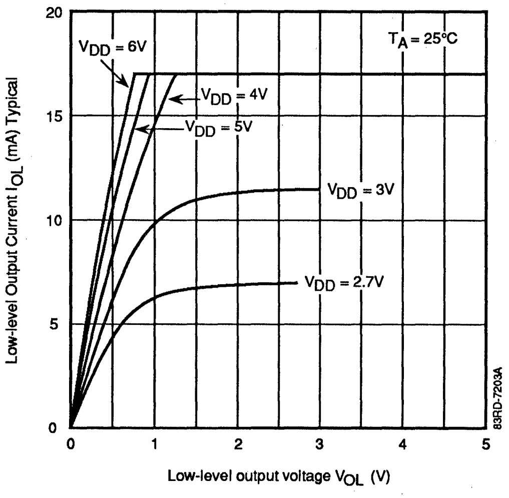


DC Characteristics (µPD752xx only)

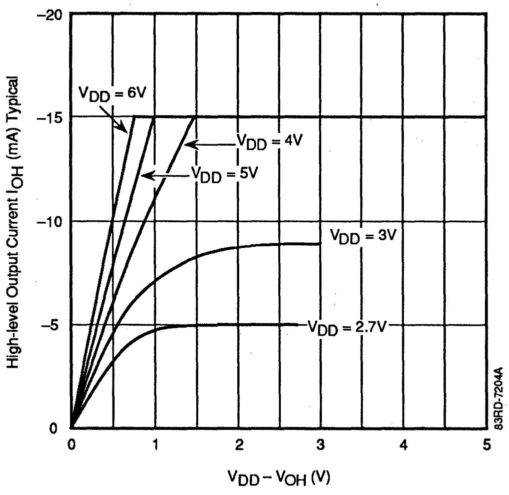
I_{DD} vs V_{DD}



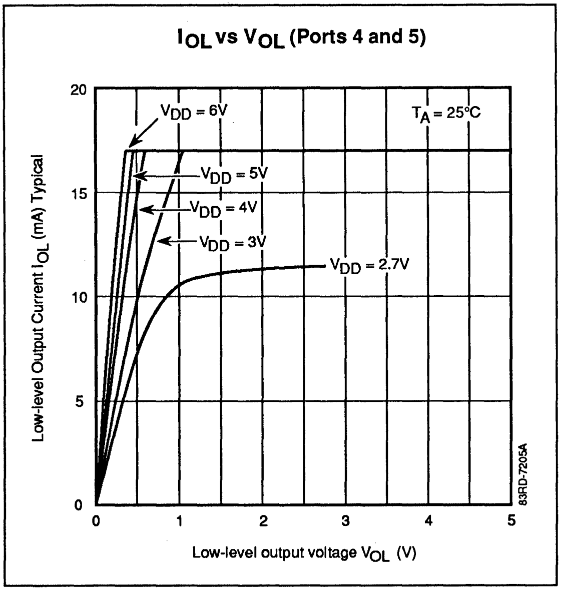
I_{OL} vs V_{OL} (Ports 0, 2, 3, 6)



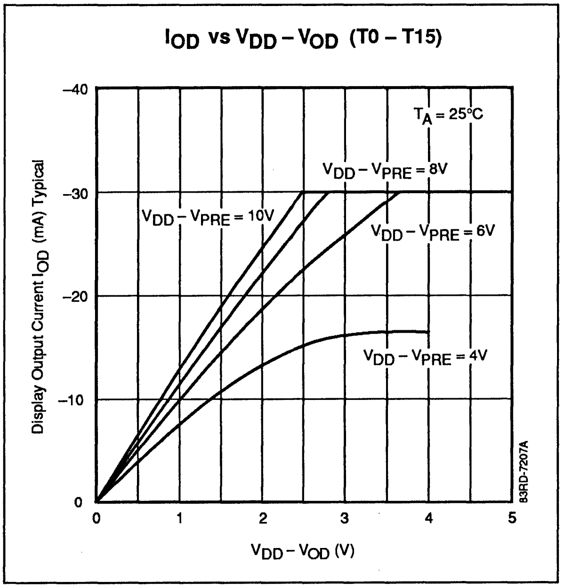
I_{OH} vs V_{DD} − V_{OH} (Ports 0, 2, 3, 6)



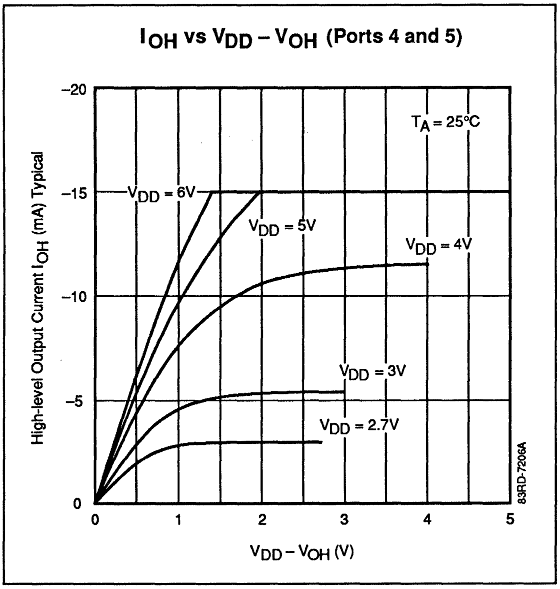
DC Characteristics (μPD752xx only) (cont)



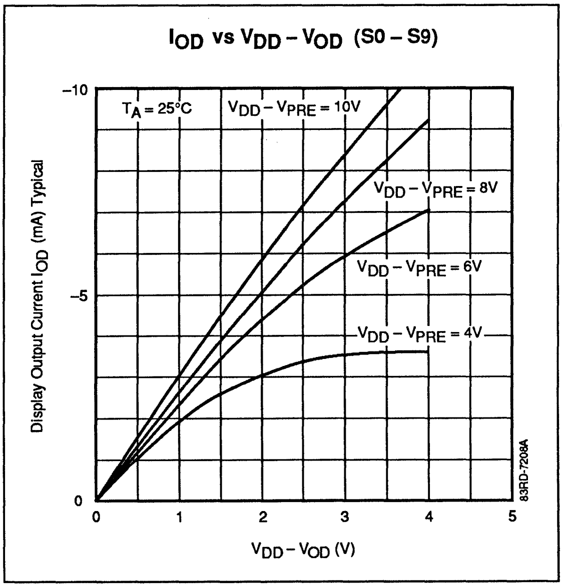
I_{OL} vs V_{OL} (Ports 4 and 5)



I_{OD} vs $V_{DD} - V_{OD}$ (T0 – T15)



I_{OH} vs $V_{DD} - V_{OH}$ (Ports 4 and 5)



I_{OD} vs $V_{DD} - V_{OD}$ (S0 – S9)

Figure 4. Guaranteed Operating Range

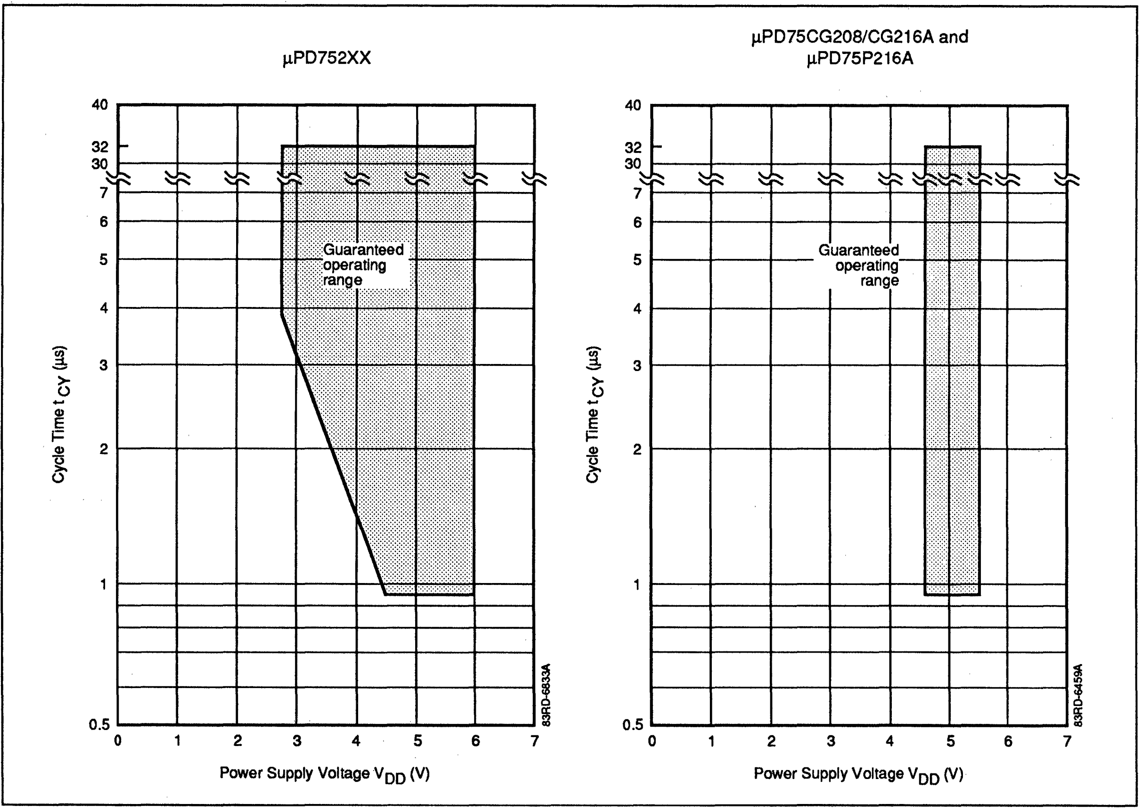


Figure 5. AC Timing Measurement Points

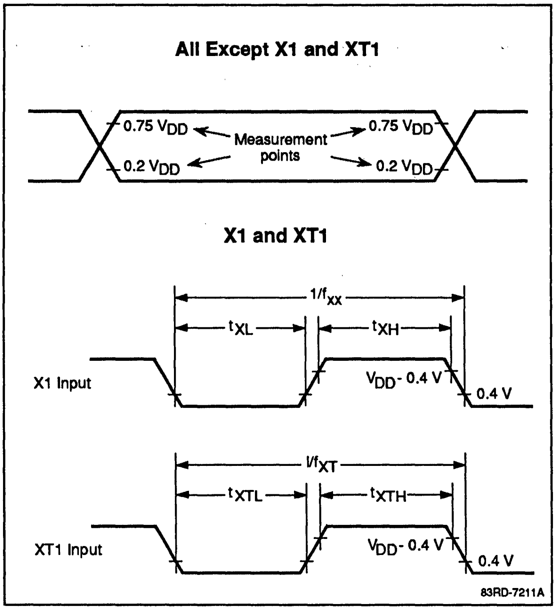


Figure 6. TI0 Timing

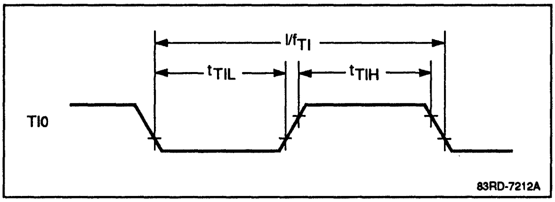


Figure 7. Serial Transfer Timing

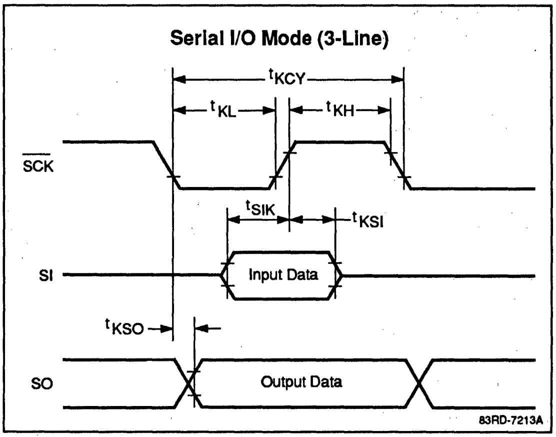


Figure 8. Interrupt Input Timing

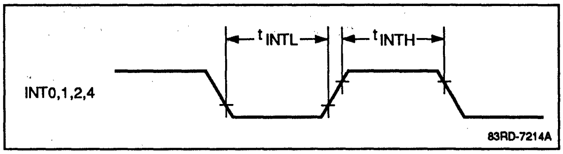
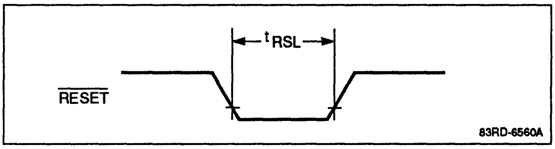


Figure 9. ~~RESET~~ Input Timing



Data Memory STOP Mode Low Voltage Data Retention Characteristics

Mask ROM parts: T_A = −40 to +85°C
Programmable parts: T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		V_{DD} max	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	µA	V_{DDDR} = 2.0 V (µPD752xx and µPD75P216A)
			10	200	µA	V_{DDDR} = 2.0 V (µPD75CG208/CG216A)
Release signal SET time	t_{SREL}	0			µs	
Oscillation stabilization time (Note 2)	t_{WAIT}		2^{17}/fx		ms	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

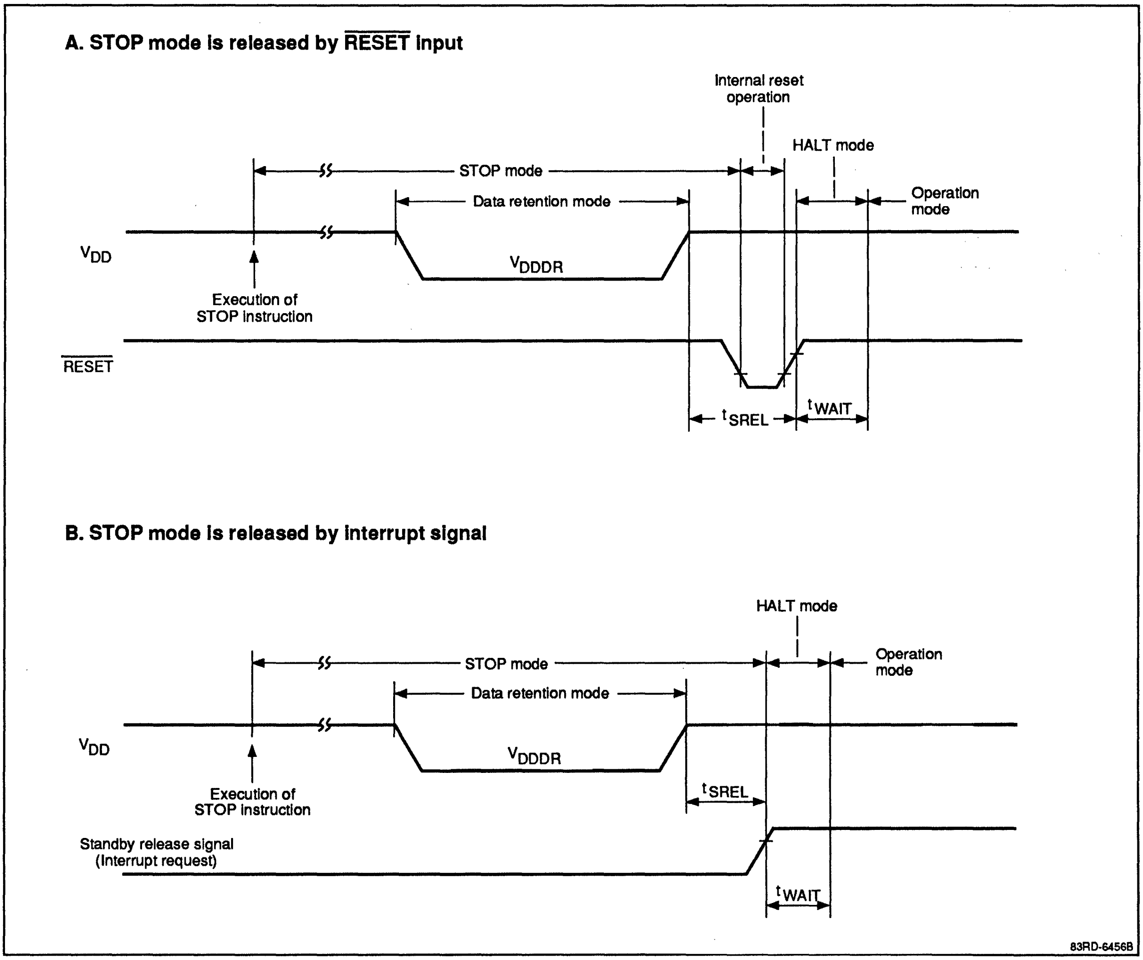
(1) Excludes the on-chip pull-down resistor and power-on reset circuit (mask option) in the mask ROM parts.

(2) Consult the vendor's resonator specification for this value.

(3) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillation is started. The interval timer can be used to delay the CPU from executing instructions using the setting of the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{XX} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{XX}$ (Approx 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (Approx 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (Approx 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (Approx 1.95 ms)

Figure 10. Data Retention Timing



A. STOP mode is released by $\overline{\text{RESET}}$ input

B. STOP mode is released by interrupt signal

DC Programming Characteristics (µPD75P216A only)

T_A = 25 ± 5°C; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	All except X1, X2
	V_{IH2}	V_{DD}–0.5		V_{DD}	V	X1, X2
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	All except X1, X2
	V_{IL2}	0		0.4	V	X1, X2
Input leakage current	I_{IL}			10	µA	V_{IN} = V_{IL} or V_{IH}
High-level output voltage	V_{OH}	V_{DD}–1.0			V	I_{OH} = –1 mA
Low-level output voltage	V_{OL}			0.4	V	I_{OL} = 1.6 mA
V_{DD} supply current	I_{DD}			30	mA	
V_{PP} supply current	I_{PP}			30	mA	MD0 = V_{IL}; MD1 = V_{IH}

Notes:

(1) V_{PP} must not exceed +22.0 V, including overshoot.

(2) V_{DD} is to be applied prior to V_{PP} and to be removed after V_{PP} is removed.

AC Programming Characteristics (μPD75P216A only)

T_A = 25 ± 5°C; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V

Parameter	Symbol	EPROM Symbol (Note 1)	Min	Typ	Max	Unit	Conditions
Address setup time (Note 2)	t_{AS}	t_{AS}	2			μs	
MD1 to MD0 ↓ setup	t_{M1S}	t_{OES}	2			μs	
Data to MD0 ↓ setup	t_{DS}	t_{DS}	2			μs	
Address hold from MD0 ↑ (Note 2)	t_{AH}	t_{AH}	2			μs	
Data hold from MD0 ↑	t_{DH}	t_{DH}	2			μs	
Data output float delay from MD0 ↑	t_{DF}	t_{DF}	0		130	ns	
V_{PP} setup to MD3 ↑	t_{VPS}	t_{VPS}	2			μs	
V_{DD} setup to MD3 ↑	t_{VDS}	t_{VCS}	2			μs	
Initialized program pulse width	t_{PW}	t_{PW}	0.95	1	1.05	ms	
Additional program pulse width	t_{OPW}	t_{OPW}	0.95		21	ms	
MD0 setup to MD1 ↑	t_{MOS}	t_{CES}	2			μs	
Data output delay from MD0 ↓	t_{DV}	t_{DV}			1	μs	MD0 = MD1 = V_{IL}
MD1 hold to MD0 ↑	t_{M1H}	t_{OEH}	2			μs	$t_{M1H} + t_{M1R} \geq 50$ μs
MD1 recovery from MD0 ↓	t_{M1R}	t_{OR}	2			μs	$t_{M1H} + t_{M1R} \geq 50$ μs
Program counter reset	t_{PCR}	–	10			μs	
X1 input high/low level width	t_{XH}, t_{XL}	–	0.125			μs	
X1 input frequency	f_{XX}	–			4.19	MHz	
Initial mode set	t_I	–	2			μs	
MD3 Setup to MD1 ↑	t_{M3S}	–	2			μs	
MD3 hold to MD1 ↓	t_{M3H}	–	2			μs	
MD3 setup to MD0 ↓	t_{M3SR}	–	2			μs	During program read cycle
Address → data output delay time (Note 2)	t_{DAD}	t_{ACC}	2			μs	
Address → data output hold time (Note 2)	t_{HAD}	t_{OH}	0		130	ns	
MD3 output hold from MD0 ↑	t_{M3HR}	–	2			μs	
Data output float delay from MD3 ↓	t_{DFR}	–	2			μs	

Notes:

(1) These symbols correspond to those of the μPD27C256 EPROM.

(2) The internal address signal is incremented by the rising edge of the fourth X1 pulse; it is not connected to an external pin.

Figure 11. OTP Memory Write Timing (Programmable)

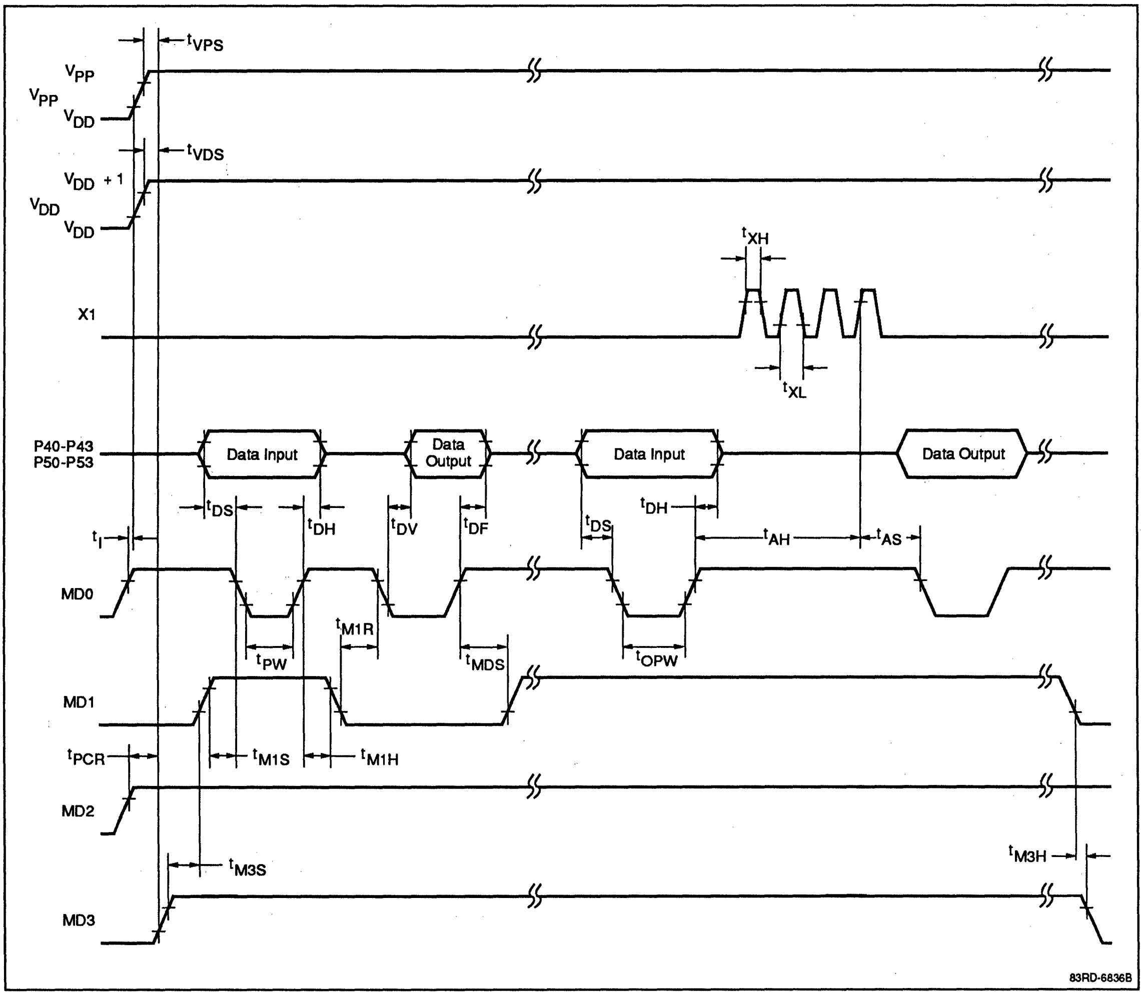
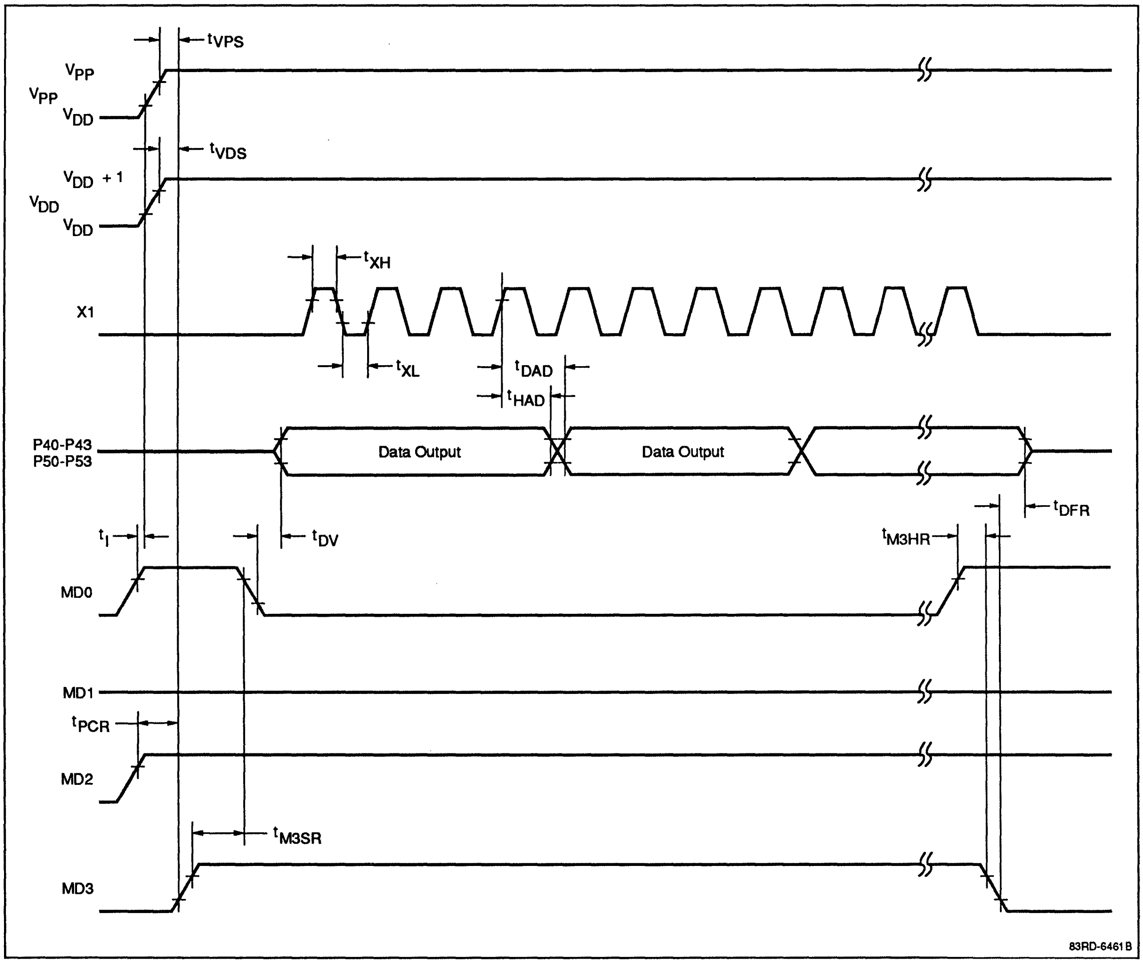


Figure 12. OTP Memory Read Timing (Programmable)



NEC

NEC Electronics Inc.

μPD75268
4-Bit Microcomputer With FIP® (VF) Controller/Driver

Description

The μPD75268 is a low-cost, high-performance, single-chip CMOS microcomputer containing CPU, ROM, RAM, I/O ports, several timer/counters, a FIP controller, vectored interrupts, main and subsystem clocks, and serial interface. The devices are ideally suited for controlling VCRs, microwave ovens, electronic stoves, washing machines, electronic cash registers, audio equipment, and meters.

(For the programmable equivalents, use μPD75CG216 or μPD75P216A.)

Features

- 103 instructions
  - Bit manipulation
  - 4-bit add and subtract
  - 4-bit and 8-bit transfer
  - GETI instruction, to convert one 2-byte or two 1-byte instructions into a single 1-byte instruction
  - 1-byte relative branch
- Fast execution time (@ 4.19 MHz)
  - High-speed cycle: 0.95 μs
  - Lower-voltage cycles: 1.91 and 15.3 μs
- 8064 bytes of program ROM
- 512 x 4 bits of program RAM
- Eight 4-bit registers
- 32 port lines
  - 20 general-purpose I/O, 8 outputs directly drive LEDs (I_{sink} = 15 mA rms)
  - 8 input-only lines
- Three timers
  - 8-bit basic interval timer
  - 8-bit timer/event counter
  - 14-bit watch timer with buzzer output
- Programmable FIP controller with memory area
  - Up to 16 segments
  - Up to 16 digits
- 8-bit serial interface
  - Data transfer can be full duplex or receive only, and can be MSB or LSB first
- Vectored interrupts
  - Three external interrupts
  - Four internal interrupts
- Two interrupt requests
- Standby modes
  - HALT mode: stops CPU only
  - STOP mode: stops main system clock
- Operates with oscillator or ceramic resonator
- CMOS technology, with V_{DD} from 2.7 to 6.0 V

FIP is a registered trademark of NEC Corporation

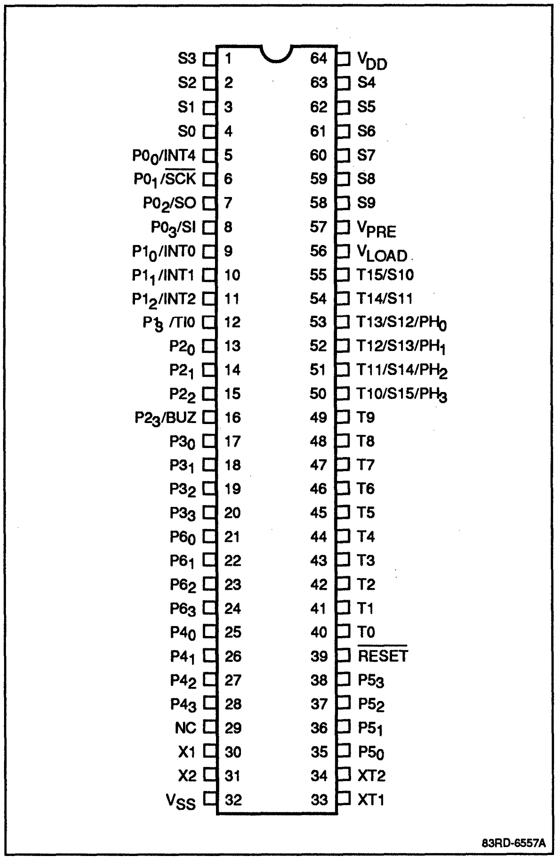
Ordering Information

Part Number	Package Type	ROM
μPD75268CW-xxx	64-pin plastic SDIP	Mask ROM
μPD75268GF-xxx-3BE	64-pin plastic QFP	Mask ROM

Note:
xxx indicates ROM code.

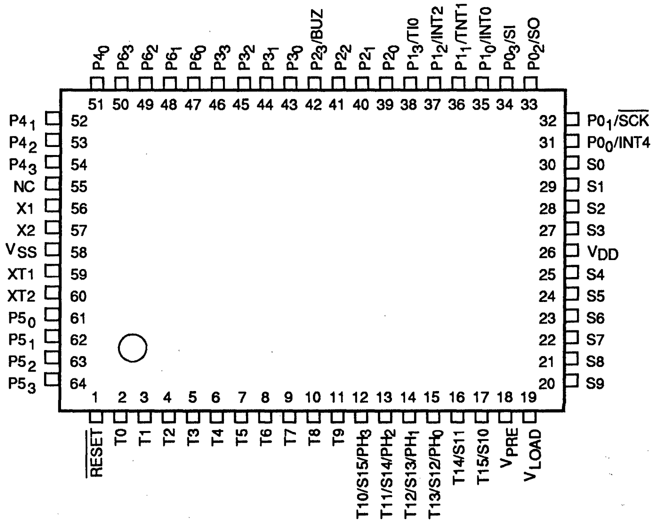
Pin Configurations

64-Pin SDIP



Pin Configurations

64-Pin QFP


P4$_0$ P6$_3$ P6$_2$ P6$_1$ P6$_0$ P3$_3$ P3$_2$ P3$_1$ P3$_0$ P2$_3$/BUZ P2$_2$ P2$_1$ P2$_0$ P1$_3$/TI0 P1$_2$/INT2 P1$_1$/TNT1 P1$_0$/INT0 P0$_3$/SI P0$_2$/SO
51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34 33
P4$_1$ 52
P4$_2$ 53
P4$_3$ 54
NC 55
X1 56
X2 57
V$_{SS}$ 58
XT1 59
XT2 60
P5$_0$ 61
P5$_1$ 62
P5$_2$ 63
P5$_3$ 64
32 P0$_1$/SCK
31 P0$_0$/INT4
30 S0
29 S1
28 S2
27 S3
26 V$_{DD}$
25 S4
24 S5
23 S6
22 S7
21 S8
20 S9
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
RESET T0 T1 T2 T3 T4 T5 T6 T7 T8 T9 T10/S15/PH$_3$ T11/S14/PH$_2$ T12/S13/PH$_1$ T13/S12/PH$_0$ T14/S11 T15/S10 V$_{PRE}$ V$_{LOAD}$

83RD-6558B

Pin Identification

Symbol	Function
P0$_0$/INT4	Port 0 input; interrupt 4
P0$_1$/$\overline{SCK}$	Port 0 input; serial clock
P0$_2$/SO	Port 0 input; serial out
P0$_3$/SI	Port 0 input; serial in
P1$_0$/INT0	Port 1 input; interrupt 0
P1$_1$/INT1	Port 1 input; interrupt 1
P1$_2$/INT2	Port 1 input; interrupt 2
P1$_3$/TI0	Port 1 input; timer 0 input
P2$_0$-P2$_2$	Port 2 I/O
P2$_3$/BUZ	Port 2 I/O; buzzer output
P3$_0$-P3$_3$	Port 3 I/O
P4$_0$-P4$_3$	Port 4 I/O
P5$_0$-P5$_3$	Port 5 I/O
P6$_0$-P6$_3$	Port 6 I/O
PH$_0$/T13/S12	Port H output; digit select line; segment select line
PH$_1$/T12/S13	Port H output; digit select line; segment select line
PH$_2$/T11/S14	Port H output; digit select line; segment select line
PH$_3$/T10/S15	Port H output; digit select line; segment select line
S0-S9	FIP segment outputs
T0-T9	FIP digit select outputs

Symbol	Function
T14/S11 T15/S10	Digit selects T14 and T15; segment selects S10 and S11.
X1, X2	Main clock inputs
XT1, XT2	Subsystem clock inputs
$\overline{RESET}$	Reset input
V$_{PRE}$	FIP predriver negative supply voltage
V$_{LOAD}$	FIP high-voltage negative supply voltage
V$_{DD}$	Positive power supply
V$_{SS}$	Ground

PIN FUNCTIONS

P0$_0$/INT4, P0$_1$/$\overline{SCK}$, P0$_2$/SO, P0$_3$/SI

These pins can be used as 4-bit input port 0. Or, P0$_0$ can also be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. P0$_1$-P0$_3$ may also be used for the serial interface in the 2/3 wire mode. SI is the serial input, SO is the serial output, and $\overline{SCK}$ is the serial clock. Reset causes these pins to default to the port 0 input mode.

$P1_0$/INT0, $P1_1$/INT1, $P1_2$/INT2, $P1_3$/TI0

These pins can be used as 4-bit input port 1. Or, $P1_0$ and $P1_1$ can also be used for edge-triggered interrupts INT0 and INT1. $P1_2$ can be used for INT2, which is also an edge-triggered input, but one which generates an interrupt request and does not cause an interrupt. $P1_3$ can be used as an input clock to the timer/event counter to count external events. Reset causes these pins to default to the port 1 input mode.

$P2_0$, $P2_1$, $P2_2$, $P2_3$/BUZ

These pins can be used as 4-bit I/O port 2. When used as an output, the data is latched. When used as an input port the port outputs are three-state. $P2_3$ can also be used to output square waves for a buzzer. Reset causes these pins to default to the port 2 input mode.

$P3_0$-$P3_3$, $P4_0$-$P4_3$, $P5_0$-$P5_3$

Ports 3, 4, and port 5 are 4-bit I/O ports with latched outputs. Ports 4 and 5 will directly drive LEDs. Each bit of port 3 can be independently programmed to be either an input or an output, while ports 4 and 5 can be programmed to be either an input port or an output port. A reset signal causes these ports to default to the input mode.

$P6_0$-$P6_3$

Port 6 is a 4-bit I/O port. Outputs are latched, and each bit can be independently programmed to be either an input or an output. Port 6 can have pulldown resistors added as a mask option. A reset signal causes this port to default to the input mode.

PH_0/T13/S12, PH_1/T12/S13, PH_2/T11/S14, PH_3/T10/S15

Port H is a 4-bit output-only port, with P-channel open-drain outputs capable of directly driving LEDs. Output pulldown resistors can be selected as a mask-option. Alternatively, these pins can be used as high-voltage digit/segment outputs (T13/S15 - T10/S12). A reset signal causes this port to default to the high-impedance state; if mask-option resistors are present the output goes low.

S0-S9

These are high-voltage outputs used as FIP controller segment signals. pulldown resistors can be selected as a mask-option. A reset signal sets these pins to the high-impedance state; if mask-option resistors are present the outputs go low.

T0-T9

These are high-voltage outputs used as FIP controller digit select timing signals. pulldown resistors can be selected as a mask-option. A reset signal sets these pins to the high-impedance state; if mask-option resistors are present the outputs go low.

T14/S11, T15/S10

These two pins provide additional digit or segment selectors. When not used for the display they can be used as static outputs. Internal pulldown resistors are available as a mask option.

X1, X2

These pins are the main system clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

XT1, XT2

These pins are the subsystem clock inputs. The clock can be either a ceramic resonator or a crystal; an external logic signal may also be used.

$\overline{\text{RESET}}$

This is the reset input, and it is active low.

V_{PRE}

This is the power supply for the predrivers of the FIP controller.

V_{LOAD}

This pin is used to supply power to the output drivers for the segment and digit select pins of the FIP controller.

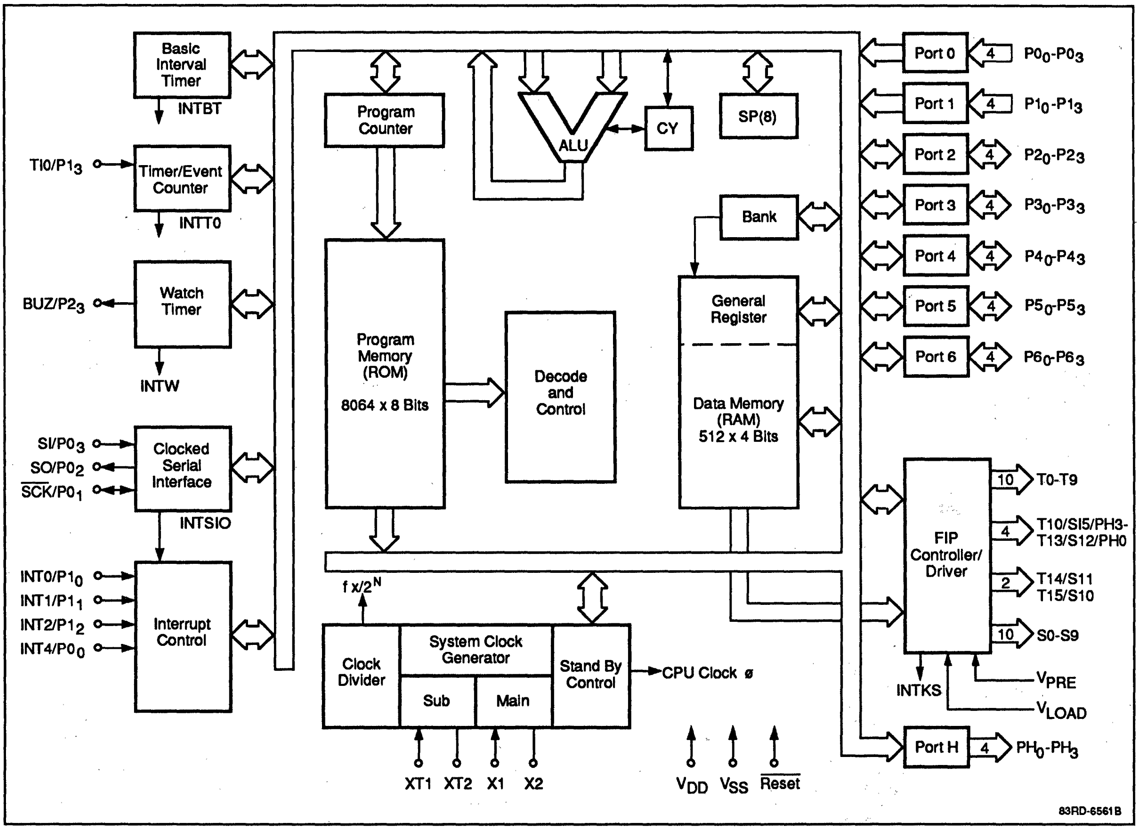
V_{DD}

The system positive power supply pin.

V_{SS}

System ground.

Block Diagram



4

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Supply voltage, V_{LOAD}	V_{DD} −40 to V_{DD}+0.3 V
Supply voltage, V_{PRE}	V_{DD} −12 to V_{DD}+0.3 V
Input voltage, V_{IN}	−0.3 to V_{DD} + 0.3 V
Output voltage, V_O (other than display)	−0.3 to V_{DD} + 0.3 V
Output voltage, V_{OD} (display pins)	V_{DD} −40 to V_{DD}+0.3 V
High-level output current, I_{OH} (single pin; other than display)	−15 mA
High-level output current, I_{OH} (single pin; S0-S9)	−15 mA
High-level output current, I_{OH} (single pin; T0-T15)	−30 mA
High-level output current, I_{OH} (total of all pins other than display)	−20 mA
High-level output current, I_{OH} (total of all display outputs)	−120 mA
Low-level output current, I_{OL} (single pin)	17 mA
Low-level output current, I_{OL} (total of all pins)	60 mA
Power dissipation, P_D (plastic QFP)	450 mW (Note 1)
Power dissipation, P_D (plastic SDIP)	600 mW (Note 1)
Storage temperature, t_{STG}	−65 to + 150°C
Operating temperature, t_{OPT}	−40 to +85°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Notes:

(1) Care must be taken when designing the microcomputer that the total power dissipation does not exceed the maximum allowable. Power is dissipated in three areas:

a. At the CPU. P_D is calculated by the product of V_{DD} (max) and I_{DD1}(max).

b. By the output pins. Total power dissipation is the sum of the values for each pin when maximum current is applied.

c. By the pulldown resistors.

Main System Clock Oscillator Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 1A)	Oscillation frequency (Note 1)	f_{XX}	2.0		5.0	MHz	
	Oscillation stabilization time (Note 2)				4 (Note 3)	ms	
Crystal resonator (Figure 1A)	Oscillation frequency (Note 1)	f_{XX}	2.0	4.19	5.0	MHz	
	Oscillation stabilization time (Note 2)				10 (Note 3)	ms	V_{DD} = 4.5 to 6.0 V
					30 (Note 3)	ms	
External clock (Figure 1B)	X1 input frequency	f_X	2.0		5.0	MHz	
	X1 input high- and low-level width	t_{XH}, t_{XL}	100		250	ns	

Notes:

(1) The oscillation frequency and X1 input frequency are included only to show the frequency range of the oscillators. Refer to the AC Characteristics table for actual instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillator to stabilize after V_{DD} is applied or the STOP mode is released.

(3) Values shown are typical values for resonators. Actual values should be obtained from the manufacturer's specification sheets.

Subsystem Clock Oscillator Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 2A)	Oscillation frequency (Note 1)	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time (Note 2)			1.0	2 (Note 3)	s	V_{DD} = 4.5 to 6.0 V
					10 (Note 3)	s	
External clock (Figure 2B)	XT1 input frequency	f_{XT}	32		100	kHz	
	XT1 input high- and low-level width	t_{XTH}, t_{XTL}	10		32	μs	

Notes:

(1) The oscillation frequency and X1 input frequency are included only to show the frequency range of the oscillators. Refer to the AC Characteristics table for actual instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillator to stabilize after V_{DD} is applied or the STOP mode is released.

(3) Values shown are typical values for resonators. Actual values should be obtained from the manufacturer's specification sheets.

Figure 1. Main System Clock Configurations

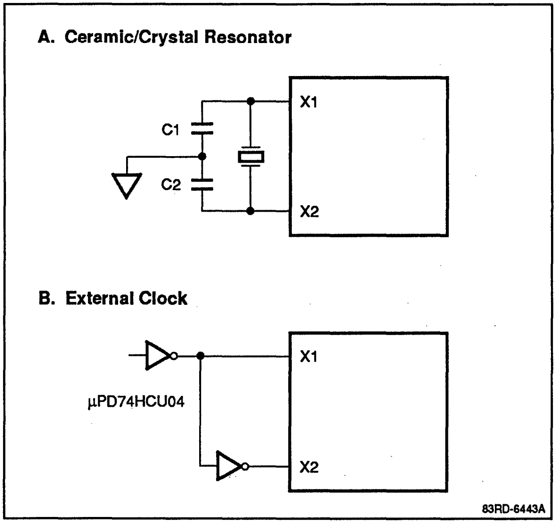
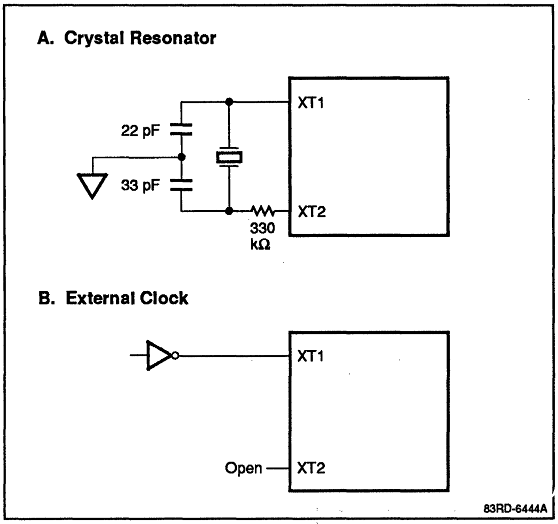


Figure 2. Subsystem Clock Configurations



Capacitance

V_{DD} = 0 V; T_A = 25°C

Parameter	Symbol	Min	Max	Unit	Conditions
Input capacitance	C_{IN}		15	pF	
Output capacitance; Other than display output pins	C_{OUT1}		15	pF	f = 1 MHz All unmeasured pins returned to ground
Output capacitance; Display output pins	C_{OUT2}		35	pF	
I/O capacitance	C_{IO}		15	pF	

Operating Supply Voltage

T_A = −40 to +85°C

Parameter	Min	Max	Unit
CPU (Note 1)	(Note 2)	6.0	V
Display controller	4.5	6.0	V
Other hardware (Note 1)	2.7	6.0	V

Notes:

(1) The CPU does not include the system clock oscillator and the display controller.

(2) Varies according to the cycle time. See AC Characteristics.

DC Characteristics

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	All except ports 0 and 1; $\overline{\text{RESET}}$; X1, X2, XT1
	V_{IH2}	0.75 V_{DD}		V_{DD}	V	Ports 0 and 1; $\overline{\text{RESET}}$
	V_{IH3}	V_{DD}−0.4		V_{DD}	V	X1, X2, XT1
	V_{IH4}	0.65 V_{DD}		V_{DD}	V	Port 6; V_{DD} = 4.5 to 6.0 V
		0.7 V_{DD}		V_{DD}	V	Port 6; V_{DD} = 2.7 to 6.0 V
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	All except ports 0, 1, and 6; $\overline{\text{RESET}}$; X1, X2, XT1
	V_{IL2}	0		0.2 V_{DD}	V	Ports 0, 1 and 6; $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2, XT1
High-level output voltage	V_{OH}	V_{DD}−1.0			V	All outputs; V_{DD} = 4.6 to 6.0 V; I_{OH} = −1 mA
		V_{DD}−0.5			V	All outputs; V_{DD} = 2.7 to 6.0 V; I_{OH} = −100 μA
Low-level output voltage	V_{OL}		0.4	2.0	V	Ports 4 and 5; V_{DD} = 4.6 to 6.0 V; I_{OL}= 15 mA
				0.4	V	All output pins; V_{DD} = 4.6 to 6.0 V; I_{OL}= 1.6 mA
				0.5	V	All output pins; V_{DD} = 2.7 to 6.0 V; I_{OL}= 400 μA
High-level input leakage current	I_{LIH1}			3	μA	All except X1, X2, and XT1; V_{IN} = V_{DD}
	I_{LIH2}			20	μA	X1, X2, and XT1; V_{IN} = V_{DD}
Low-level input leakage current	I_{LIL1}			−3	μA	All except X1, X2, and XT1; V_{IN} = 0 V
	I_{LIL2}			−20	μA	X1, X2, and XT1; V_{IN} = 0 V
High-level output leakage current	I_{LOH}			3	μA	All output pins; V_{OUT} = V_{DD}
Low-level output leakage current	I_{LOL1}			−3	μA	All except display output pins; V_{OUT} = 0 V
	I_{LOL2}			−10	μA	Display output pins; V_{OUT} = V_{LOAD} = V_{DD}−35 V
Display output current	I_{OD}	−3	−5.5		mA	S0 - S9; (Note 1) and Recommended External Circuit (figure 3)
		−1.5	−3.5		mA	S0 -S9; (Note 2)
		−15	−22		mA	T0 - T15; (Note 1) and Recommended External Circuit (figure 3)
		−7	−15		mA	T0 - T15; (Note 2)

DC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Internal pulldown resistor (mask option)	R_{P6}	20	80	200	kΩ	Port 6; V_{DD} = 4.5 to 6.0 V; V_{IN} = V_{DD}
		20		1000	kΩ	Port 6; V_{DD} = 2.7 to 6.0 V; V_{IN} = V_{DD}
	R_L	25	70	135	kΩ	Display output pins; V_{DD}–V_{LOAD} = 35 V
Supply current (Note 6)	I_{DD1}		3.0	9.0	mA	V_{DD} = 5 V ± 10% (Notes 3, 4)
			0.55	1.5	mA	V_{DD} = 3 V ± 10% (Notes 3, 5)
	I_{DD2}		600	1800	μA	HALT mode; V_{DD} = 5 V ± 10% (Note 3)
			200	600	μA	HALT mode; V_{DD} = 3 V ± 10% (Note 3)
	I_{DD3}		40	120	μA	V_{DD} = 3 V ± 10% (Notes 7, 8)
	I_{DD4}		5	15	μA	HALT mode; V_{DD} = 3 V ± 10% (Notes 7, 8)
	I_{DD5}		0.5	20	μA	STOP mode; XT1 = 0V; V_{DD} = 5 V ± 10 %
			0.1	10	μA	STOP mode; XT1 = 0V; V_{DD} = 3 V ± 10 %

Notes:

(1) V_{DD} = 4.5 to 6.0 V; V_{OD} = V_{DD} −2 V; V_{PRE} = V_{DD} −9 ±1 V

(2) V_{DD} = 4.5 to 6.0 V; V_{OD} = V_{DD} −2 V; V_{PRE} = 0 V

(3) 4.19 MHz crystal oscillator; C1 = C2 = 15 pF.

(4) Value during high-speed operation; processor control clock (PCC) is set to 0011.

(5) Value during low-speed operation; processor control clock (PCC) is set to 0000.

(6) Does not include internal pulldown resistor current.

(7) 32 MHz crystal oscillator

(8) Value when the system clock control register (SCC) is set to 1001, main system clock is stopped, and the subsystem clock operates the chip.

AC Characteristics

T_A = -40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time minimum instruction execution time — (Note 1)	t_{CY}	0.95		32	μs	CPU using main system clock; V_{DD} = 4.5 to 6.0 V
		3.8		32	μs	CPU using main system clock; V_{DD} = 2.7 to 6.0 V
		114	122	125	μs	CPU using subsystem clock; V_{DD} = 2.7 to 6.0 V
TIO input frequency	f_{TI}	0		0.6	MHz	V_{DD} = 4.5 to 6.0 V
		0		165	kHz	V_{DD} = 2.7 to 6.0 V
TIO input high- and low-level width	t_{IH}, t_{IL}	0.83			μs	V_{DD} = 4.5 to 6.0 V
		3			μs	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ cycle time	t_{KCY}	0.8			μs	Input; V_{DD} = 4.5 to 6.0 V
		0.95			μs	Output; V_{DD} = 4.5 to 6.0 V
		3.2			μs	Input; V_{DD} = 2.7 to 6.0 V
		3.8			μs	Output; V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ high- and low-level width	t_{KH}, t_{KL}	0.4			μs	Input; V_{DD} = 4.5 to 6.0 V
		$0.5t_{KCY}$−50			ns	Output; V_{DD} = 4.5 to 6.0 V
		1.6			μs	Input; V_{DD} = 2.7 to 6.0 V
		$0.5t_{KCY}$−150			ns	Output; V_{DD} = 2.7 to 6.0 V

AC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
SI to $\overline{SCK}$ ↑ setup time	t_{SIK}	100			ns	
SI to $\overline{SCK}$ ↑ hold time	t_{KSI}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO}			300	ns	V_{DD} = 4.5 to 6.0 V
				1000	ns	V_{DD} = 2.7 to 6.0 V
Interrupt inputs low- and high-level width	t_{INTH}, t_{INTL}	(Note 2)			µs	INT0
		$2t_{CY}$			µs	INT1
		10			µs	INT2, INT4
$\overline{RESET}$ low-level width	t_{RSL}	10			µs	

Notes:

(1) Cycle time is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See the graph depicting the Supply Voltage to the cycle time (figure 4) when the microcomputer is operating on the main system clock.

(2) $2t_{CY}$ or $128/f_{XX}$, depending on the setting of the interrupt mode register (IM0).

Data Memory STOP Mode Low Voltage Data Retention Characteristics

T_A = −40 to +85°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		6.0	V	
Data retention current	I_{DDDR}		0.1	10	µA	V_{DDDR} = 2.0 V (Note 1)
Release signal SET time	t_{SREL}	0			µs	
Oscillation stabilization time (Note 2)	t_{WAIT}		(2)		ms	Release by $\overline{RESET}$ input
			(2)		ms	Release by interrupt request

Notes:

(1) Excludes current in the internal pulldown resistors.

(2) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing; consult the vendor's resonator or crystal specifications sheet for this value. This time is required to prevent unstable operation while the oscillation is started. The interval timer can be used to delay the CPU from executing instructions using the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time
–	0	0	0	$2^{20}/f_{XX}$ (Approx 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (Approx 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (Approx 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (Approx 1.95 ms)

Figure 3. Recommended External Circuit

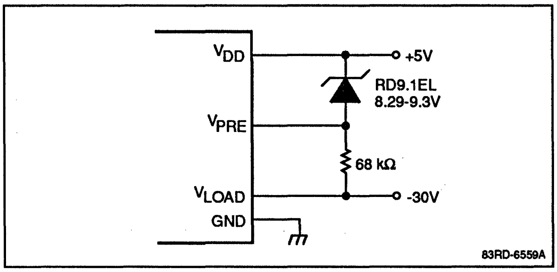
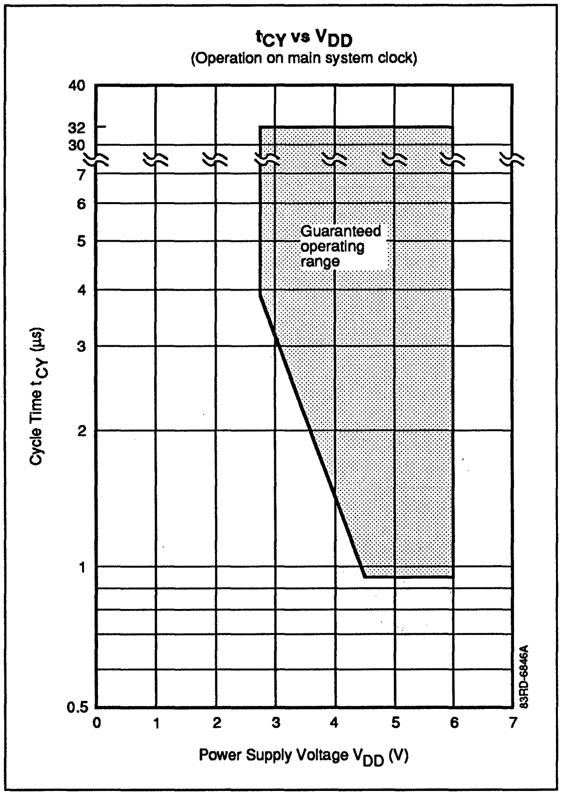
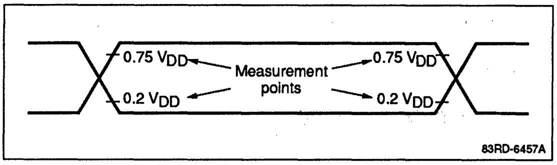


Figure 4. Guaranteed Operating Range

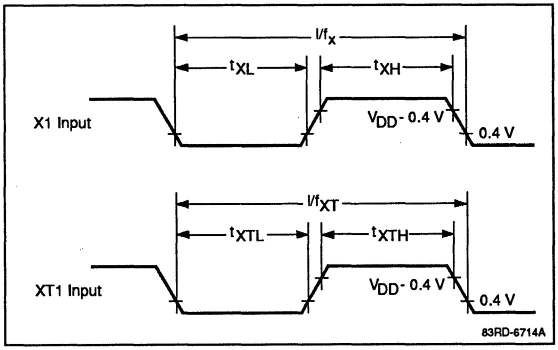


TIMING WAVEFORMS

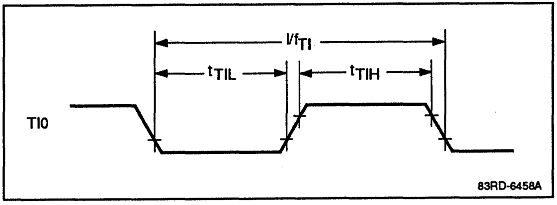
AC Timing Measurement Points (Excluding X1 and XT1 input pins)



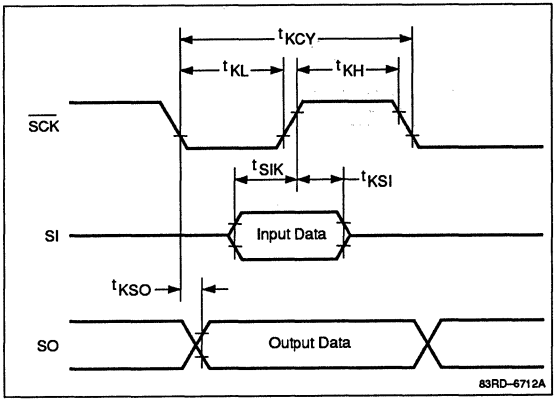
Clock Timing



TI0 Timing

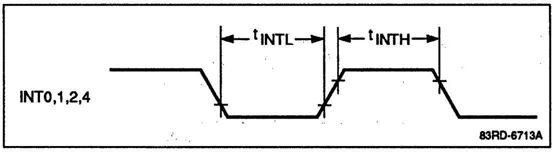


Serial Transfer Timing

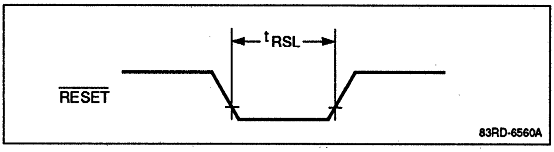


4

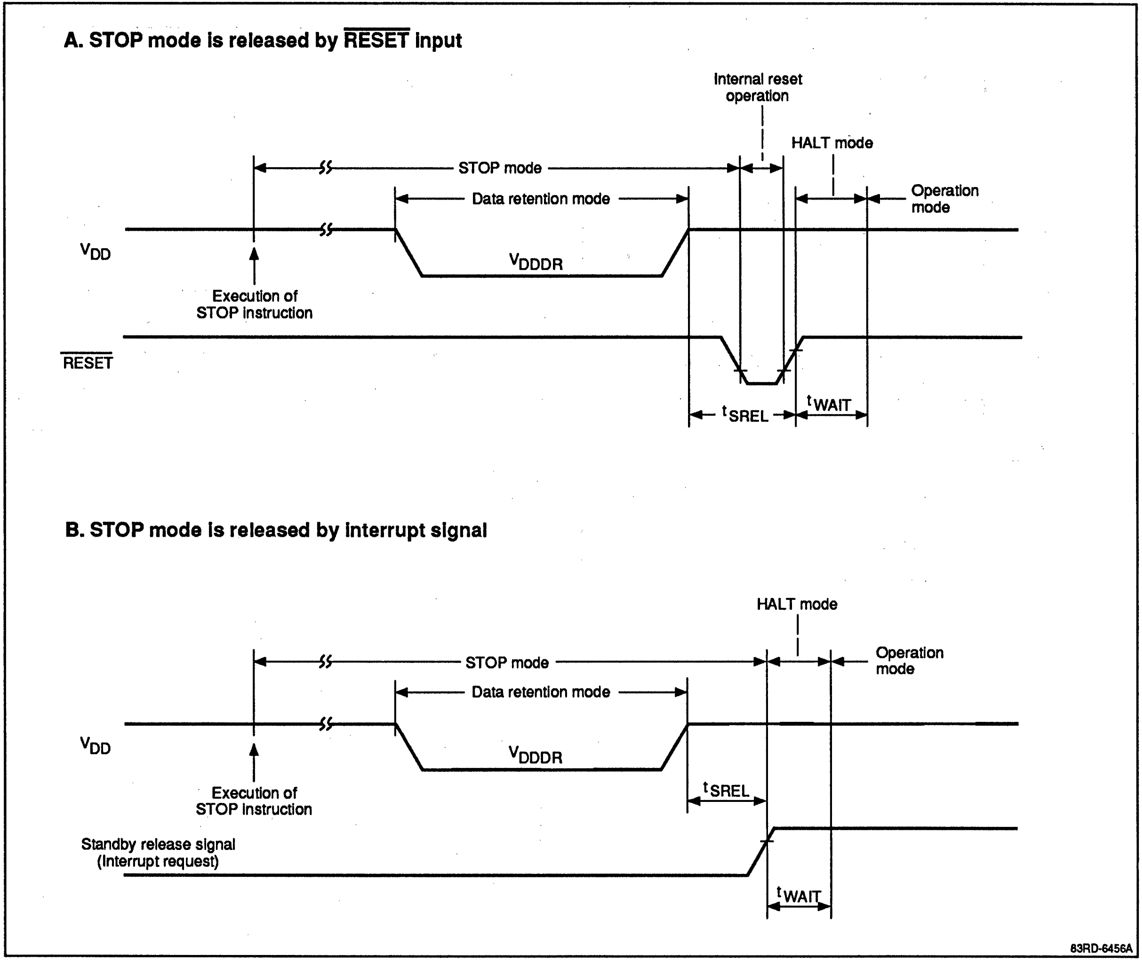
Interrupt Input Timing



RESET Input Timing



Data Retention Timing


A. STOP mode is released by RESET input
Internal reset operation
HALT mode
STOP mode
Data retention mode
Operation mode
VDD
VDDDR
Execution of STOP instruction
RESET
tSREL
tWAIT
B. STOP mode is released by interrupt signal
HALT mode
STOP mode
Operation mode
Data retention mode
VDD
VDDDR
Execution of STOP instruction
tSREL
Standby release signal (Interrupt request)
tWAIT
83RD-6456A

NEC
NEC Electronics Inc.

μPD7530x/31x/P308/P316 4-Bit Microcomputers With LCD Controller/Driver

Description

The μPD7530x/31x is a family of high-performance single-chip CMOS microcomputers containing CPU, ROM, RAM, I/O ports, several timer/counters, vectored interrupts, subsystem clock, and serial interface.

The instruction set allows the user to manipulate RAM data and I/O ports in 1-, 4-, and 8-bit units. The devices are ideally suited for controlling VCRs, telephones, meters, handheld instruments, and devices with LCDs.

Development tools include a low-cost in-circuit emulator, relocatable assembler, and C-like structured assembler.

Both EPROM and OTP versions are available. See ordering information.

Features

□ 103 instructions
—Bit manipulation
—4-bit and 8-bit transfer
—1-byte relative branch
—GETI instruction converts one 2-byte/3-byte or two 1-byte instructions into a single 1-byte instruction

□ Fast execution time
(Main system clock @ 4.19 MHz)
—High-speed cycle: 0.95 μs
—Lower-voltage cycles: 1.91 and 15.3 μs

□ Program ROM
—μPD75304: 4096 bytes
—μPD75306: 6016 bytes
—μPD75308/P308: 8064 bytes
—μPD75312: 12160 bytes
—μPD75316/P316/P316A: 16256 bytes

□ Data memory (RAM)
—512 x 4 bits
—Allows operation on 1, 4, or 8 bits

□ Bit sequential buffer
—16-bit, bit manipulation memory

□ Eight 4-bit registers or four 8-bit registers

□ Accumulators
—1-bit (CY)
—4-bit (A)
—8-bit (XA)

□ 24 I/O lines
—All outputs directly drive LEDs
(I_{sink} = 15 mA rms)
—8 N-channel open-drain, can withstand 10 V
—8 input-only lines

□ One external event input

□ Subsystem clock allows watch timer and LCD controller to operate in STOP mode

□ Three timers
—8-bit basic interval timer
—8-bit timer/event counter
—14-bit watch timer

□ LCD controller/driver
—32 segment lines
—4 common lines
—4 operating modes: static; multiplexed 1/2 bias; triplexed 1/2 or 1/3 bias; quadruplexed 1/3 bias
—LCD resistor ladder available as a mask option

□ 8-bit serial interface
—SBI mode
—2- or 3-wire mode: Data transfer can be full duplex or receive only, and can be MSB or LSB first

□ Vectored interrupts
—Three external interrupts
—Three internal interrupts
—Nine inputs which generate an interrupt request

□ Standby modes
—HALT mode: stops CPU only
—STOP mode: stops main system clock

□ Optional pullup resistors
—By software: 23 lines
—By mask option: 8 lines

□ Operates with oscillator or ceramic resonator

□ CMOS operation, with V_{DD} from 2.7 to 6.0 V

□ Programmable versions
—OTP & EPROM: μPD75P308
—OTP: μPD75P316
—OTP, low voltage: μPD75P316AGF (Note)
—EPROM, low voltage: μPD75P316AK (Note)

□ Low operating current (@5 V and 4.19 MHz)
—Normal operation: 2.5 mA typical
—HALT mode: 0.5 mA typical
—STOP mode: 0.1 mA typical

Note: Low voltage target spec of 2.7 to 6.0 V operation. Contact your local NEC Sales Office for latest information; none of the electrical specifications in this data sheet directly apply to these parts.

Ordering Information

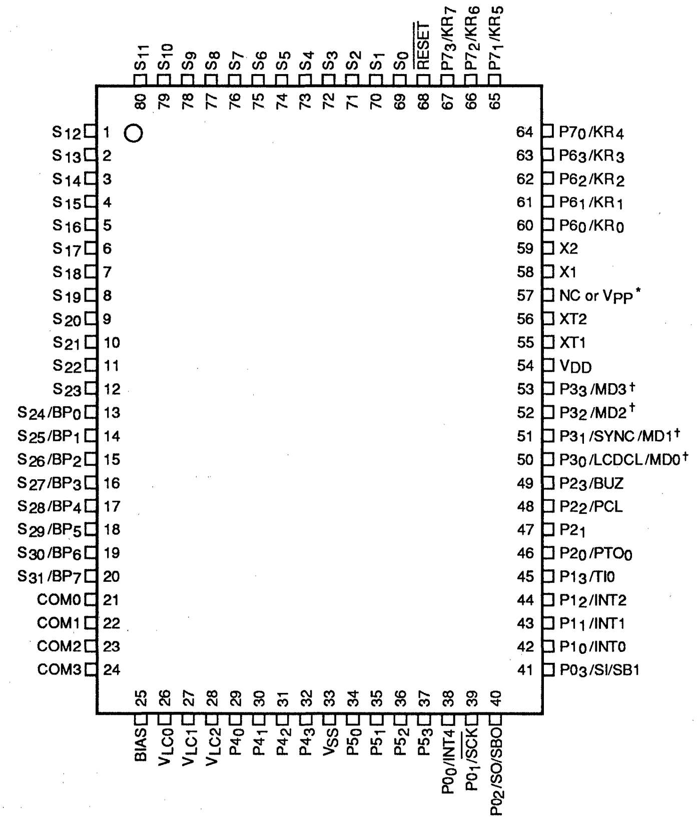
Part Number	Package Type	ROM
µPD75304GF-xxx-3B9	80-pin plastic QFP	Mask ROM
µPD75306GF-xxx-3B9	80-pin plastic QFP	Mask ROM
µPD75308GF-xxx-3B9	80-pin plastic QFP	Mask ROM
µPD75P308GF-3B9	80-pin plastic QFP	OTP
µPD75P308K	80-pin ceramic LCC w/window	EPROM
µPD75312GF-xxx-3B9	80-pin plastic QFP	Mask ROM

Part Number	Package Type	ROM
µPD75316GF-xxx-3B9	80-pin plastic QFP	Mask ROM
µPD75P316GF-3B9	80-pin plastic QFP	OTP
µPD75P316AGF-3B9	80-pin plastic QFP	Low voltage OTP
µPD75P316AK	80-pin ceramic LCC w/window	Low voltage EPROM

Notes:

(1) xxx indicates ROM code suffix.

Pin Configurations



* Pin 57 is VPP in the programmable package. Connect this pin to VDD in the µPD75P308/P316/P316A.

† MD0-MD3 are used as the programming mode selection pins on the µPD75P308/P316/P316A during EPROM and OTP programming and verification.

49NR-562B

Pin Identification

Symbol	Function
BIAS	LCD power bias output
BP_0/S24 BP_1/S25 BP_2/S26 BP_3/S27 BP_4/S28 BP_5/S29 BP_6/S30 BP_7/S31	1-bit output ports BP_0-BP_7; LCD segments S24-S31
COM0-COM3	LCD Common output 0-3
NC/V_{PP}	No connection (programming pin for µPD75P308/P316/P316A)
$P0_0$/INT4	Port 0 input; interrupt 4
$P0_1$/$\overline{SCK}$	Port 0 input; serial clock
$P0_2$/SO/SB0	Port 0 input; serial out
$P0_3$/SI/SB1	Port 0 input; serial in
$P1_0$/INT0	Port 1 input; interrupt 0
$P1_1$/INT1	Port 1 input; interrupt 1
$P2_1$/INT2	Port 1 input; interrupt 2
$P1_3$/TI0	Port 1 input; timer 0 input
$P2_0$/PTO_0	Port 2 I/O; timer/event counter output
$P2_1$	Port 2 I/O
$P2_2$/PCL	Port 2 I/O; clock output
$P2_3$/BUZ	Port 2 I/O; buzzer output
$P3_0$/LCDCL/MD0	Port 3 I/O; LCD clock output ; programming mode select 0 (µPD75P308/P316/P316A)
$P3_1$/SYNC/MD1	Port 3 I/O; SYNC output; programming mode select 1 (µPD75P308/P316/P316A)
$P3_2$/MD2	Port 3 I/O; programming mode select 2 (µPD75P308/P316/P316A)
$P3_3$/MD3	Port 3 I/O; programming mode select 3 (µPD75P308/P316/P316A)
$P4_0$-$P4_3$	Port 4 I/O
$P5_0$-$P5_3$	Port 5 I/O
$P6_0$/KR0	Port 6 I/O; key scan input 0
$P6_1$/KR1	Port 6 I/O; key scan input 1
$P6_2$/KR2	Port 6 I/O; key scan input 2
$P6_3$/KR3	Port 6 I/O; key scan input 3
$P7_0$/KR4	Port 7 I/O; key scan input 4
$P7_1$/KR5	Port 7 I/O; key scan input 5
$P7_2$/KR6	Port 7 I/O; key scan input 6
$P7_3$/KR7	Port 7 I/O; key scan input 7
$\overline{RESET}$	Reset input
S0-S23	LCD segment output
V_{LC0}	LCD drive level 0
V_{LC1}	LCD drive level 1
V_{LC2}	LCD drive level 2
X1, X2	Main clock inputs
XT1, XT2	Subsystem clock inputs
V_{DD}	Positive power supply
V_{SS}	Ground

PIN FUNCTIONS

$P0_0$/INT4, $P0_1$/$\overline{SCK}$, $P0_2$/SO/SB0, $P0_3$/SI/SB1

These pins can be used as 4-bit input port 0. $P0_0$ can also be used for vectored interrupt 4, which interrupts on either the leading edge or the trailing edge of the signal. $P0_1$-$P0_3$ may also be used for the serial interface. SI is the serial input, SO is the serial output, and $\overline{SCK}$ is the serial clock. Reset causes these pins to default to the port 0 input mode.

$P1_0$/INT0, $P1_1$/INT1, $P1_2$/INT2, $P1_3$/TI0

These pins can be used as 4-bit input port 1. $P1_0$ and $P1_1$ can also be used for edge-triggered interrupts INT0 and INT1. $P1_2$ can be used for INT2, which is also an edge-triggered input, but one which generates an Interrupt request and does not cause an Interrupt. $P1_3$ can be used as an input clock to the timer/event counter to count external events. Reset causes these pins to default to the port 1 input mode.

$P2_0$/PTO0, $P2_1$, $P2_2$/PCL, $P2_3$/BUZ

These pins can be used as 4-bit I/O port 2. When used as an output the data is latched. When used as an input port the port outputs are three-state. $P2_0$ can also be used as the output of the timer/event counter flip flop (TOUT); $P2_2$ can be used as the output (PCL) for the clock generator; and $P2_3$ can be used to output square waves for a buzzer. Reset causes these pins to default to the port 2 input mode.

$P3_0$/LCDCL/MD0, $P3_1$/SYNC/MD1, $P3_2$/MD2, $P3_3$/MD3

These pins are used for I/O Port 3. Each bit in this port can be independently programmed to be either an input or an output. This port has latched outputs, and can directly drive LEDs. $P3_0$ and $P3_1$ can also be used respectively as LCD clock and LCD sync outputs. $P3_0$-$P3_3$ are used as the programming mode select pins for the μPD75P308/P316/P316A during EPROM/OTP programming and verification. A reset signal causes this port to default to the input mode.

$P4_0$-$P4_3$, $P5_0$-$P5_3$

Port 4 and Port 5 are identical 4-bit I/O ports which can be combined together to function as a single 8-bit port. Latched outputs will directly drive LEDs. Outputs are N-channel open drain, and can withstand up to 10 volts; pull-up resistor mask options are available for these ports. A reset signal causes these ports to default to the input mode.

$P6_0$/KR0, $P6_1$/KR1, $P6_2$/KR2, $P6_3$/KR3 $P7_0$/KR4, $P7_1$/KR5, $P7_2$/KR6, $P7_3$/KR7

Ports 6 and 7 are 4-bit I/O ports which can be combined together to function as a single 8-bit port. Outputs are latched. Each pin of port 6 can be independently programmed to be either an input or an output, while port 7 can be programmed to be either all inputs or all outputs. Alternately, these pins may be used to detect the falling edge of inputs KR0 - KR3 (port 6) and KR4 - KR7 (port 7). A reset signal causes these ports to default to the input mode.

S0-S23

These are the LCD segment drivers.

COM0-COM3

These are the LCD common input drivers.

BP_0/S24-BP_7/S31

These can be used either as eight 1-bit ports or as additional LCD segment drivers. When used as segment outputs they are selectable in 4-bit increments.

V_{LC0}-V_{LC2}

These pins are used to set the drive levels for the LCD. If the internal resistor ladder mask option is selected, these pins are outputs; if the internal resistor ladder is not selected, these pins are inputs to which an external resistor network must be connected.

BIAS

Thes output is used in conjunction with the V_{LC0} - V_{LC2} pins to set the LCD contrast level.

NC/V_{PP}

This pin may be left unconnected when using the μPD7530x/31x. For the μPD75P308/P316/P316A, this pin is used as the programming voltage input during the EPROM write/verify cycles. When the devices are not being programmed, this pin should be connected to V_{DD}.

X1, X2

These pins are the main system clock inputs. The input can be either a ceramic resonator or a crystal; an external logic signal may also be used.

XT1, XT2

These pins are the subsystem clock inputs. The input can be either a ceramic resonator or a crystal; an external logic signal may also be used.

$\overline{\text{RESET}}$

This is the reset input, and it is active low.

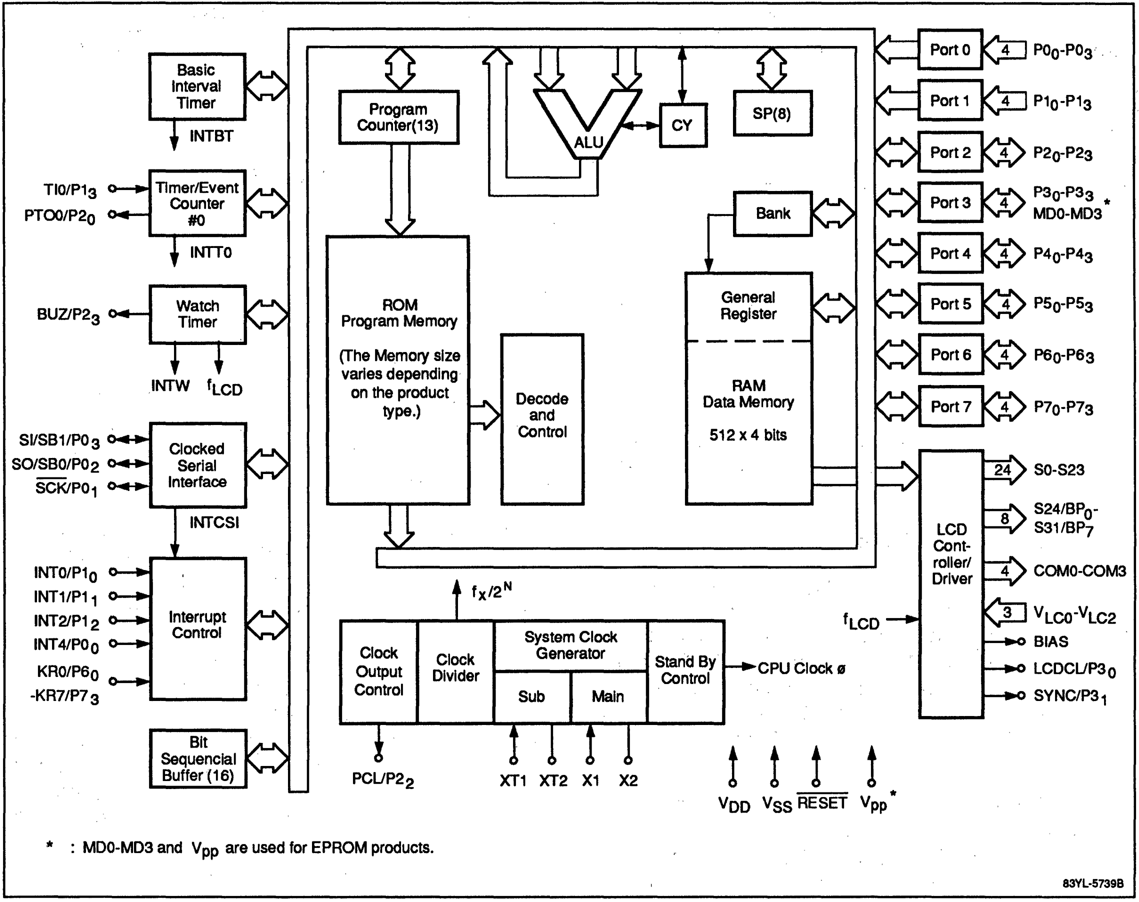
V_{DD}

The system positive power supply pin.

V_{SS}

System ground.

Block Diagram



* : MD0-MD3 and V_{PP} are used for EPROM products.

Product Comparison

Item	μPD75304	μPD75306	μPD75308	μPD75P308	μPD75312	μPD75316	μPD75P316/A
Program memory	Mask ROM 000H–FFFH 4096 x 8 bits	Mask ROM 0000H–177FH 6016 x 8 bits	Mask ROM 0000H–1F7FH 8064 x 8 bits	EPROM/OTP 0000H–1F7FH 8064 x 8 bits	Mask ROM 0000H–2F7FH 12160 x 8 bits	Mask ROM 0000H–3F7FH 16256 x 8 bits	OTP & EPROM* 0000H–3F7FH 16256 x 8 bits
Data memory	512 x 4 bits						
3-byte branch instructions	Not included	Included	Included	Included	Included	Included	Included
Other instruction set	Common to the products						
Program counter	12 bit	13 bit	13 bit	13 bit	14 bit	14 bit	14 bit
Ports 4 and 5 pullup resistor	Mask option	Mask option	Mask option	None	Mask option	Mask option	None
LCD resistor ladder	Mask option	mask option	Mask option	Not included	Mask option	Mask option	Not included

Product Comparison (cont)

Item	µPD75304	µPD75306	µPD75308	µPD75P308	µPD75312	µPD75316	µPD75P316/A
V_{PP}, PROM programming pins	None	None	None	Included	None	None	Included
Operating voltage range	2.7 to 6.0 V	2.7 to 6.0 V	2.7 to 6.0 V	5 V ± 10%	2.7 to 6.0 V	2.7 to 6.0 V	5 V ± 10% 2.7 to 6.0 V*
Package	80-pin plastic QFP			80-pin plastic QFP 80-pin ceramic LCC with window	80-pin plastic QFP		80-pin plastic QFP 80-pin ceramic LCC w/window*

*µPD75P316A only.

ADDRESS SPACES AND MEMORY MAPS

The 75X architecture has two separate address spaces, one for program memory (ROM), and another for data memory (RAM).

Program Memory (ROM)

The ROM is addressed by the program counter. The size of the program counter is 12, 13, or 14 bits; its size depends on which member of the family is being used, as does the amount of ROM present. The ROM contains program object code, interrupt vector table, a GETI instruction reference table, and table data. Table data can be obtained using the table reference instruction, MOVT.

Figure 1 shows the addressing range which can be made using a branch instruction or subroutine call instruction. In addition, the BR PCDE and BR PCXA instructions can be used for a branch where only the low 8 bits of the PC are changed. The program memory addresses are:

75304: 000H to FFFH
75306: 0000H to 177FH
75308: 0000H to 1F7FH
75P308: 0000H to 1F7FH
75312: 0000H to 2F7FH
75316: 0000H to 3F7FH
75P316: 0000H to 3F7FH
75P316A: 0000H to 3F7FH

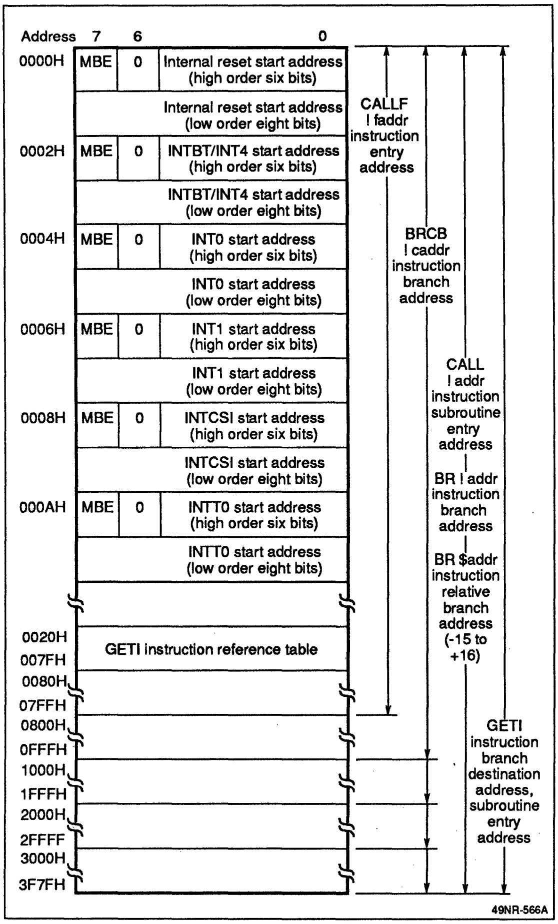
All locations if ROM except 0000H and 0001H can be used as program memory. However, if interrupts or GETI instructions are used, the locations corresponding to those functions cannot be used. Addresses are normally reserved as follows:

000H to 0001H: This address area is used as the vector address for $\overline{\text{RESET}}$, and also contains the MBE bit.

0002H to 000BH: This area is used for interrupt vector addresses. Each vector address contains an MBE bit value, and the interrupts can start from any location except where noted.

0020H to 007FH: This is the table area for GETI instructions. The GETI instruction is used to access one 2-byte/3-byte or two 1-byte instructions using one byte of program memory. This is useful in compacting code.

Figure 1. Program Memory Map


Address 7 6 0
0000H
MBE
0
Internal reset start address (high order six bits)
Internal reset start address (low order eight bits)
0002H
MBE
0
INTBT/INT4 start address (high order six bits)
INTBT/INT4 start address (low order eight bits)
0004H
MBE
0
INT0 start address (high order six bits)
INT0 start address (low order eight bits)
0006H
MBE
0
INT1 start address (high order six bits)
INT1 start address (low order eight bits)
0008H
MBE
0
INTCSI start address (high order six bits)
INTCSI start address (low order eight bits)
000AH
MBE
0
INTT0 start address (high order six bits)
INTT0 start address (low order eight bits)
0020H
007FH
GETI instruction reference table
0080H
07FFH
0800H
0FFFH
1000H
1FFFH
2000H
2FFFF
3000H
3F7FH
CALLF ! faddr instruction entry address
BRCB ! caddr instruction branch address
CALL ! addr instruction subroutine entry address
BR ! addr instruction branch address
BR $addr instruction relative branch address (-15 to +16)
GETI instruction branch destination address, subroutine entry address
49NR-566A

Program Counter (PC)

This is a 12/13/14-bit binary counter that contains the address of the current program memory location. The 75304 contains a 12-bit PC, the 75306/8 has a 13-bit PC, and the 75312/16 each contain a 14-bit PC.

When an instruction is executed, the PC is automatically incremented by the number of bytes of the current instruction. When a branch instruction (BR, BRCB) is executed, the contents of the immediate data or register pair indicating the new address are loaded into some or all the bits of the PC. When a subroutine call instruction (CALL, CALLF) is executed or an interrupt is generated, the PC is incremented to point to the next instruction, and this information is saved on the stack. During an interrupt, the program status word (PSW) is also automatically saved on the stack. The address to be jumped to by the CALL or interrupt is then loaded into the PC.

When a return instruction (RET, RETS, or RETI) is executed, the contents of the stack are restored to the PC.

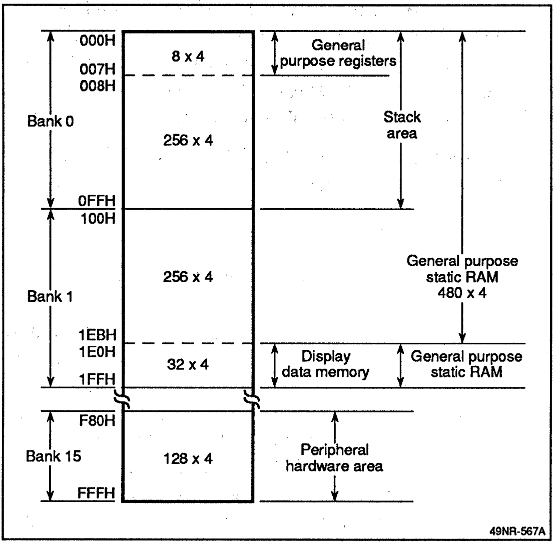
Data Memory (RAM)

The data memory contains three memory banks, 0, 1, and 15. The RAM memory map is shown in figure 2. The memory consists of general purpose static RAM and peripheral control registers, and accessed by using the MBE (memory bank enable) and by programming the BS (bank select register). If MBE = 0, the lower 128 nibbles of memory bank 0 and the upper 128 nibbles of memory bank 15 are accessed. If MBE = 1, the upper four bits in the BS register will specify the memory bank. The values are 0H for memory bank 0, 1H for memory bank 1, and 0FH for memory bank 15. Memory banks 0 and 1 each contain 256 nibbles; while the memory is organized in nibbles, the 75X architecture allows the data to be manipulated in bytes, nibbles and individual bits.

The data memory is used for storing processed data, general purpose registers, and as a stack for subroutine or interrupt service. The last 32 nibbles of bank 1 are used to store the LCD display data. If this area is not completely used by the LCD, it may be used as general-purpose RAM. Because of its static nature, the RAM will retain its data when CPU operation is stopped and the chip is in the standby mode, provided V_{DD} is at least 2 volts.

There are eight 4-bit general-purpose registers in bank 0 starting at location 00H. These registers may also be used as four 8-bit registers. The on-chip peripheral control registers and ports reside in the upper 128 nibbles of bank 15. Bank 15 addresses which are not assigned to a register are not available as random memory except for the 16-bit sequential buffer. Also, the lower 128 nibbles of bank 15 do not contain RAM.

Figure 2. Data Memory Map



Addressing Modes

The μPD7530x/31x is able to address data memory and ports as individual bits, nibbles, or bytes. The addressing modes are as follows:

1-bit direct data memory
4-bit direct data memory
4-bit register indirect (@rpa)
8-bit direct data memory
8-bit register indirect (@HL)

See table 1 for data memory addressing and table 2 for peripheral control register addressing.

Table 1. Data Memory Addressing Modes

Addressing Mode	Representation Format	How the Address is Created
1-bit direct addressing	mem.bit	If MBE = 0, the memory bank is Bank 0 for addresses 00H-7FH, and Bank 15 for addresses 80H-FFH. If MBE = 1, the memory bank is selected by the four bits of the MBS. The bit to be manipulated is specified in mem.bit
4-bit direct addressing	mem	If MBE = 0, the memory bank is Bank 0 for addresses 00H-7FH, and Bank 15 for addresses 80H-FFH. If MBE = 1, the memory bank is selected by the four bits of the MBS. The bit to be manipulated is specified in mem.
8-bit direct addressing	mem (must be an even address)	If MBE = 0, the memory bank is Bank 0 for addresses 00H-7FH, and Bank 15 for addresses 80H-FFH. If MBE = 1, the memory bank is selected by the four bits of the MBS. The bit to be manipulated is specified in mem.
4-bit register indirect addressing	@HL	The memory bank is selected by the four bits of the MBS, and the location within the memory bank is contained in register HL.
	@DE	The memory bank is always Bank 0, and the location within the memory bank is contained in register DE
	@DL	The memory bank is always Bank 0, and the location within the memory bank is contained in register DL
8-bit register indirect addressing	@HL (must be an even address)	The memory bank is selected by the four bits of the MBS, and the location within the memory bank is contained in register HL.

Table 1. Data Memory Addressing Modes

Addressing Mode	Representation Format	How the Address Is Created

Bit manipulation addressing	fmem.bit	The memory bank is Bank 15, and the location is fmem, where
		fmem = FB0H-FBFH for interrupts
		fmem = FF0H-FFFH I/O ports
		The actual bit is specified in fmem.bit

	pmem.@L	The memory location is independent of MBE and MBS. The upper 10 address bits of the location are contained in the ten high order bits of pmem and the two lower address bits are contained in the two upper bits of register L. The bit to be manipulated is specified by the two LSBs of register L.
	@H + mem.bit	The memory bank is selected by the four bits of the MBS, and the location is determined by the following: The four upper bits are the contents of register H The four lower bits are mem. The actual bit is specified in mem.bit.
Stack addressing		The memory bank is always Bank 0, and the location is indicated by the stack pointer (SP)

MBE: memory bank enable bit
MB: memory bank
MBS: memory bank select register
mem: a location within a memory bank
mem.bit: a bit at a specified memory location.
fmem and pmem are specialized cases of mem.

4

Table 2. Addressing Modes During Peripheral Hardware Operation

Manipulation	Addressing Mode	Applicable Hardware
1-bit	With MBE = 0 (or MBE = 1 and MBS = 15) direct addressing (specification in mem.bit)	All hardware where bit manipulation can be performed
	Direct addressing regardless of how MBE and MBS are set. (specification in fmem.bit)	ISTO, MBE IExxx, IRQxxx, PORTn.x
	Indirect addressing regardless of how MBE and MBS are set. (specification in pmem. @L)	BSBn.x PORTn.x
4-bit	With MBE = 0 (or MBE = 1 and MBS = 15) direct addressing (specification in mem.bit)	All hardware where 4-bit manipulation can be performed
	With MBE = 1 and MBS = 15, register indirect addressing (specification in @HL)	
8-bit	With MBE = 0 (or MBE = 1 and MBS = 15) direct addressing (specification in mem); mem must be an even address	All hardware where 8-bit manipulation can be performed
	With MBE = 1 and MBS = 15, register indirect addressing (specification in @HL); L register must contain an even number	

Instruction Execution Times

The minimum instruction execution time is 0.95 µs with a 4.19 MHz clock. The PCC register can be used to program the CPU's minimum instruction cycle time to 0.95, 1.91, or 15.3 µs; all three speeds presuppose a 4.19 MHz crystal. Reducing the CPU clock speed will reduce the microprocessor's power consumption.

Instruction Set

The instruction set contains the following features:

- Versatile bit manipulation instructions
- Efficient 4-bit manipulation instructions
- 8-bit data transfer instructions
- GETI instruction to reduce program size
- Vertically stored instructions and base correction instructions
- Table reference instructions
- 1-byte relative branch instructions

Symbol Definitions

The µPD7530x/31x family uses the following symbol definitions:

Symbol	Definition
A	A register; 4-bit accumulator
B	B register; 4-bit accumulator
C	C register; 4-bit accumulator
D	D register; 4-bit accumulator
E	E register; 4-bit accumulator
H	H register; 4-bit accumulator
L	L register; 4-bit accumulator
X	X register; 4-bit accumulator
XA	XA register pair; 8-bit accumulator
BC	BC register pair
DE	DE register pair
DL	DL register pair
HL	HL register pair
PC	Program counter
SP	Stack pointer
CY	Carry flag; bit accumulator
PSW	Program status word
MBE	Memory bank enable flag
PORTn	Port n (n = 0-7)
IME	Interrupt master enable
IExxx	Interrupt enable flag
MBS	Memory bank selection register
PCC	Clock processor control register
.	Separation between address and bit
(xx)	The contents addressed by xx
xxH	Hexadecimal data

Operation Code Symbols

The following opcode symbols are used with the µPD7530x/31x family.

reg, reg1

R_2	R_1	R_0	Register
0	0	0	A (reg only)
0	0	1	X (reg, reg1)
0	1	0	L (reg, reg1)
0	1	1	H (reg, reg1)
1	0	0	E (reg, reg1)
1	0	1	D (reg, reg1)
1	1	0	C (reg, reg1)
1	1	1	B (reg, reg1)

@rpa, @rpa1

Q_2	Q_1	Q_0	Addressing
0	0	1	@HL (@rpa only)
1	0	0	@DE (@rpa, @rpa1)
1	0	1	@DL (@rpa, @rpa1)

N_5	N_2	N_1	N_0	IExxx
0	0	0	0	IEBT
0	0	1	0	IEW
0	1	0	0	IETO
0	1	0	1	IECSI
0	1	1	0	IE0
0	1	1	1	IE2
1	0	0	0	IE4
1	1	1	0	IE1

register pairs

P_2	P_1	reg-pair	rp	rp1	rp2
0	0	XA	x	–	–
0	1	HL	x	x	–
1	0	DE	x	x	x
1	1	BC	x	x	x

Operation Representation Format and Description Method

An operand is entered in the operand field of each instruction according to the format of the instruction (see assembler specifications). When two or more entries are indicated in the description method, one should be selected. Capital letters and symbols must be entered exactly as shown. For immediate data, a proper numeric value or label should be entered.

Table 3. Symbol Abbreviations

Symbol	Description
reg	X, A, B, C, D, E, H, L
reg1	X, B, C, D, E, H, L
rp	XA, BC, DE, HL
rp1	BC, DE, HL
rp2	BC, DE
rpa	HL, DE, DL
rpa1	DE, DL
n4	4-bit immediate data or label
n8	8-bit immediate data or label
mem (Note 1)	8-bit immediate data or label
bit	2-bit immediate data or label
fmem	FB0H-FBFH, FF0H-FFFH immediate data or label
pmem	FC0H-FFFH immediate data or label
addr, caddr	μPD75304: 000H-FFFH immediate data or label
	μPD75306: 0000H-177FH immediate data or label
	μPD75308/P308: 0000H-1F7FH immediate data or label
	μPD75312: 0000H-2F7FH immediate data or label
	μPD75316/P316: 0000H-3F7F immediate data or label
faddr	11-bit immediate data or label
taddr	20H-7FH immediate data (where bit 0 = 0) or label
PORTn	Port 0-Port 7
IExxx	IEBT, IECSI, IET0, IE0-IE4, IEW
MBn	MB0, MB1, MB15

Notes:

(1) Memory address must be an even number in 8-bit processing.

Instruction Set

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Transfer					
MOV	A, #n4	1	1	A ← n4	String A
	reg1, #n4	2	2	reg1 ← n4	
	XA, #n8	2	2	XA ← n8	String A
	HL, #n8	2	2	HL ← n8	String B
	rp2, #n8	2	2	rp2 ← n8	
	A, @HL	1	1	A ← (HL)	
	A, @rpa1	1	1	A ← (rpa1)	
	XA, @HL	2	2	XA ← (HL)	
	@HL, A	1	1	(HL) ← A	
	@HL, XA	2	2	(HL) ← XA	
	A, mem	2	2	A ← (mem)	
	XA, mem	2	2	XA ← (mem)	
	mem, A	2	2	(mem) ← A	
	mem, XA	2	2	(mem) ← XA	
	A, reg1	2	2	A ← (reg1)	
	XA, rp	2	2	XA ← rp	
	reg1, A	2	2	reg1 ← A	
	rp1, XA	2	2	rp1 ← XA	

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Transfer (cont)					
XCH	A, @HL	1	1	A ↔ (HL)	
	A, @rpa1	1	1	A ↔ (rpa1)	
	XA, @HL	2	2	XA ↔ (HL)	
	A, mem	2	2	A ↔ (mem)	
	XA, mem	2	2	XA ↔ (mem)	
	A, reg1	1	1	A ↔ (reg1)	
	XA, rp	2	2	XA ↔ rp	
MOVT	XA, @PCDE	1	3	XA ← $(PC_{12\text{-}8}+DE)_{ROM}$	
	XA, @PCXA	1	3	XA ← $(PC_{12\text{-}8}+XA)_{ROM}$	
Arithmetic					
ADDS	A, #n4	1	1+S	A ← A+ n4	Carry
	A, @HL	1	1+S	A ← A+ (HL)	Carry
ADDC	A, @HL	1	1	A, CY ← A+ (HL)+ CY	
SUBS	A, @HL	1	1+S	A ← A–(HL)	Borrow
SUBC	A, @HL	1	1	A, CY ← A – (HL) – CY	
AND	A, #n4	2	2	A←A $\wedge$ n4	
	A, @HL	1	1	A←A $\wedge$ (HL)	
OR	A, #n4	2	2	A ← A $\vee$ n4	
	A, @HL	1	1	A ←A $\vee$ (HL)	
XOR	A, #n4	2	2	A ← A **XOR** n4	
	A, @HL	1	1	A ← A **XOR** (HL)	
Accumulator Manipulation					
RORC	A	1	1	CY ← A_0, A_3 ← CY, A_{n-1} ← A_n	
NOT	A	2	2	A ← $\overline{A}$	
Increment/Decrement					
INCS	reg	1	1+S	reg ← reg+1	reg = 0
	@HL	2	2+S	(HL) ← (HL)+1	(HL) = 0
	mem	2	2+S	(mem) ← (mem)+1	(mem) = 0
DECS	reg	1	1+S	reg ← reg–1	reg = FH
Comparison					
SKE	reg, #n4	2	2+S	skip if reg = n4	reg = n4
	@HL, #n4	2	2+S	skip if (HL) = n4	(HL) = n4
	A, @HL	1	1+S	skip if A = (HL)	A = (HL)
	A, reg	2	2+S	skip if A = reg	A = reg
Carry Flag Manipulation					
SET1	CY	1	1	CY ← 1	
CLR1	CY	1	1	CY ← 0	
SKT	CY	1	1+S	skip if CY = 1	CY = 1
NOT1	CY	1	1	CY ← $\overline{CY}$	

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Memory Bit Manipulation					
SET1	mem.bit	2	2	(mem.bit) ← 1	
	fmem.bit	2	2	(fmem.bit) ← 1	
	pmem.@L	2	2	(pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$)) ← 1	
	@H+mem.bit	2	2	(H +mem$_{3-0}$.bit) ← 1	
CLR1	mem.bit	2	2	(mem.bit) ← 0	
	fmem.bit	2	2	(fmem.bit) ← 0	
	pmem.@L	2	2	(pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$)) ← 0	
	@H+mem.bit	2	2	(H +mem$_{3-0}$.bit) ← 0	
SKT	mem.bit	2	2+S	skip if (mem.bit) = 1	(mem.bit) = 1
	fmem.bit	2	2+S	skip if (fmem.bit) = 1	(fmem.bit) = 1
	pmem.@L	2	2+S	skip if (pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$)) = 0	(pmem.@L = 1)
	@H+mem.bit	2	2+S	skip if (H+mem$_{3-0}$.bit) = 1	(@H+mem.bit) = 1
SKF	mem.bit	2	2+S	skip if (mem.bit) = 0	(mem.bit) = 0
	fmem.bit	2	2+S	skip if (fmem.bit) = 0	(fmem.bit) = 0
	pmem.@L	2	2+S	skip if (pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$)) = 0	(pmem.@L = 0)
	@H+mem.bit	2	2+S	skip if (H+mem$_{3-0}$.bit) = 0	(@H+mem.bit) = 0
SKTCLR	fmem.bit	2	2+S	skip if (fmem.bit) = 1 and clear	(fmem.bit) = 1
	pmem.@L	2	2+S	skip if (pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$)) = 1 and clear	(pmem.@L = 1)
	@H+mem.bit	2	2+S	skip if (H+mem$_{3-0}$.bit) = 1 and clear	(@H+mem.bit) = 1
AND1	CY, fmem.bit	2	2	CY ← CY ∧ (fmem.bit)	
	CY, pmem.@L	2	2	CY ← CY ∧ (pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$))	
	CY, @H+mem.bit	2	2	CY ← CY ∧ (H+mem$_{3-0}$.bit)	
OR1	CY, fmem.bit	2	2	CY ← CY ∨ (fmem.bit)	
	CY, pmem.@L	2	2	CY ← CY ∨ (pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$))	
	CY, @H+mem.bit	2	2	CY ← CY ∨ (H+mem$_{3-0}$.bit)	
XOR1	CY, fmem.bit	2	2	CY ← CY **XOR** (fmem.bit)	
	CY, pmem.@L	2	2	CY ← CY **XOR** (pmem$_{7-2}$+L$_{3-2}$.bit(L$_{1-0}$))	
	CY, @H+mem.bit	2	2	CY ← CY **XOR** (H+mem$_{3-0}$.bit)	
Branch					
BR (Note 1)	addr	–	–	PC$_{12-0}$ ← addr	
	!addr	3	3	PC$_{12-0}$ ← addr	
	$addr	1	2	PC$_{12-0}$ ← addr	
BRCB	!caddr	2	2	PC$_{12-0}$ ← PC$_{12-0}$+caddr$_{11-0}$	
Subroutine Stack Control					
CALL	!addr	3	3	(SP−4)(SP−1)(SP−2) ← PC$_{11-0}$ (SP−3) ← (MBE, 0, 0, PC$_{12}$) PC$_{12-0}$ ← addr, SP ← (SP−4)	
CALLF	!faddr	2	2	(SP−4)(SP−1)(SP−2) ← PC$_{11-0}$ (SP−3) ← (MBE, 0, 0, PC$_{12}$) PC$_{12-0}$ ← 00, faddr, SP ← (SP−4)	

Instruction Set (cont)

Mnemonic	Operand	Bytes	Machine Cycle	Operation	Skip Condition
Subroutine Stack Control (cont)					
RET		1	3	(MBE, PC$_{12}$) ← (SP+1) PC$_{11-0}$ ← (SP)(SP+3)(SP+2) SP ← (SP+4)	
RETS		1	3+S	(MBE, PC$_{12}$) ← (SP+1) PC$_{11-0}$ ← (SP)(SP+3)(SP+2) SP ← (SP+4), then skip unconditionally	Unconditional
RETI		1	3	(PC$_{12}$) ← (SP+1) PC$_{11-0}$ ← (SP)(SP+3)(SP+2) PSW ← (SP+4)(SP+5), SP ← (SP+6)	
PUSH	rp	1	1	(SP−1)(SP−2) ← rp, SP ← (SP−2)	
	BS	2	2	(SP−1) ← MBS, (SP−2) ← 0,SP ← (SP−2)	
POP	rp	1	1	rp ← (SP+1)(SP), SP ← (SP+2)	
	BS	2	2	MBS ← (SP+1), SP ← (SP+2)	
Interrupt Control					
EI		2	2	IME ← 1	
	IExxx	2	2	IExxx ← 1	
DI		2	2	IME ← 0	
	IExxx	2	2	IExxx ← 0	
Input/Output (Note 2)					
IN	A, PORT$_n$	2	2	A ← PORT$_n$; (n = 0 to 7)	
	XA, PORT$_n$	2	2	XA ← PORT$_{n+1}$, PORT$_n$; (n = 4, 6)	
OUT	PORT$_n$, A	2	2	PORT$_n$ ← A; (n = 2 to 7)	
	PORT$_n$, XA	2	2	PORT$_{n+1}$, PORT$_n$ ← XA; (n = 4, 6)	
CPU Control					
HALT		2	2	Set HALT mode (PCC.2 ← 1)	
STOP		2	2	Set STOP mode (PCC.3 ← 1)	
NOP		1	1	No operation	
Special					
SEL	MB$_n$	2	2	MBS ← n; (n = 0, 1, 15)	
GETI	taddr	1	3	When (taddr)$_{7-6}$ = 00, PC$_{12-0}$ ← (taddr)$_{4-0}$ + (taddr+1) When (taddr)$_{7-6}$ = 01, (SP−4)(SP−1)(SP−2) ← PC$_{11-0}$; (SP−3) ← (MBE,0,0,PC$_{12}$); PC$_{12-0}$ ← (taddr)$_{4-0}$ + (taddr+1); SP ← SP−4 When (taddr)$_{7-6}$ = 10, (taddr), (taddr+1) instructions are executed.	Depends on the referenced instruction

Notes:

(1) Appropriate instructions are selected from BR !addr, BRCB !caddr, and BR $saddr by the assembler.

(2) When executing the IN/OUT instruction, either MBE must be reset to 0, or MBE and MBS must be set to 1 and 15, respectively.

Input/Output Ports

There are eight 4-bit ports; some are I/O ports and some are input only. Figure 3 shows the structure of the ports and table 4 lists the features. Figure 3 also shows the structure of inputs and outputs of the other pins.

Table 4. Types and Features of Digital Ports

Port	Function	Operation and Features	Remarks
PORT 0	4-bit input	Can always be read or tested regardless of the operation mode.	Pins also used for INT4, $\overline{\text{SCK}}$ SO/SB0, SI/SB1.
PORT 1			Pins also used for INT0-2 and TI0.
PORT 3 (Note 1)	4-bit input/output	Can be placed in input or output mode in 1-bit units.	Pins also used for LCDCL, SYNC and MD0-MD3, (Note 2)
PORT 6			Pins also used for KR0-KR3.
PORT 2	4-bit input/output	Can be placed in input or output mode in 4-bit units. Ports 6 and 7 can be paired for data input/output in 8-bit units.	Port 2 pins are also used for PTO0, PCL and BUZ.
PORT 7			Pins also used for KR4-KR7.
PORT 4 (Note 1) PORT 5 (Note 1)	4-bit input/output (N-channel open drain, 10 volts)	Can be placed in input or output mode in 4-bit units. Ports 4 and 5 can be paired for data input/output in 8-bit units.	Internal pull-up resistor can be specified in 1-bit units by using mask option.
BP0-BP7	1-bit output	Data is output in 1-bit units. The BP0-BP7 pins are also used as LCD segment pins S24-S31. BP0-BP7 and S24-S31 can be changed by using software.	The capacity of drive is very small. Used for CMOS load drive.

Notes:

(1) These ports directly drive LEDs.

(2) PORT 3 lines are also used for MD0-MD3 in μPD75P308/P316/P316A only.

Figure 3. I/O Circuits

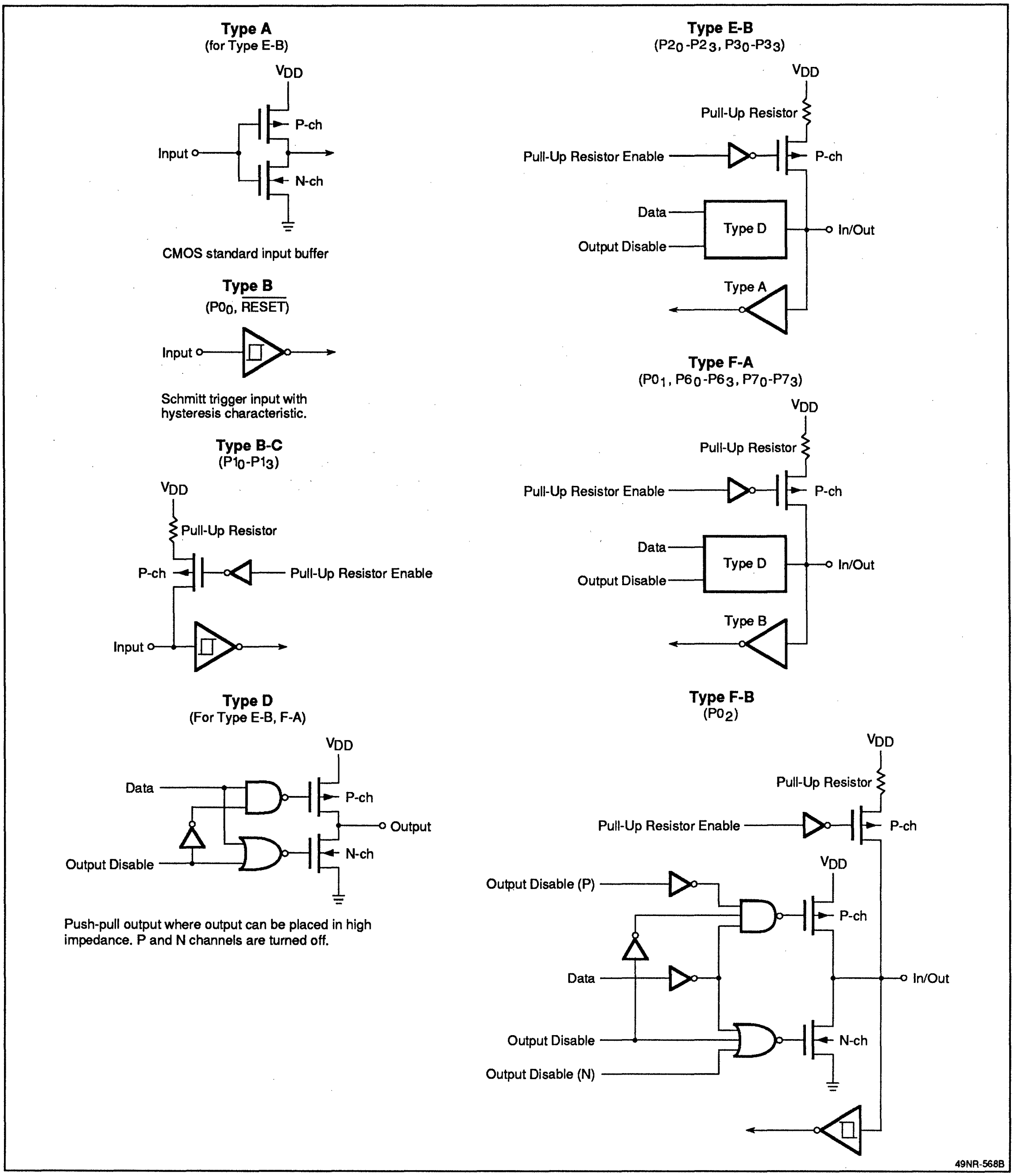
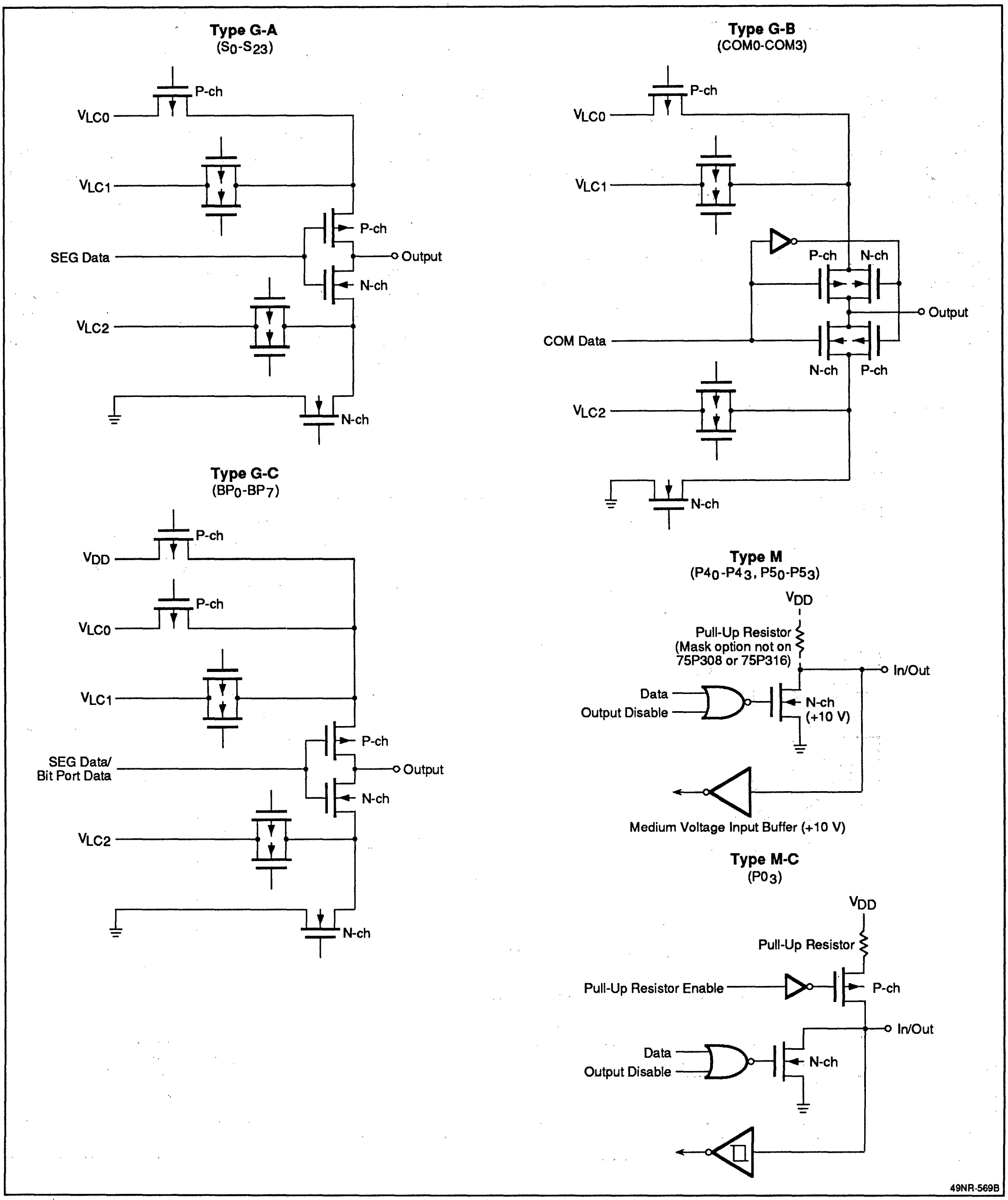


Figure 3. I/O Circuits (cont)



Clock Generator

The clock generator (figure 4) uses the crystal inputs X1 and X2 as a time base to provide clocks for the µPD7530x/31x. The generator consists of an oscillator, frequency dividers, multiplexers, and three control registers, PCC, SCC, and CLOM. By programming PCC and CLOM, frequencies derived from the crystal are supplied to the CPU, the interval timer, the timer/event counter, the watch timer, the serial interface, and the output pin, PCL.

The PCC and SCC registers control the HALT and STOP logic and can also be used to set the CPU to operate at one of four speeds. The CLOM register controls the output clock PCL.

The µPD7530x/31x family also contains a subsystem clock, consisting of an oscillator driven by an external crystal. It operates at 32-35 kHz, and can be used as a clock source to the watch timer and the CPU.

Basic Interval Timer

The basic interval timer (figure 5) is used to provide continuous real-time interrupts. It consists of a multiplexer, an 8-bit free-running counter, and a 4-bit BTM control register. Each time the counter reaches FFH it causes an interrupt, overflows to 00H and continues to count. The BTM register is used to select one of four clock inputs to the counter as well as clear the counter and its interrupt request. The counter can generate 250 ms interrupts with a 4.19 MHz crystal and also provides oscillator stabilization time when the chip comes out of the STOP mode.

Figure 4. Clock Generator

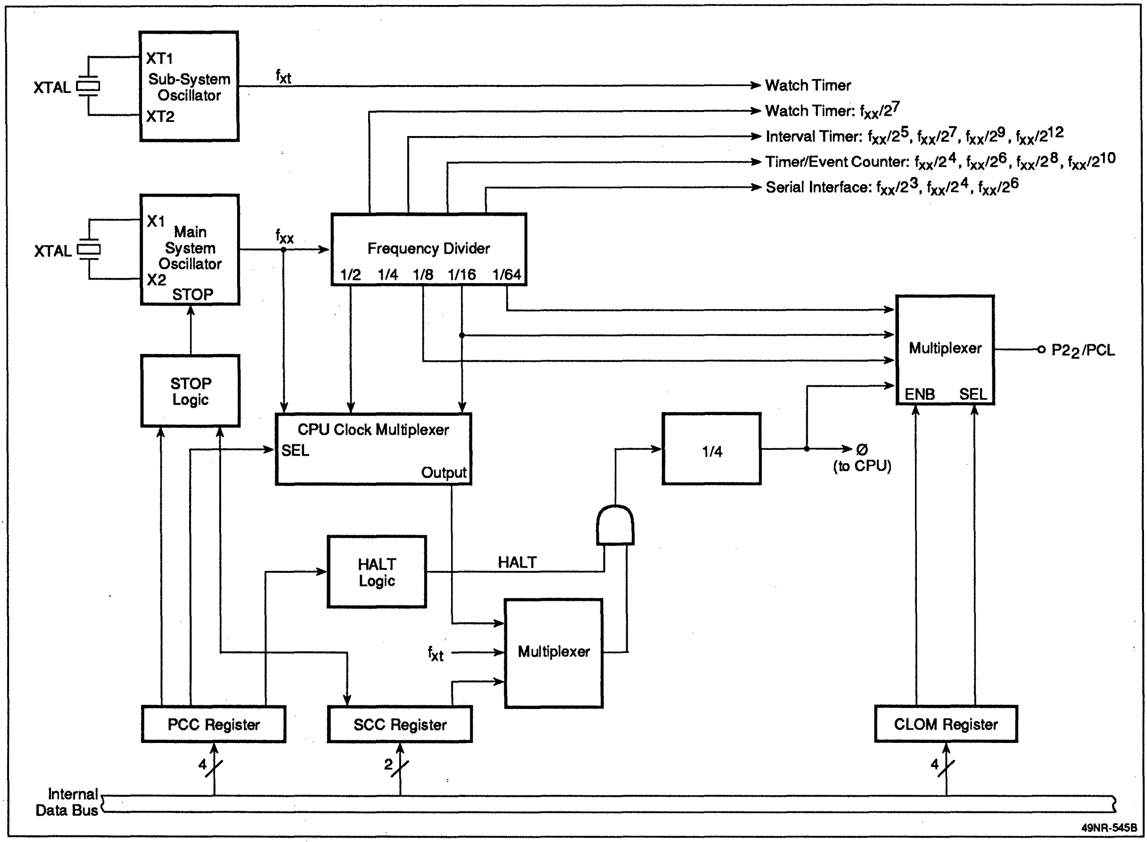
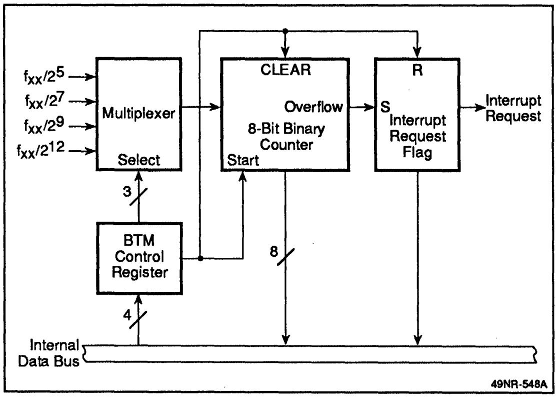


Figure 5. Basic Interval Timer

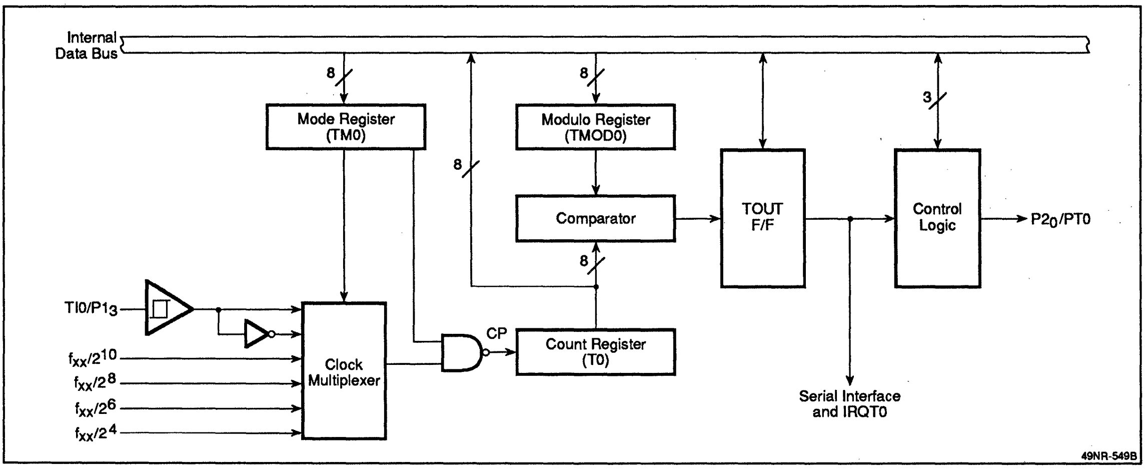


Timer/Event Counter (TM0)

The timer/event counter (figure 6) consists of an 8-bit modulo register, 8-bit comparator, 8-bit count register, clock multiplexer, mode control register TM0, and a TOUT flip flop. There is also some control logic so that the timer's TOUT flip flop can be sent to port 2.

An 8-bit value is loaded into the modulo register, and a count register clock is selected by the clock multiplexer, via control register TM0. The count register is incremented each time it receives a CP pulse. When the value in the count register is equal to the count in the modulo register, the comparator generates a signal which toggles the TOUT flip flop and causes the count register to be reset to 00H. The count register will continue to count up unless stopped. Each time TOUT changes state it causes an interrupt. This signal can also be used as a clock for the serial interface.

Figure 6. Timer/Event Counter



Watch Timer

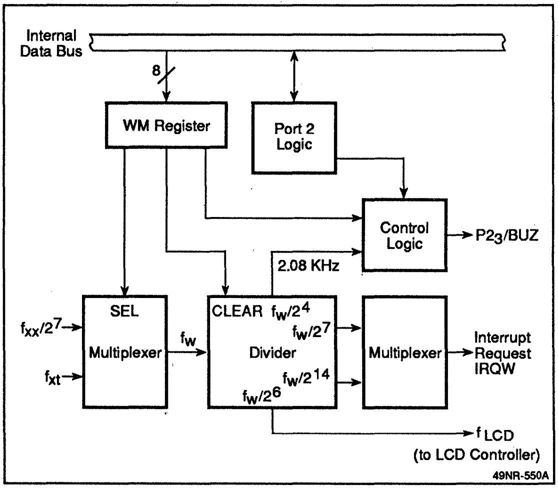
The watch timer (figure 7) generates interrupt requests (but no interrupts) at 0.5 second intervals when using a 4.19 MHz crystal. It is commonly used as a time source for keeping track of the time of day, can operate in the STOP mode and is capable of generating a 2 kHz buzzer output signal.

The watch timer consists of an input multiplexer, divider, output multiplexer, control logic, and control register WM. It is also used as a clock source for the LCD controller.

Serial Interface

The 8-bit serial interface (figure 8) allows the μPD7530x/31x to communicate with other NEC or NEC-like serial interfaces. It consists of an 8-bit shift register (SIO), serial-out latch (SO), 8-bit address comparator, slave address register (SVA), control registers CSIM and SBIC, busy/acknowledge circuitry, bus release/detect circuitry, serial clock counter, clock multiplexer, and clock control circuitry. The three-wire interface consists of the serial data in (SI/SB1), serial data out (SO/SB0), and serial shift clock ($\overline{\text{SCK}}$).

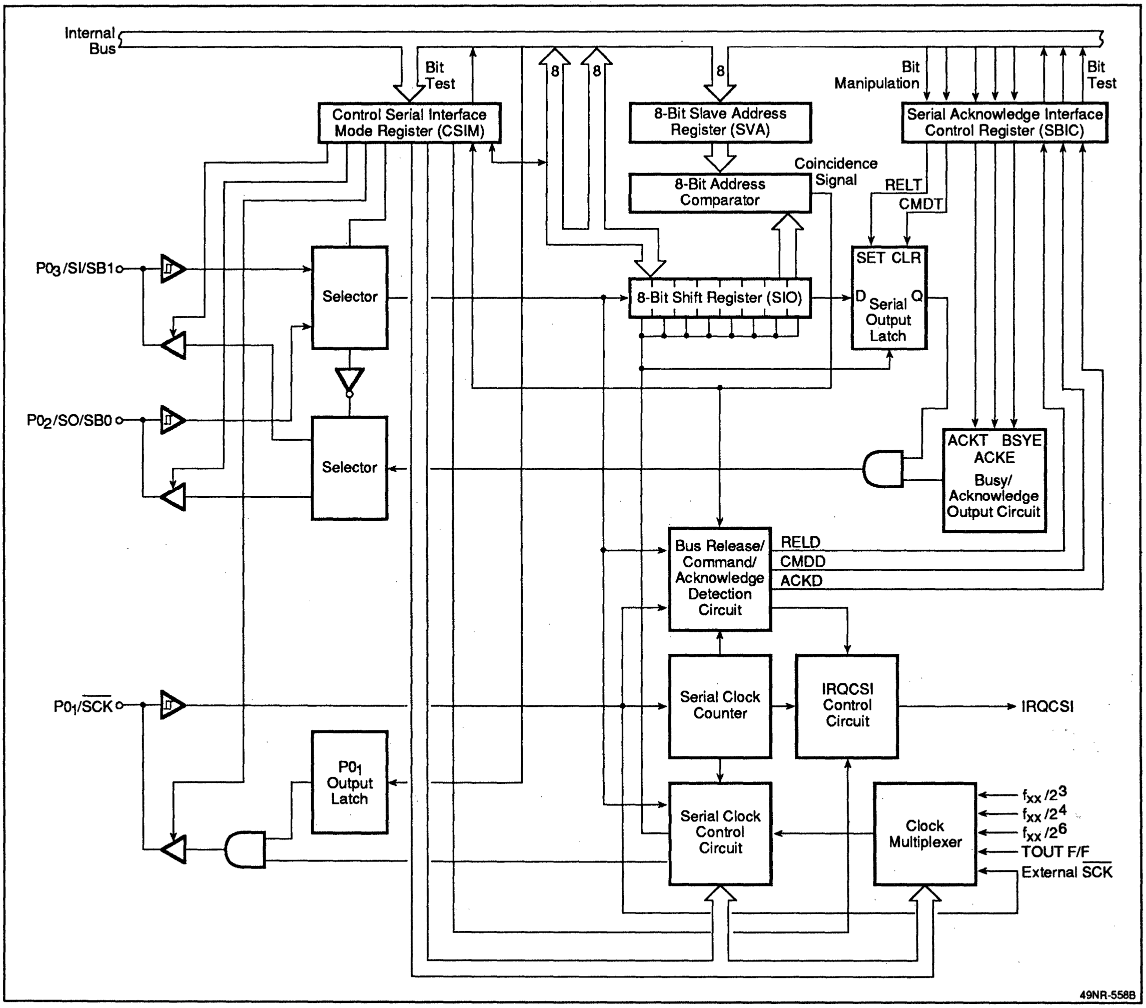
Figure 7. Watch Timer



There are three modes of operation, 2-wire serial, 3-wire serial, and 2-wire SBI. The simplest modes are the 2/3-wire serial. In these modes, the 8-bit shift register is loaded with a byte of data and 8 clock pulses are generated. These pulses shift data out the SO line and data in from the SI line, thus, communicating in full duplex. Each time a byte of data is sent, a burst of eight clock pulses is generated and eight bits of data will be sent. Data may be sent either LSB or MSB first. The interface may also be set to receive data only; in this case SO is in the high-impedance state. One of four internal clocks or an external clock may be used to clock the data.

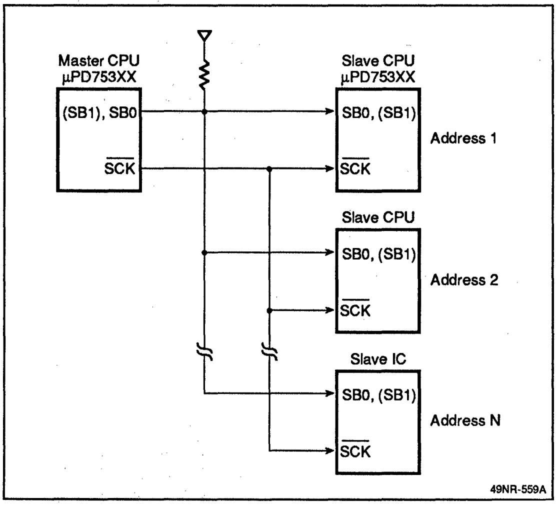
The SBI mode uses a 2-wire interface (figure 9) with devices in a master/slave configuration. At any one time, there is a single master, with all other devices being slaves. The master can send addresses, commands, and data over the bus. The slaves are able to detect in hardware if their particular address has been sent, and can also detect whether a command or piece of data has been sent. There can be as many as 256 slave addresses, 256 commands, and 256 data types. All commands are user-defined, and it is possible to send commands which change slaves into masters; when this happens, the previous master becomes a slave. This type of work is done in firmware, and the bus can be as simple or complex as the user wishes.

Figure 8. Serial Interface Block Diagram



4

Figure 9. ***SBI Mode Master/Slave Configuration***



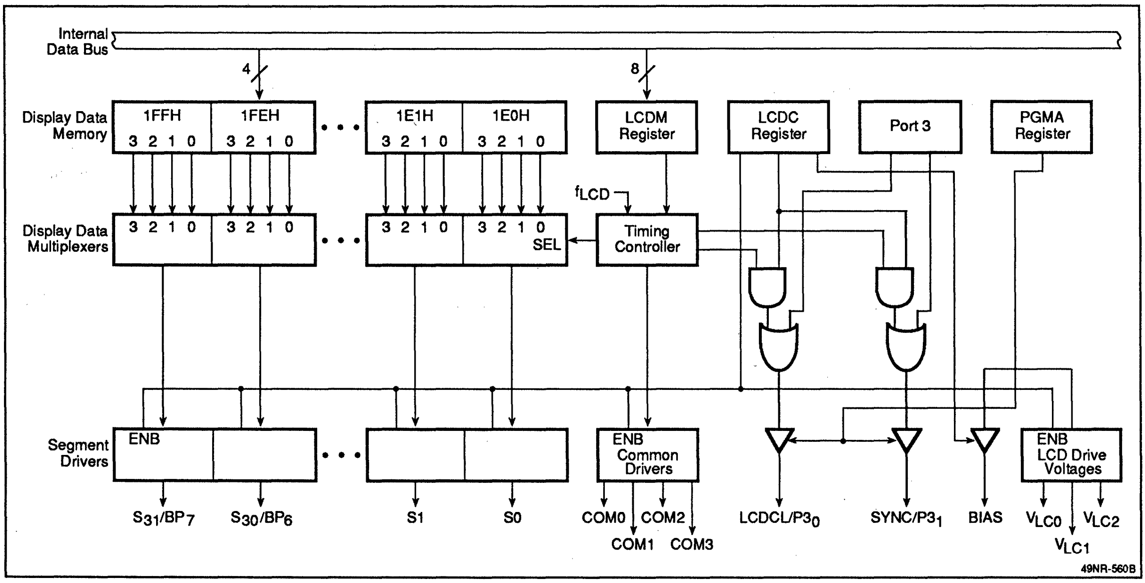
LCD Controller/Driver

The LCD controller/driver (figure 10) can be programmed to operate in any of four modes. It can operate in the static mode (drive 32 segments), the multiplexed mode (drive 64 segments), the triplexed mode (drive 96 segments), or quadruplexed mode (drive 128 segments). The multiplexed mode uses 1/2 bias, the triplexed mode can use either 1/2 or 1/3 bias, and the quadruplexed mode uses 1/3 bias.

The controller automatically refreshes the LCD by taking data from the upper 32 nibbles of RAM memory bank 1, and uses display data multiplexers, segment drivers S0–S31, and common drivers COM0–COM3 to drive the LCD. It is controlled by registers LCDM, LCDC, and PGMA. The LCD main controller clock (f_{LCD}) is provided by the watch timer. Because the watch timer operates while the chip is in the STOP mode, so does the LCD controller.

The SYNC signal and clock LCDCL are provided so that additional LCD controllers can be added. Drive levels can be set internally by ordering the resistor ladder mask option, otherwise, external resistors can be connected to pins V_{LC0}–V_{LC2} and the BIAS pin. The BIAS pin can be used to control the contrast of the LCD.

Figure 10. ***LCD Controller Block Diagram***



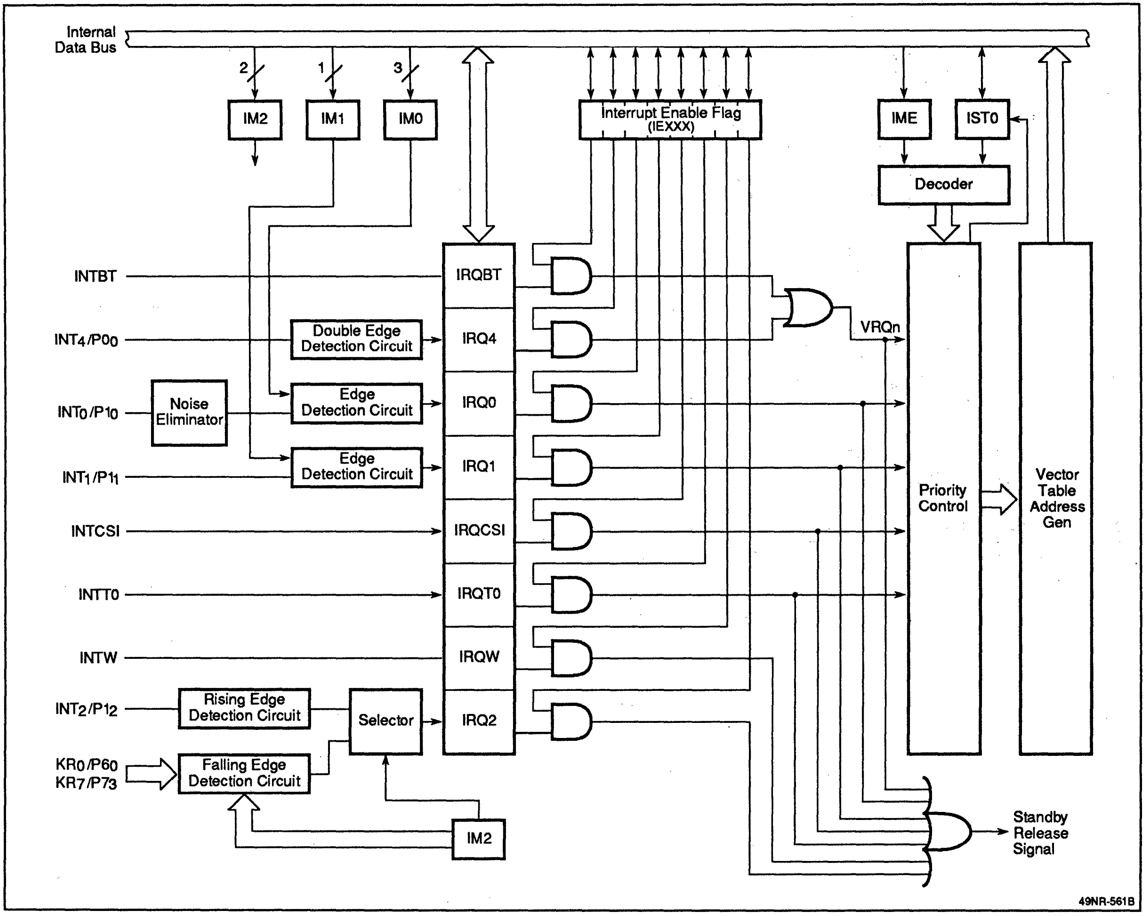
Bit Sequential Buffer

The bit sequential buffer is 16 bits of general-purpose RAM located in the upper half of memory bank 15, and is the only general-purpose RAM in this area. All other locations in this bank contain either the on-chip peripheral control registers or are unused addresses. A typical application of this buffer might be to store data for the next serial output or to store data from a serial input. It could also be used to store data which is to be sent from a port. This area can be bit, nibble, or byte manipulated.

Interrupts

The µPD7530x/31x family interrupts (figure 11) are all vectored; there are three external and three internal interrupts. Table 5 gives a summary of the interrupts. In addition, INT2 will sense the rising edge inputs and generate an interrupt request flag which is testable. Inputs KR0–KR7 will detect falling edges, and generate the same interrupt request flag as INT2. Neither INT2 nor KR0–KR7 will cause an interrupt, but they can be used to release the STANDBY mode. All interrupts and interrupt requests except INT0 will release the STANDBY mode.

Figure 11. Interrupt Controller Block Diagram



Standby Modes

The standby mode is summarized in table 6 and consists of three submodes.

HALT mode. The HALT mode is entered by executing the HALT instruction. In this mode, the clock to the CPU is shut off (thus stopping the CPU), while all other parts of the chip, with the exception of INT0, remain fully functional.

STOP mode. The STOP mode is entered by executing the STOP instruction. In this mode, the chip's main system oscillator is shut off, thereby stopping all portions of the chip except those which function off the subsystem clock. If the subsystem clock is used, it always remains on.

The HALT and STOP modes are released by a $\overline{\text{RESET}}$ or by any interrupt request except INT0.

Data Retention mode. This mode may be entered after entering the STOP mode. Here, supply voltage V_{DD} may be lowered to 2 volts to further reduce power consumption. The contents of the RAM and registers are retained. This mode is released by first raising V_{DD} to the proper operating range, then releasing the STOP mode.

Table 5. Interrupt Sources

Interrupt Source	Operation	Internal/ External	Interrupt Priority (Note)	Vectored Interrupt Request Signal (Vector Table Address)
INTBT	Reference time interval signal from basic interval timer	Internal	1	VRQ1 (0002H)
INT4	Both rising and falling edge detection	External	1	VRQ1 (0002H)
INT0	Selection of rising or falling edge detection	External	2	VRQ2 (0004H)
INT1	Selection of rising or falling edge detection	External	3	VRQ3 (0006H)
INTCSI	Serial data transfer end signal	Internal	4	VRQ4 (0008H)
INTT0	Coincidence signal between programmable timer/counter count register and modulo register	Internal	5	VRQ5 (000AH)
INT2	Rising edge detection of input to INT2 pin, or falling edge detection of any input to KR0–KR7	External	Testable input signals (IRQ2 and IRQW are set)	
INTW	Signal from watch timer	Internal		

Notes:

(1) The interrupt priority order is used to determine the priority when two or more interrupts are generated simultaneously.

Table 6. Standby Mode Operation

Setting Instruction	STOP Instruction	HALT Instruction
System clock when standby mode is set	Can be set only during main system or subsystem clock	Can be set during either main system or subsystem clock
Clock oscillator	Only the main system clock oscillator is stopped	Only CPU clock ϕ is stopped (oscillation continues)
Basic interval timer	Operation stopped	Can Operate
Serial interface	Can operate only when external $\overline{\text{SCK}}$ input is selected for serial clock	Can operate
Timer/event counter	Can operate only when TIO pin input is selected for count clock	Can operate
Watch timer	Can operate only when f_{XT} is selected for count clock	Can operate

Table 6. Standby Mode Operation (cont)

Setting Instruction	STOP Instruction	HALT Instruction
LCD controller	Can operate only when f_{XT} is selected for LCDCL	Can operate
External interrupts	INT1, INT2, INT4 can operate; INT0 cannot	
CPU	Operation stop	
Release signal	Interrupt request signal (enabled with interrupt enable flag) from operating hardware or $\overline{\text{RESET}}$	

Reset

See table 7 for the state of the chip after a $\overline{\text{RESET}}$ is applied.

Table 7. State of the Device after Reset

Hardware		$\overline{\text{RESET}}$ Input During Standby Mode	$\overline{\text{RESET}}$ Input During Operation
Program counter (PC)	μPD75304	The low-order 4 bits of program memory address 0000H are loaded into PC11–PC8. The contents of address 0001H are loaded into PC7–PC0.	
	μPD75306 μPD75308 μPD75P308	The low-order 5 bits of program memory address 0000H are loaded into PC12–PC8. The contents of address 0001H are loaded into PC7–PC0.	
	μPD75312 μPD75316 μPD75P316 μPD75P316A	The low-order 6 bits of program memory address 0000H are loaded into PC13–PC8. The contents of address 0001H are loaded into PC7–PC0.	
PSW	Carry flag (CY)	Held	Undefined
	Skip flags (SK0–SK2)	0	0
	Interrupt status flag (IST0)	0	0
	Bank enable flag (MBE)	Bit 7 of program memory address 0000H is loaded into MBE	
Stack pointer (SP)		Undefined	Undefined
Data memory (RAM)		Held (Note 1)	Undefined
General purpose registers (X, A, H, L, D, E, B, C)		Held	Undefined
Bank selection register (MBS)		0	0
Basic interval timer	Counter (BT)	Undefined	Undefined
	Mode register (BTM)	0	0
Timer/event counter	Counter (T0)	0	0
	Modulo register (TMOD0)	FFH	FFH
	Mode register (TM0)	0	0
	TOE0, TOUT F/F	0, 0	0, 0
Watch timer	Mode register (WM)	0	0
Serial interface	Shift register (SIO)	Held	Undefined
	Operation mode register (CSIM)	0	0
	SBI control register (SBIC)	0	0
	Slave address register (SVA)	Held	Undefined

Table 7. State of the Device after Reset (cont)

Hardware		$\overline{\text{RESET}}$ Input During Standby Mode	$\overline{\text{RESET}}$ Input During Operation
Clock generator and clock output circuit	Processor clock control register (PCC)	0	0
	System clock control register (SCC)	0	0
	Clock output mode register (CLOM)	0	0
LCD controller	Display mode register (LCDM)	0	0
	Display control register (LCDC)	0	0
Interrupt function	Interrupt request flags (IRQxxx)	Reset to 0	Reset to 0
	Interrupt enable flags (IExxx)	0	0
	Interrupt master enable flag (IME)	0	0
	INT0, INT1, and INT2 and mode registers (IM0, IM1, and IM2)	0, 0, 0	0, 0, 0
Digital ports	Output buffers	Off	Off
	Output latches	Cleared	Cleared
	Input/output mode registers (PGMA, B)	0	0
	Pullup resistor specification register (POGA)	0	0
Bit sequential buffer		Held	Undefined
Pin conditions	$P0_0$–$P0_3$, $P1_0$–$P1_3$, $P2_0$–$P2_3$, $P3_0$–$P3_3$, $P6_0$–$P6_3$, $P7_0$–$P7_3$	Input	Input
	$P4_0$–$P4_3$, $P5_0$–$P5_3$,	With incorporated pullup resistor, high level; with open drain, high impedance	
	S0–S23 COM0–COM3	Undefined	Undefined
	BIAS	With incorporated resistor ladder, low level; with no incorporated resistor ladder, high impedance	

Notes:

(1) The data of data memory address 0F8H-0FDH is undefined by RESET.

EPROM Write and Verify

The μPD75P308 contains 8064 bytes of EPROM, while the μPD75P316/16A have 16256 bytes. Table 8 shows the pin functions during the write and verify cycles. Note that it is not necessary to enter an address, since the address is updated by pulsing the clock pins. When 6 V and 12.5 V are applied to the V_{DD} and V_{PP} pins, respectively, the EPROM is placed in the write/verify mode. The operation is selected by the MD0–MD3 pins, as shown in table 9.

Table 8. EPROM Write and Verify Pin Functions

Pin Name	Function
X1, X2	After a write/verify write, the X1, and X2 clock pins are pulsed. (Note that these pins are also pulsed during a read).
MD0–MD3	These are the operation mode selection pins.
$P4_0$–$P4_3$ (four low-order bits) $P5_0$–$P5_3$ (four high-order bits)	8-bit data input/output pins for write verify
V_{DD}	Supply voltage. Normally 5 volts; 6 volts is applied during write/verify
V_{PP}	Normally 5 volts; 12.5 volts is applied during write/verify

Notes:

(1) A cover should be placed over the UV erase window. The μPD75P308GF/P316GF/P316AGF do not have windows, thus the EPROM contents cannot be erased.

Table 9. Write/Verify Operation

V_{PP} = +12.5 V; V_{DD} = +6.0 V

Operation Mode Specification				
MD0	**MD1**	**MD2**	**MD3**	**Operation Mode**
1	0	1	0	Clear program memory address
0	1	1	1	Write mode
0	0	1	1	Verify mode
1	x	1	1	Program inhibit

Notes:

(1) x = Don't care.

EPROM Write/Verify Procedure

EPROMs can be written at high speed using the following procedure: (see figure 12)

(1) Pull unused pins to V_{SS} through resistors. Set the X1 pin low.

(2) Supply 5 volts to the V_{DD} and V_{PP} pins.

(3) Wait for 10 μs.

(4) Select the *clear program memory address* mode.

(5) Supply 6 volts to the V_{DD} and 12.5 volts to the V_{PP} pins.

(6) Select the *program inhibit* mode.

(7) Write data in the 1 ms *write* mode.

(8) Select the *program inhibit* mode.

(9) Select the *verify* mode. If the data is correct, proceed to step 10. If not, repeat steps 7, 8, and 9.

(10) Perform one additional write.

(11) Select the *program inhibit* mode.

(12) Apply four pulses to the X1 pin to increment the program memory address by one.

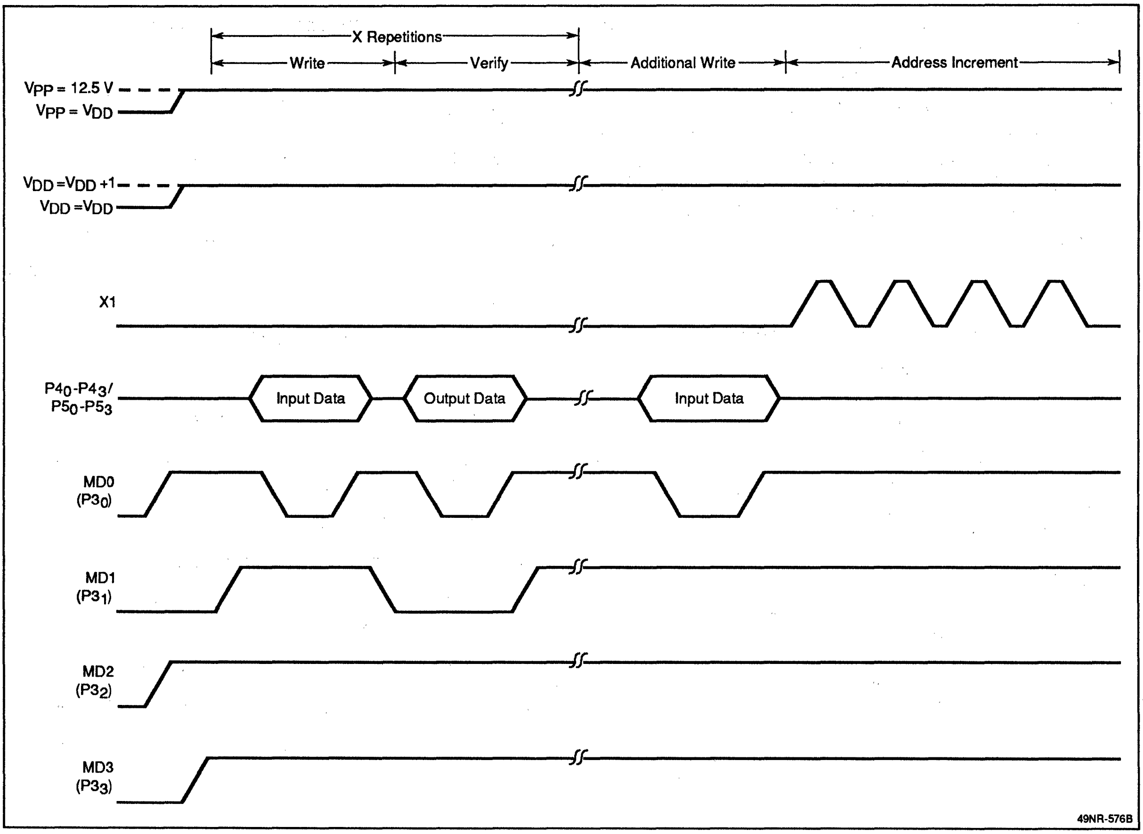
(13) Repeat steps 7-12 until the end address is reached.

(14) Select the *clear program memory address* mode.

(15) Return the V_{DD} and V_{PP} pins back to + 5 volts.

(16) Turn off the power.

Figure 12. EPROM Write/Verify Cycle Timing



EPROM Read Procedure

The EPROM contents can be read by using the following procedure: (see figure 13)

(1) Pull unused pins to V_{SS} through resistors. Set the X1 pin low.

(2) Supply 5 volts to the V_{DD} and V_{PP} pins.

(3) Wait for 10 μs.

(4) Select the *clear program memory address* mode.

(5) Supply 6 volts to the V_{DD} and 12.5 volts to the V_{PP} pins.

(6) Select the *program inhibit* mode.

(7) Select the *verify* mode. Apply four pulses to the X1 pin. Every four clock pulses will output the data stored in one address.

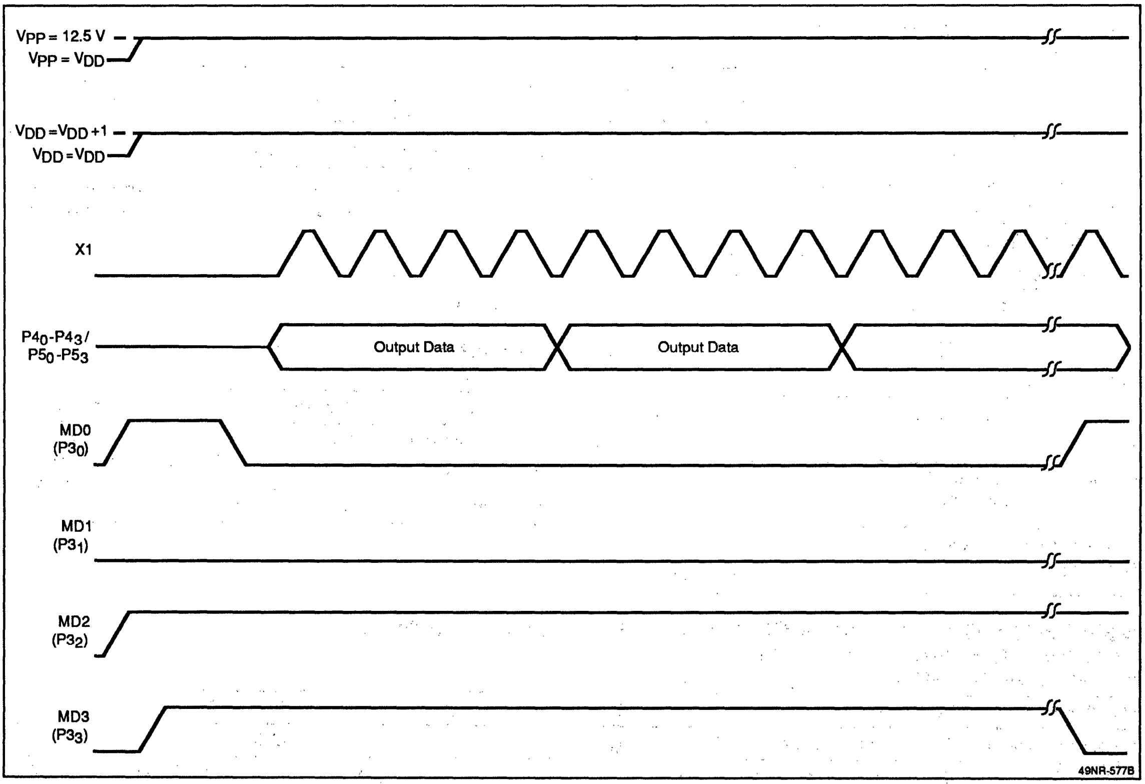
(8) Select the *program inhibit* mode.

(9) Select the *clear program memory address* mode.

(10) Return the V_{DD} and V_{PP} pins back to + 5 volts.

(11) Turn off the power.

Figure 13. EPROM Read Cycle Timing



Program Memory Erase (μPD75P308K/P316AK only)

The μPD75P308K/P316AK allows the programmed data contents to be erased by light rays whose wavelength is shorter than about 400 nm. The programmed data contents may also be erased if the uncovered window is exposed to direct sunlight or a fluorescent light for several hours. Thus, to protect the data contents, cover the window with an opaque film. NEC attaches quality-tested shading film to the UV EPROM products for shipping.

For normal EPROM erase, place the device under an ultraviolet light source (254 nm). The minimum amount of radiation exposure required to erase the μPD75P308K completely is 15 Ws/cm^2 (ultraviolet ray strength times erase time). This corresponds to about 15 to 20 minutes when using a UV lamp of 12000 μW/cm^2. However, the erase time may be prolonged if the UV lamp is old or if the device window is dirty. The distance between the light source and the window should be 2.5 cm or less.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings (All devices)

T_A = 25°C

Supply voltage, V_{DD}	−0.3 to +7.0 V
Supply voltage, V_{PP} (75P308/P316 only)	−0.3 to +13.5 V
Input voltage, V_{I1} (other than ports 4, 5)	−0.3 to V_{DD} + 0.3 V
Input voltage, V_{I2} (ports 4, 5; internal pullup resistor; 7530x/31x only)	−0.3 to V_{DD} + 0.3 V
Input voltage, V_{I3} (ports 4, 5; open drain)	−0.3 to +11 V
Output voltage, V_O	−0.3 to V_{DD} + 0.3 V
High-level output current, I_{OH} (Single pin)	−15 mA
High-level output current, I_{OH} (Total of all pins)	−30 mA
Low-level output current, I_{OL} (Single pin)	30 mA peak
	15 mA rms (Note 1)
Low-level output current, I_{OL} (Total of ports 0, 2, 3, 5)	100 mA peak
	60 mA rms (Note 1)
Low-level output current, I_{OL} (Total of ports 4, 6, 7)	100 mA peak
	60 mA rms (Note 1)
Storage temperature, t_{STG}	−65 to + 150°C
Operating temperature, t_{OPT} (7530x/31x)	−40 to +85°C
Operating temperature, t_{OPT} (75P308/P316)	−10 to +70°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Notes:

(1) rms value = peak x (duty cycle)$^{1/2}$.

Capacitance (All devices)

V_{DD} = 0 V; T_A = 25°C

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_{IN}	15	pF	f = 1 MHz; all unmeasured pins returned to ground
Output capacitance	C_{OUT}	15	pF	
I/O capacitance	$C_{I/O}$	15	pF	

Main System Clock Oscillator Characteristics (All devices, see figure 16)

μPD7530x/31x: T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V
μPD75P308/P316: T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 14)	Oscillation frequency (Note 1)	f_{XX}	1.0		5.0	MHz	
	Oscillation stabilization time (Note 2)				4 (Note 3)	ms	After V_{DD} reaches oscillator operating voltage
Crystal resonator (Figure 14)	Oscillation frequency (Note 1)	f_{XX}	1.0	4.19	5.0	MHz	
	Oscillation stabilization time (Note 2)				10 (Note 3)	ms	V_{DD} = 4.5 to V_{DD} max
					30 (Note 3)	ms	V_{DD} = 2.7 to 6.0 V (μPD7530x/31x only)
External clock (Figure 14)	X1 input frequency (Note 1)	f_{XX}	1.0		5.0	MHz	
	X1 input low- and high-level width	t_{XH}, t_{XL}	100		500	ns	

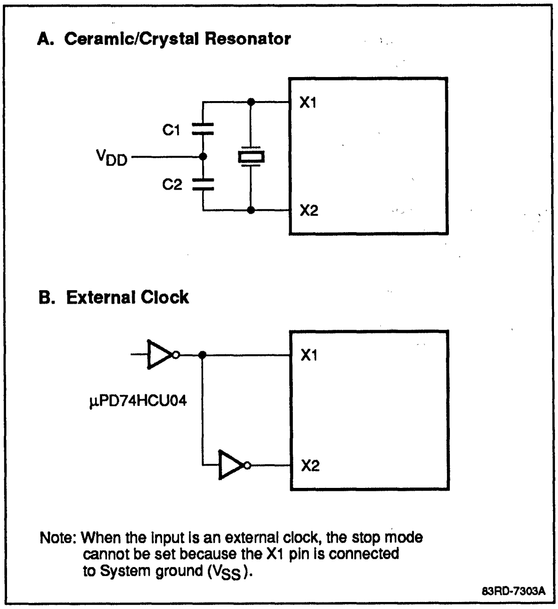
Notes:

(1) The oscillation frequency and X1 input frequency are included only to show the characteristics of the oscillators. Refer to the AC Characteristics table for actual instruction execution times.

(2) The oscillation stabilization time is the time required for the oscillator to stabilize after voltage is applied or the STOP mode is released.

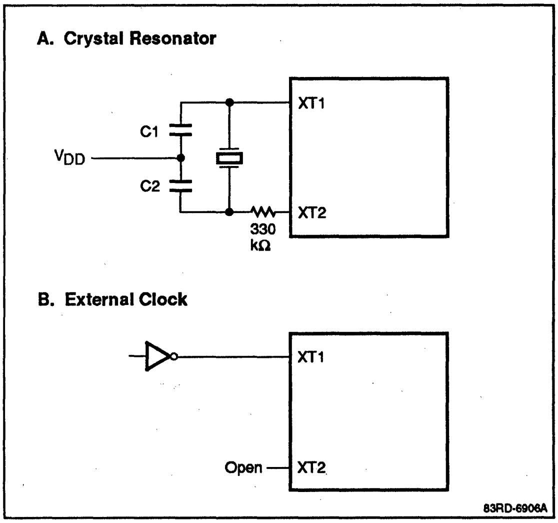
(3) Values shown are for the recommended resonators. Values for resonators not shown in this data sheet should be obtained from the manufacturer's spec sheets.

Figure 14. Main System Clock Configurations



Note: When the input is an external clock, the stop mode cannot be set because the X1 pin is connected to System ground (V_{SS}).

Figure 15. Subsystem Clock Configurations



Subsystem Clock Oscillator Characteristics (All devices, see figure 16)

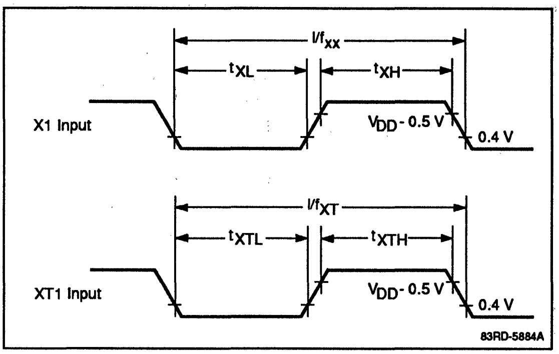
µPD7530x/31x: T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V
µPD75P308/P316: T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Oscillator	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 14A)	Oscillation frequency	f_{XT}	32	32.768	35	kHz	
	Oscillation stabilization time (Note 1)			1.0	2	s	V_{DD} = 4.5 to V_{DD} max
					10	s	V_{DD} = 2.7 to 6.0 V (PD7530x/31x only)
External clock (Figure 14B)	XT1 input frequency	f_{XT}	32		100	kHz	
	XT1 input low- and high-level width	t_{XTH}, t_{XTL}	10		15	µs	

Notes:

(1) Values shown are for the recommended. Values for resonators not shown in this data sheet should be obtained from the manufacturer's spec sheets.

Figure 16. Clock AC Timing Points X1 and XT1



Recommended Main System Ceramic Resonators (μPD7530x/31x only)

Manufacturer	Part Number	C1 (pF)	C2 (pF)	Remarks
Murata	CSA 2.00MG093	15	15	V_{DD} = 2.5 to 3.5 V
	CSB 1000D20	220	220	V_{DD} = 2.7 to 6.0 V
	CSA 2.00MG093 CSA 4.19MGU CSA 4.91MGU	30 30 30	30 30 30	
	CST 2.00MG093 CST 4.19MGU CST 4.91MGU	None None None	None None None	V_{DD} = 2.7 to 6.0 V (Note 1)
Kyocera	KBR-1000H	100	100	V_{DD} = 3.0 to 6.0 V
	KBR-2.0MS	68	68	
	KBR-4.0MS KBR-4.19MS KBR-4.91MS	33 33 33	33 33 33	

Notes:

(1) C1 and C2 are contained in the oscillator.

Recommended Main System Crystal Resonators (μPD7530x/31x only)

Manufacturer	Frequency (MHz)	Retainer	C1 (pF)	C2 (pF)	Remarks
Kinseki	2.00	HC-18/U	22	22	V_{DD} = 2.7 to 6.0 V
	4.19	HC-49/U	22	22	
	4.91	HC-43/U	22	22	

Recommended Subsystem Crystal Resonators (μPD7530x/31x only)

Manufacturer	Type	C1 (pF)	C2 (pF)	R (kΩ)	Remarks
Kinseki	P-3	22	22	330	V_{DD} = 2.7 to 6.0 V

DC Characteristics (μPD7530x/31x)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	$0.7V_{DD}$		V_{DD}	V	Ports 2, 3
	V_{IH2}	$0.8V_{DD}$		V_{DD}	V	Ports 0, 1, 6, 7; and $\overline{\text{RESET}}$
	V_{IH3}	$0.7V_{DD}$		V_{DD}	V	Ports 4 and 5; built-in pullup resistor
		$0.7V_{DD}$		10	V	Ports 4 and 5; open drain
	V_{IH4}	V_{DD}−0.5		V_{DD}	V	X1, X2, XT1
Low-level input voltage	V_{IL1}	0		$0.3V_{DD}$	V	Ports 2, 3, 4, 5
	V_{IL2}	0		$0.2V_{DD}$	V	Ports 0, 1, 6, 7; $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2, XT1
High-level output voltage	V_{OH1}	V_{DD}−1.0			V	Ports 0, 2, 3, 6, 7, BIAS; V_{DD} = 4.5 to 6.0 V; I_{OH} = −1 mA
		V_{DD}−0.5			V	Ports 0, 2, 3, 6, 7, BIAS; V_{DD} = 2.7 to 6.0 V; I_{OH} = −100 μA
	V_{OH2}	V_{DD}−2.0			V	BP_{0-7} (with two I_{OL} outputs) V_{DD} = 4.5 to 6.0 V; I_{OH} = −100 μA
		V_{DD}−1.0			V	BP_{0-7} (with two I_{OL} outputs) V_{DD} = 2.7 to 6.0 V; I_{OH} = −30 μA
Low-level output voltage	V_{OL1}		0.4	2.0	V	Ports 3, 4, 5; V_{DD} = 4.5 to 6.0 V; I_{OL}= 15 mA
				0.4	V	Ports 0, 2-7; V_{DD} = 4.5 to 6.0 V; I_{OL}= 1.6 mA
				0.5	V	Ports 0, 2-7; V_{DD} = 2.7 to 6.0 V; I_{OL}= 400 μA
				$0.2V_{DD}$	V	SB0, 1; V_{DD} = 2.7 to 6.0 V; pullup resistance ≥ 1kΩ
	V_{OL2}			1.0	V	BP_{0-7} (with two I_{OL} outputs) V_{DD} = 4.5 to 6.0 V; I_{OL} = 100 μA
				1.0	V	BP_{0-7} (with two I_{OL} outputs) V_{DD} = 2.7 to 6.0 V; I_{OL} = 50 μA
High-level input leakage current	I_{LIH1}			3	μA	All except X1, X2, XT1 and ports 4, 5; V_{IN} = V_{DD}
	I_{LIH2}			20	μA	X1, X2, and XT1; V_{IN} = V_{DD}
	I_{LIH3}			20	μA	Ports 4 and 5 (with open drain); V_{IN} = 10 V
Low-level input leakage current	I_{LIL1}			−3	μA	All except X1, X2, and XT1; V_{IN} = 0 V
	I_{LIL2}			−20	μA	X1, X2, and XT1; V_{IN} = 0 V
High-level output leakage current	I_{LOH1}			3	μA	Other than Ports 4 and 5; V_{OUT} = V_{DD}
	I_{LOH2}			20	μA	Ports 4 and 5 (open drain); V_{OUT} = 10 V
Low-level output leakage current	I_{LOL}			−3	μA	V_{OUT} = 0 V

DC Characteristics (μPD7530x/31x) (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Built-in pullup resistor	R_{L1}	15	40	80	kΩ	Ports 0-3, 6, 7 (except $P0_0$); V_{IN} = 0 V; V_{DD} = 5.0 V ± 10%
		30		200	kΩ	Ports 0-3, 6, 7 (except $P0_0$); V_{IN} = 0 V; V_{DD} = 3.0 V ± 10%
	R_{L2}	15	40	70	kΩ	Ports 4, 5; V_{OUT} = V_{DD}–2 V; V_{DD} = 5.0 V ± 10%
		10		60	kΩ	Ports 4, 5; V_{OUT} = V_{DD}–2 V; V_{DD} = 3.0 V ± 10%
LCD drive voltage	V_{LCD}	2.5		V_{DD}	V	
LCD split resistor	R_{LCD}	60	100	150	kΩ	
LCD output voltage deviation; common (Note 1)	V_{ODC}	0		±0.2	V	I_O = ±5 μA; V_{LCD} = V_{LCD0} = 2.75 V to V_{DD}; V_{LCD1} = 2/3 V_{LCD} V_{LCD2} = 1/3 V_{LCD}
LCD output voltage deviation; segment (Note 1)	V_{ODS}	0		±0.2	V	I_O = ±1 μA; V_{LCD} = V_{LCD0} = 2.75 V to V_{DD}; V_{LCD1} = 2/3 V_{LCD} V_{LCD2} = 1/3 V_{LCD}
Supply current (Note 3)	I_{DD1} (Note 2)		2.5	8.0	mA	V_{DD} = 5 V ± 10% (Note 4)
			0.35	1.2	mA	V_{DD} = 3 V ± 10% (Note 5)
	I_{DD2} (Note 2)		500	1500	μA	HALT mode; V_{DD} = 5 V ± 10%
			150	450	μA	HALT mode; V_{DD} = 3 V ± 10%
	I_{DD3}		30	90	μA	V_{DD} = 3 V ± 10% (Note 6)
	I_{DD4}		5	15	μA	HALT mode; V_{DD} = 3 V ± 10% (Note 6)
	I_{DD5}		0.5	20	μA	STOP mode; XT1 = 0 V; V_{DD} = 5 V ± 10%
			0.1	10	μA	STOP mode; XT1 = 0 V; V_{DD} = 3 V ± 10%
			0.1	5	μA	STOP mode; XT1 = 0 V; V_{DD} = 3 V ± 10%; T_A = 25°C

Notes:

(1) Voltage deviation is the difference between the ideal value of segment or common output (V_{LCDn}; n = 0, 1, 2) and the output voltage.

(2) 4.19 MHz crystal oscillator; C1 = C2 = 22 pF.

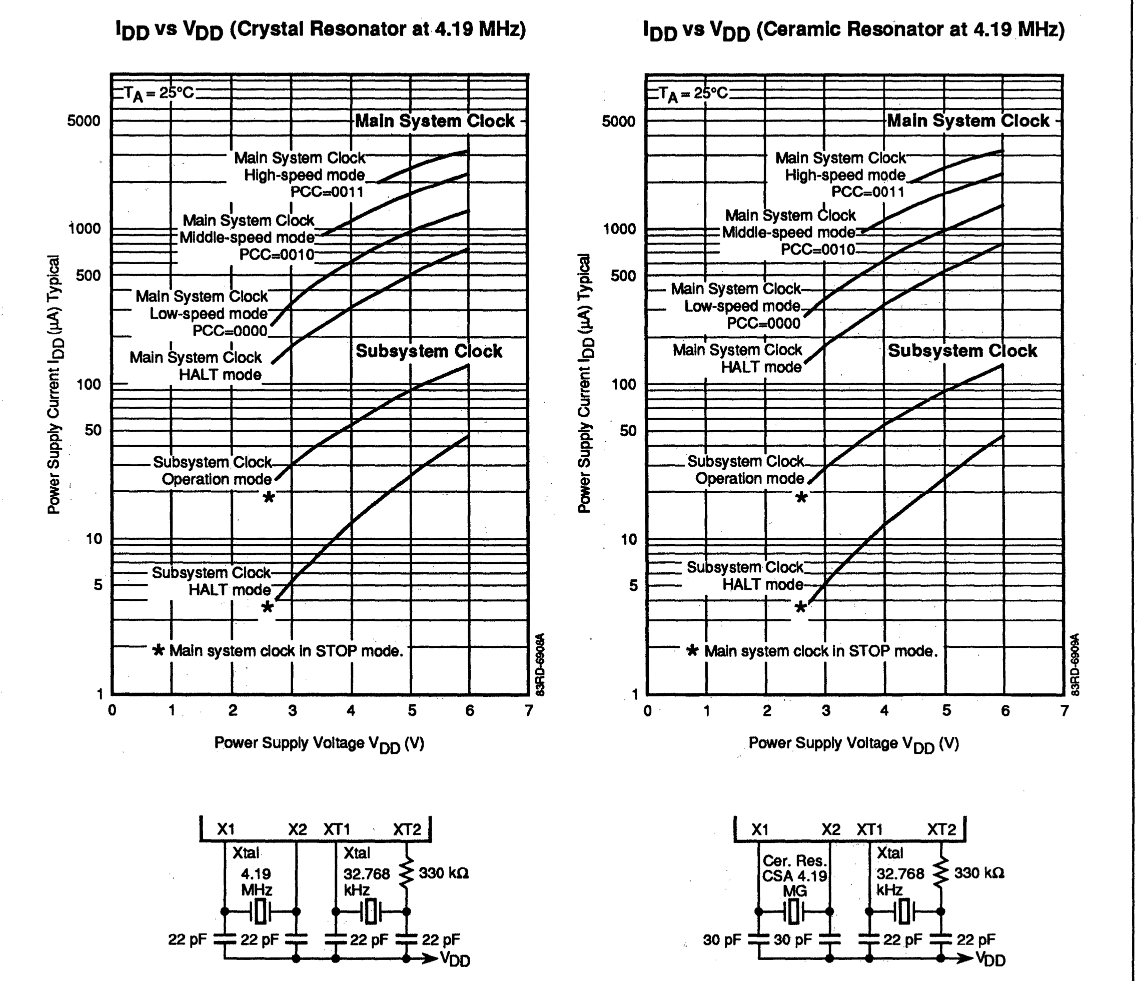
(3) Does not include pullup resistor current and current through LCD resistor ladder.

(4) When operated in the high-speed mode with the processor clock control register (PCC) set to 0011.

(5) When operated in the low-speed mode with the PCC set to 0000.

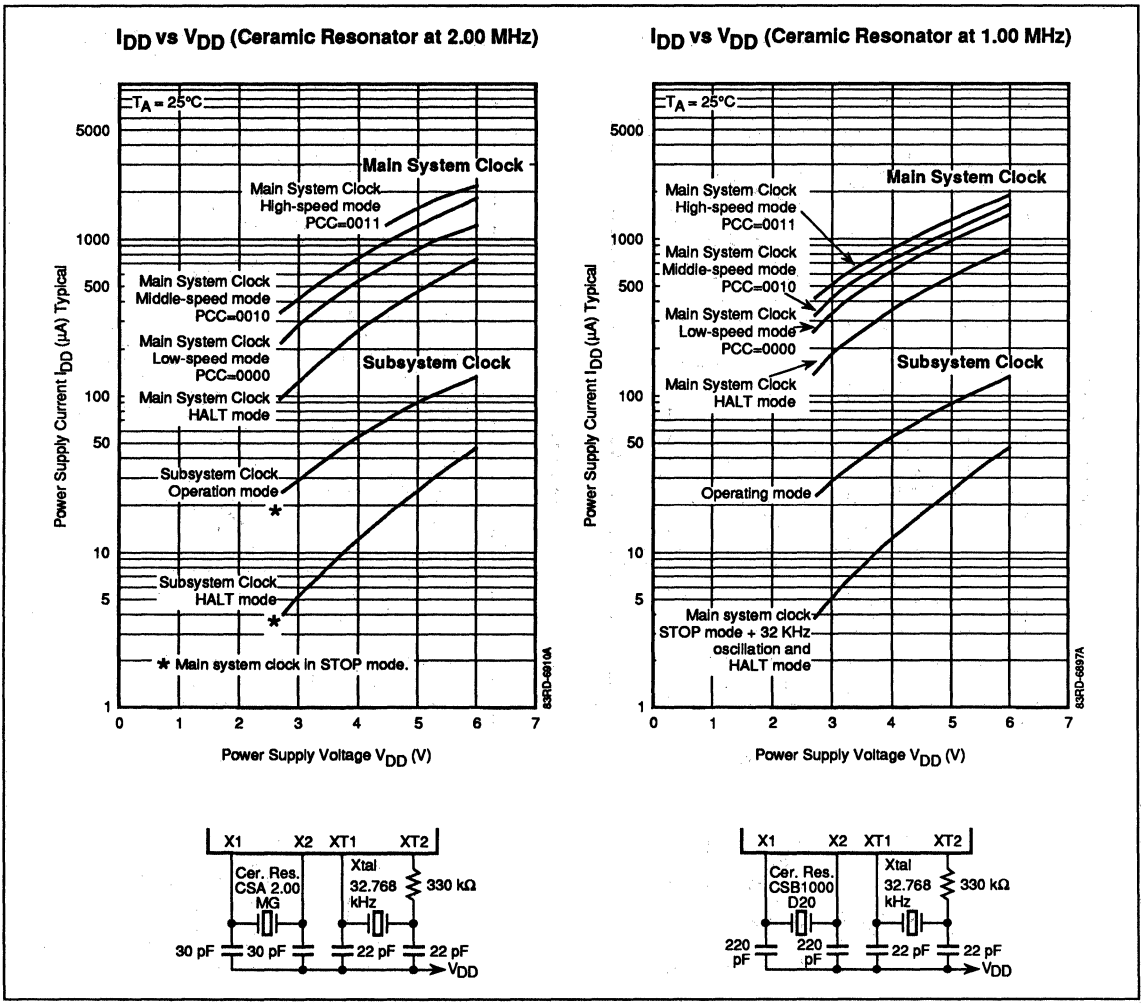
(6) Main system clock stopped and subsystem clock running (SCC = 1001).

Figure 17. DC Characteristics



Note : Values of I_{DD} are about 10% larger using a ceramic resonator as compared to a crystal resonator.

Figure 17. DC Characteristics (cont)



I_{DD} vs V_{DD} (Ceramic Resonator at 2.00 MHz)

I_{DD} vs V_{DD} (Ceramic Resonator at 1.00 MHz)

Figure 17. DC Characteristics (cont)

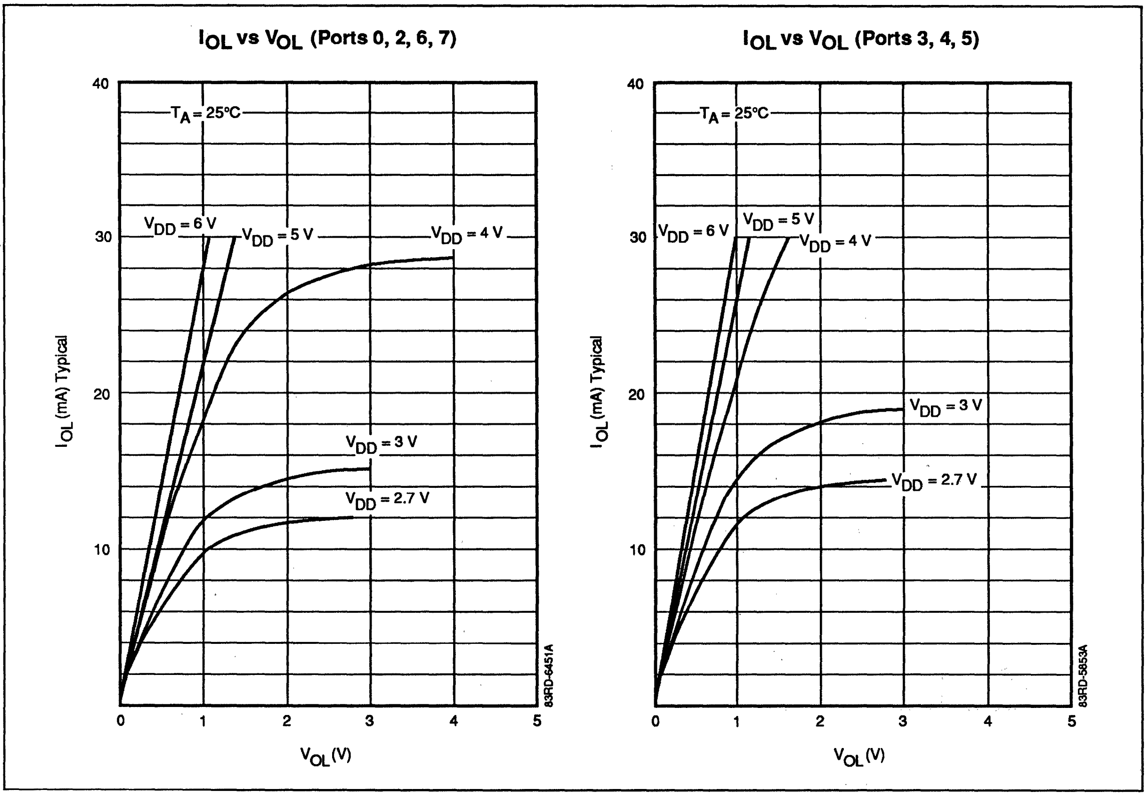


Figure 17. DC Characteristics (cont)

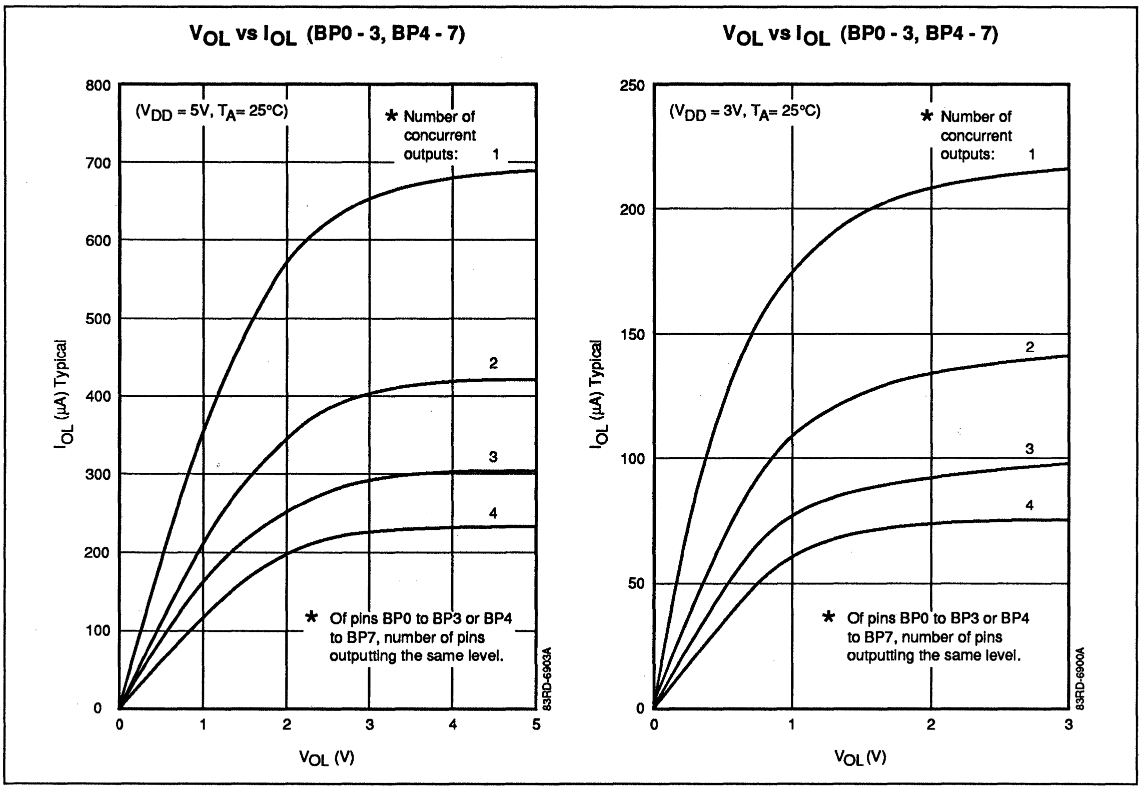
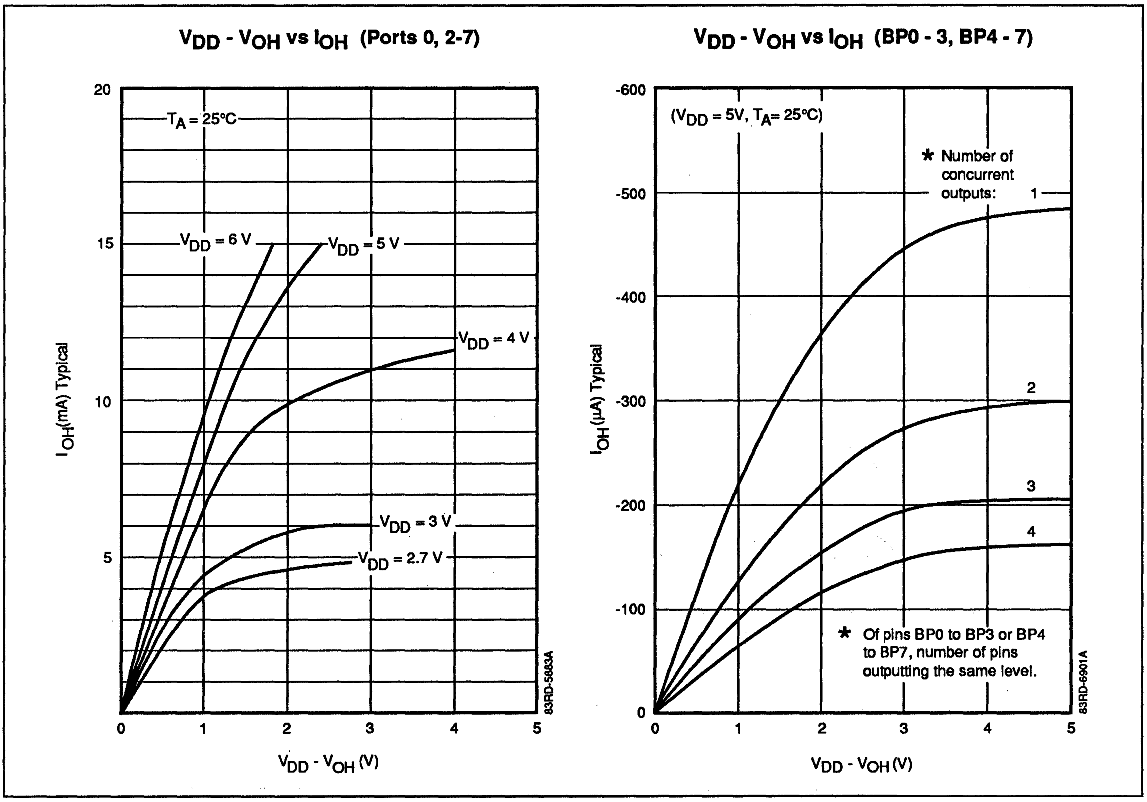


Figure 17. DC Characteristics (cont)



4

Figure 17. DC Characteristics (cont)

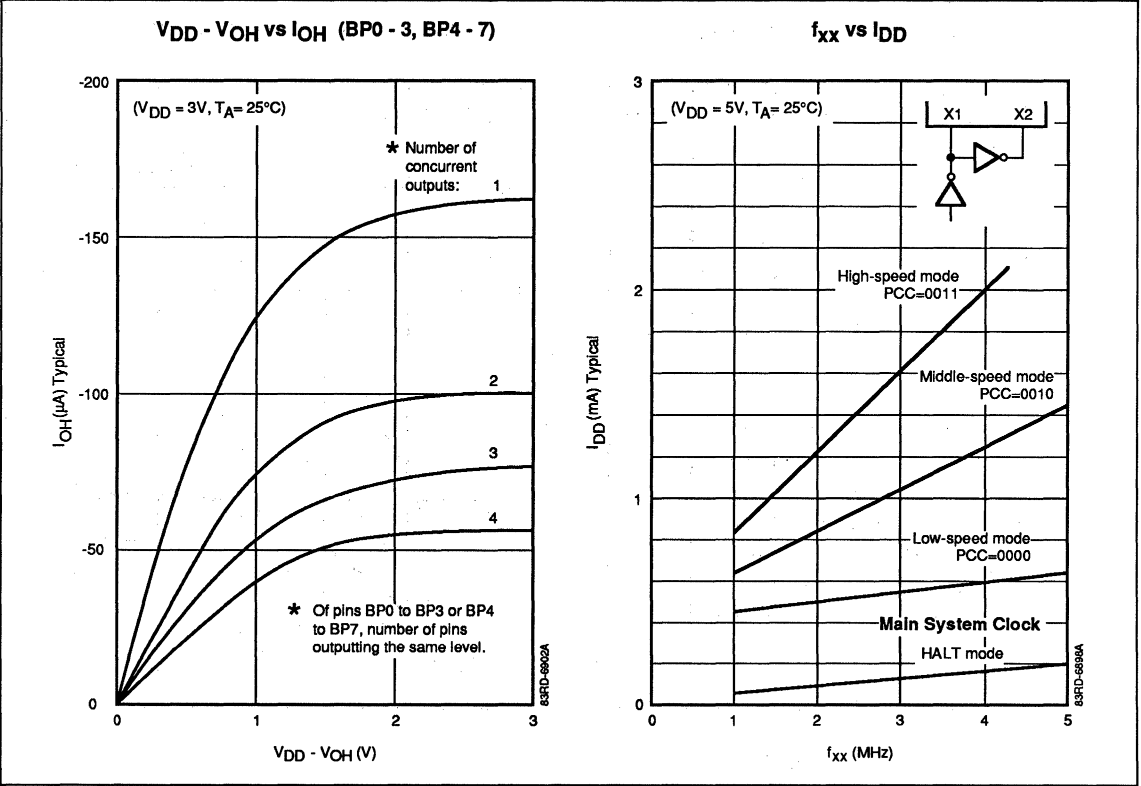
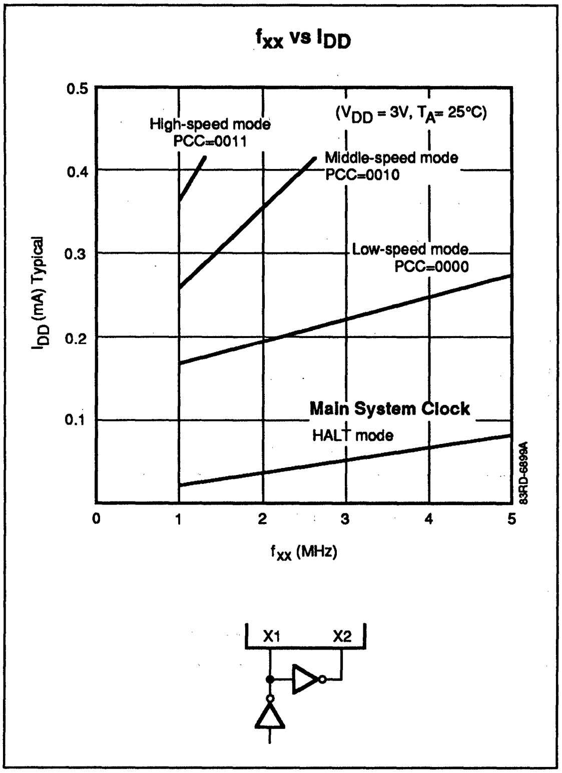


Figure 17. DC Characteristics (cont)



AC Characteristics (μPD7530x/31x only)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Note 1)	t_{CY} (Figure 19)	0.95		64	μs	Main system clock; V_{DD} = 4.5 to 6.0 V
		3.8		64	μs	Main system clock; V_{DD} = 2.7 to 6.0 V
		114	122	125	μs	Subsystem clock
TI0 input frequency	f_{TI} (Figure 20)	0		1	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	V_{DD} = 2.7 to 6.0 V
TI0 input low- and high-level width	t_{TIH}, t_{TIL} (Figure 20)	0.48			μs	V_{DD} = 4.5 to 6.0 V
		1.8			μs	V_{DD} = 2.7 to 6.0 V
Interrupt inputs low- and high-level width	t_{INTH}/t_{INTL} (Figure 21)	(Note 2)			μs	INT0
		10			μs	INT1, 2, 4
		10			μs	KR0-KR7
$\overline{\text{RESET}}$ low level width	t_{RSL} (Figure 22)	10			μs	

Notes:

(1) Cycle time (minimum instruction execution time) is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See figure 19.

(2) $2t_{CY}$ or $128/f_X$, depending on the setting of the interrupt mode register (IM0).

Figure 18. AC Timing Measurement Points (except X1 and XT1)

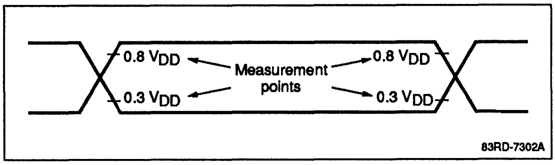


Figure 19. Guaranteed Operating Range (µPD7530x/31x only)

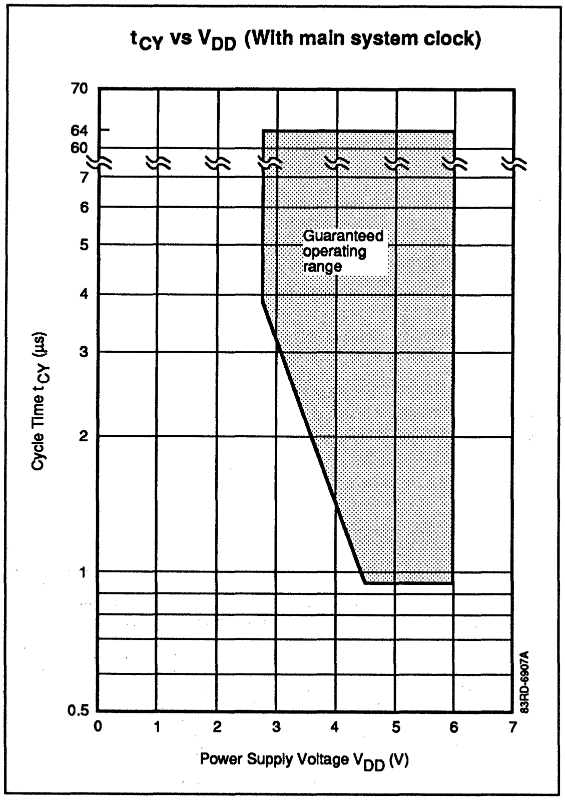


Figure 20. TI0 Timing

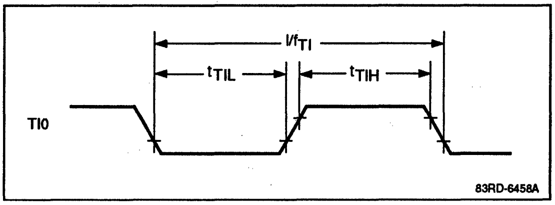


Figure 21. Interrupt Input Timing

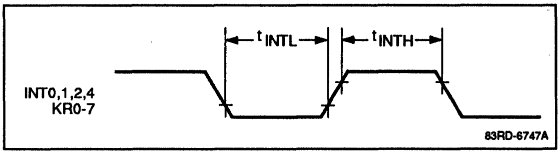
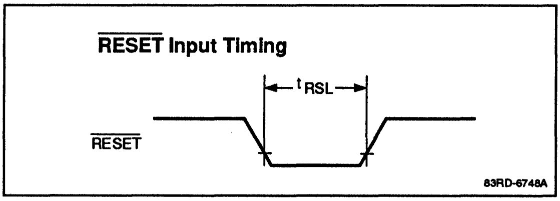


Figure 22. $\overline{\text{RESET}}$ Timing


$\overline{\text{RESET}}$ Input Timing
t_{RSL}
$\overline{\text{RESET}}$
83RD-6748A

Serial Transfer Operation (μPD7530x/31x only) (see figures 18, 23)

2-line/3-line Serial I/O mode ($\overline{SCK}$ = internal clock output)
T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KH1}/ t_{KL1}	0.5 t_{KCY1} −50			ns	V_{DD} = 4.5 to 6.0 V
		0.5 t_{KCY1} −150			ns	V_{DD} = 2.7 to 6.0 V
SI vs $\overline{SCK}$ ↑ setup time	t_{SIK1}	150			ns	
SI vs $\overline{SCK}$ ↑ hold time	t_{KSI1}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO1}			250	ns	V_{DD} = 4.5 to 6.0 V
				1000	ns	V_{DD} = 2.7 to 6.0 V

Serial Transfer Operation (μPD7530x/31x only) (see figures 18, 23)

2-line/3-line Serial I/O mode ($\overline{SCK}$ = external clock output)
T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY2}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KH2}/ t_{KL2}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	V_{DD} = 2.7 to 6.0 V
SI vs $\overline{SCK}$ ↑ setup time	t_{SIK2}	100			ns	
SI vs $\overline{SCK}$ ↑ hold time	t_{KSI2}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO2}			300	ns	V_{DD} = 4.5 to 6.0 V
				1000	ns	V_{DD} = 2.7 to 6.0 V

SBI Mode (μPD7530x/31x only) (see figures 18, 23)

$\overline{SCK}$ = internal clock output (master)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY3}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KH3}/ t_{KL3}	0.5 t_{KCY3}−50			ns	V_{DD} = 4.5 to 6.0 V
		0.5 t_{KCY3}−150			ns	V_{DD} = 2.7 to 6.0 V
SB0, SB1 vs $\overline{SCK}$ ↑ setup time	t_{SIK3}	150			ns	
SB0, SB1 vs $\overline{SCK}$ ↑ hold time	t_{KSI3}	0.5 t_{KCY3}			ns	
$\overline{SCK}$ ↓ to SB0, SB1 output delay time	t_{KSO3}	0		250	ns	V_{DD} = 4.5 to 6.0 V
		0		1000	ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY3}			ns	
SB0, SB1 ↓ to $\overline{SCK}$ ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	

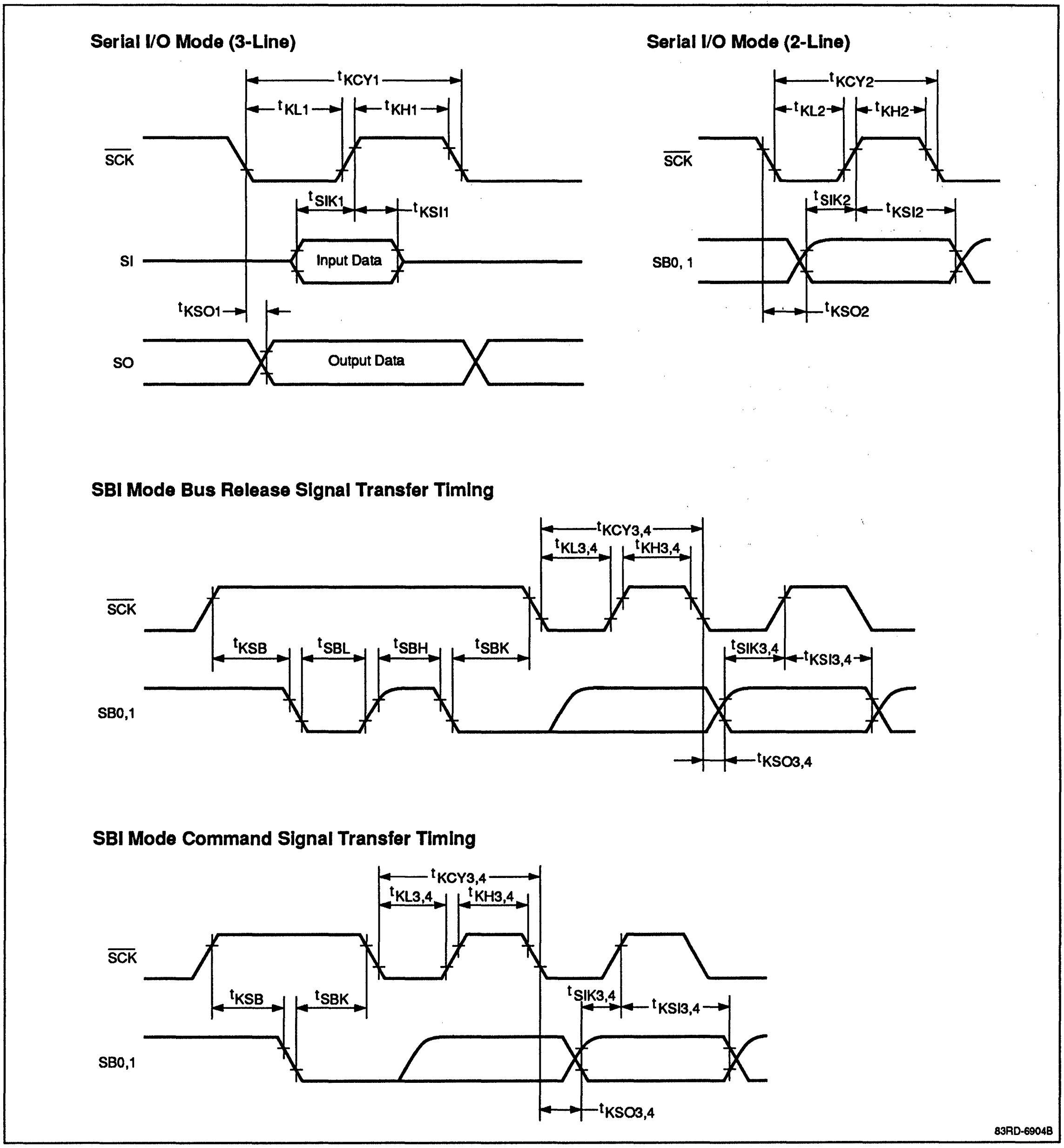
SBI Mode (μPD7530x/31x only) (see figures 18, 23)

$\overline{SCK}$ = external clock output (slave)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY4}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ low- and high-level width	t_{KH4}/ t_{KL4}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	V_{DD} = 2.7 to 6.0 V
SB0, SB1 vs $\overline{SCK}$ ↑ setup time	t_{SIK4}	100			ns	
SB0, SB1 vs $\overline{SCK}$ ↑ hold time	t_{KSI4}	0.5 t_{KCY4}			ns	
$\overline{SCK}$ ↓ to SB0, SB1 output delay time	t_{KSO4}	0		300	ns	V_{DD} = 4.5 to 6.0 V
		0		1000	ns	V_{DD} = 2.7 to 6.0 V
$\overline{SCK}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY4}			ns	
SB0, SB1 ↓ to $\overline{SCK}$ ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	

Figure 23. Serial Transfer Timing



Serial I/O Mode (3-Line)

Serial I/O Mode (2-Line)

SBI Mode Bus Release Signal Transfer Timing

SBI Mode Command Signal Transfer Timing

Data Memory STOP Mode Low Voltage Data Retention Characteristics (μPD7530x/31x) (see figure 24)

T_A = −40 to +85°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		6.0	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization time (Note 2)	t_{WAIT}		(Notes 3, 4)		ms	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

(1) Excludes current in the pullup resistors.

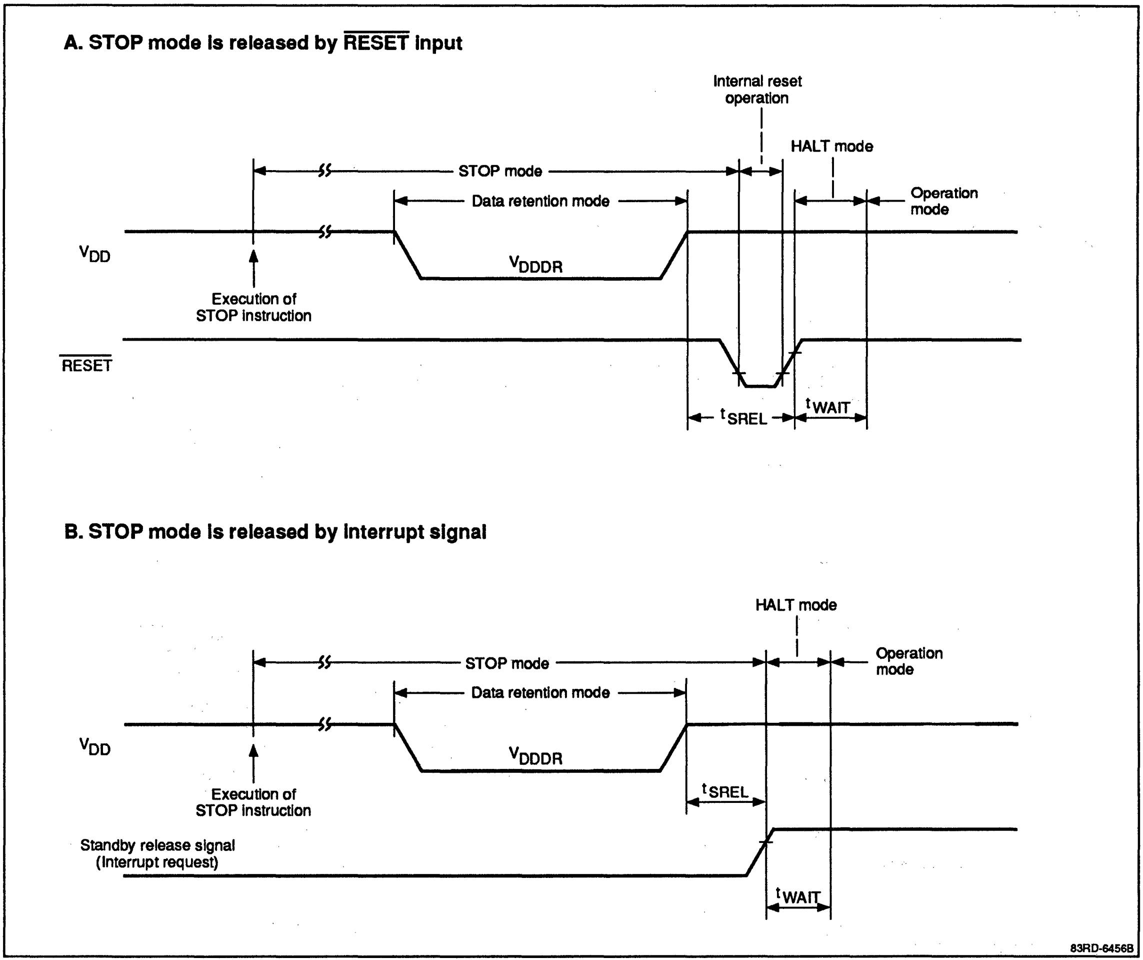
(2) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillation is started. The interval timer can be used to delay the CPU from executing instructions using the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{XX} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{XX}$ (Approx 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (Approx 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (Approx 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (Approx 1.95 ms)

(3) Consult the manufacturer's resonator or crystal specification sheet for this value.

(4) The interval timer will cause a delay of 2^{17}/fxx after a reset.

Figure 24. Data Retention Timing



A. STOP mode is released by $\overline{\text{RESET}}$ input

B. STOP mode is released by interrupt signal

Recommended Ceramic Resonators (μPD75P308/P316)

V_{DD} = 4.75 to 5.25 V; T_A = −10 to +70°C

Manufacturer	Part Number	C1 (pF)	C2 (pF)	Remarks
Murata	CSA 2.00MG	30	30	
	CSA 4.19MG	30	30	
	CSA 4.91MGU	30	30	
	CST 4.19MG	(Note 1)	(Note 1)	

Notes:

(1) 30 pF capacitors are internally provided.

DC Characteristics (μPD75P308/P316)

T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Ports 2, 3
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	Ports 0, 1, 6, 7; $\overline{\text{RESET}}$
	V_{IH3}	0.7		10	V	Ports 4, 5; open drain
	V_{IH4}	V_{DD}−0.5		V_{DD}	V	X1, X2, XT1
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Ports 2, 3, 4, 5
	V_{IL2}	0		0.2 V_{DD}	V	Ports 0, 1, 6, 7, $\overline{\text{RESET}}$
	V_{IL3}	0		0.4	V	X1, X2, XT1
High-level output voltage	V_{OH1}	V_{DD}−1.0			V	Ports 0, 2, 3, 6, 7, BIAS; I_{OH} = −1 mA
	V_{OH2}	V_{DD}−2.0			V	BP_{0-7}; I_{OH} = −100 μA (Note 1)
Low-level output voltage	VOL1		0.4	2.0	V	Ports 3, 4, 5; I_{OL}= 15 mA
				0.4	V	All output pins; I_{OL}= 1.6 mA
	V_{OL2}			0.2 V_{DD}	V	SB0, SB1; open drain pullup resistor ≥ 1kΩ
	V_{OL3}			1.0	V	BP_{0-7}; I_{OL} = 100 μA (Note 1)
High-level input leakage current	I_{LIH1}			3	μA	All except X1, X2, and XT1; V_{IN} = V_{DD}
	ILIH2			20	μA	X1, X2, and XT1; V_{IN} = V_{DD}
	ILIH3			20	μA	Ports 4 and 5; V_{IN} = 10 V
Low-level input leakage current	I_{LIL1}			−3	μA	All except X1, X2, and XT1; V_{IN} = 0 V
	I_{LIL2}			−20	μA	X1, X2, and XT1; V_{IN} = 0 V
High-level output leakage current	I_{LOH1}			3	μA	All output pins except ports 4, 5; V_{OUT} = V_{DD}
	I_{LOH2}			20	μA	Ports 4, 5; V_{OUT} = 10 V
Low-level output leakage current	I_{LOL}			−3	μA	V_{OUT} = 0 V
Internal pullup resistor	R_{LI}	15	40	80	kΩ	Ports 0-3, 6, 7 (except PO_0); V_{IN} = 0 V
LCD drive voltage	V_{LCD}	2.5		V_{DD}	V	
LCD output voltage deviation; common (Note 7)	V_{ODC}	0		±0.2	V	I_O = ±5 μA; V_{LCD} = V_{LCD0} = 2.75 V to V_{DD}; V_{LCD1} = 2/ 3V_{LCD} V_{LCD2} = 1/3V_{LCD}
LCD output voltage deviation; segment (Note 7)	V_{ODS}	0		±0.2	V	I_O = ±1 μA; V_{LCD} = V_{LCD0} = 2.75 V to V_{DD}; V_{LCD1} = 2/3V_{LCD} V_{LCD2} = 1/3V_{LCD}

DC Characteristics (μPD75P308/P316) (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Supply current (Note 2)	I_{DD1}		5.0	15.0	mA	(Notes 3, 4)
	I_{DD2}		500	1500	μA	HALT Mode (Note 3)
	I_{DD3}		350	1000	μA	(Notes 5, 6)
			35	100	μA	HALT mode (Notes 5, 6)
	I_{DD4}		0.5	20	μA	STOP mode; XT1 = 0V (Note 6)

Notes:

(1) When any two pins of BP0-BP3 and any two pins of BP4-BP7 are used simultaneously for output.

(2) Does not include pullup resistor current.

(3) 4.19 MHz crystal oscillator; C1 = C2 = 15 pF, and subsystem clock running.

(4) Value during high-speed operation and the processor control clock (PCC) is set to 0011.

(5) Value when the system clock control register (SCC) is set to 1001, generation of the main system clock pulse is stopped, and the SCC is operated by the subsystem clock.

(6) 32 MHz crystal oscillator.

(7) Voltage deviation is the difference between the ideal value of segment or common output (V_{LCDn}; n = 0, 1, 2) and the output voltage.

AC Characteristics (μPD75P308/P316) (see figure 18)

T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

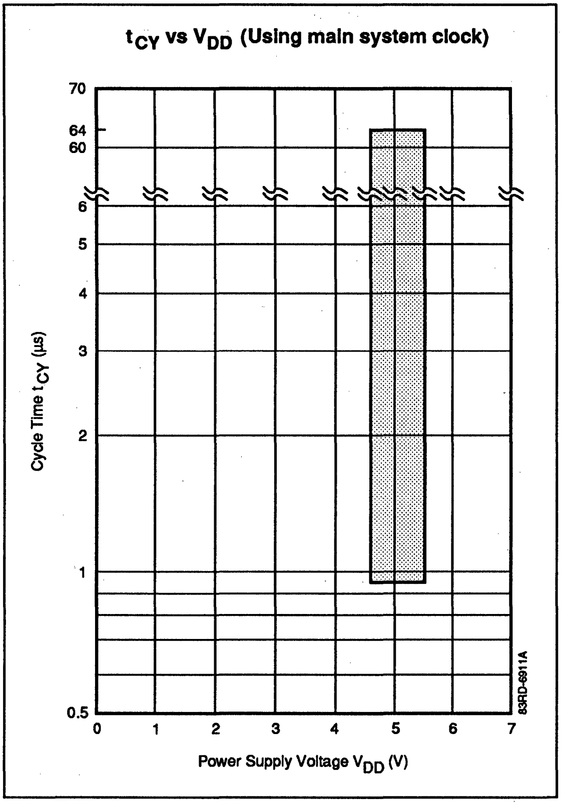
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (minimum instruction execution time — Note 1)	t_{CY} (Figure 20)	0.95		64	μs	Main system clock
		114	122	125	μs	Subsystem clock
TI0 input frequency	f_{TI} (Figure 20)	0		1	MHz	
TI0 input low- and high-level width	t_{TIH}, t_{TIL} (Figure 20)	0.48			μs	
Interrupt inputs low- and high-level width	t_{INTH}/t_{INTL} (Figure 21)	(Note 2)			μs	INT0
		10			μs	INT1, INT2, INT4 KR0–KR7
$\overline{\text{RESET}}$ low level width	t_{RSL} (Figure 22)	10			μs	

Notes:

(1) Cycle time is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See figure 25.

(2) $2t_{CY}$ or $128/f_{xx}$, depending on the setting of the interrupt mode register(IM0).

Figure 25. Guaranteed Operating Range (μPD75P308/P316)



Serial Transfer Operation (μPD75P308/P316) (see figures 18, 23)

2-line/3-line serial I/O mode (SCK = internal clock output

T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	1600			ns	
$\overline{SCK}$ low- and high-level width	t_{KL1}/t_{KH1}	0.5 t_{KCY1}−50			ns	
SI set-up time (against $\overline{SCK}$ ↑)	t_{SIK1}	150			ns	
SI hold time (against $\overline{SCK}$ ↑)	t_{KSI1}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO1}			250	ns	

Serial Transfer Operation (μPD75P308/P316) (see figures 18, 23)

2-line/3-line serial I/O mode ($\overline{\text{SCK}}$ = external clock input
T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY2}	800			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KL2}/ t_{KH2}	400			ns	
SI set-up time (against $\overline{\text{SCK}}$ ↑)	t_{SIK2}	100			ns	
SI hold time (against $\overline{\text{SCK}}$ ↑)	t_{KSI2}	400			ns	
$\overline{\text{SCK}}$ ↓ to SO output delay time	t_{KSO2}			300	ns	

SBI Mode (μPD75P308/P316) (see figures 18, 23)

$\overline{\text{SCK}}$ = internal clock output (master)
T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY3}	1600			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KH3}/ t_{KL3}	0.5 t_{KCY3}−50			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK3}	150			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI3}	0.5 t_{KCY3}			ns	
$\overline{\text{SCK}}$ ↓ to SB0, SB1 output delay time	t_{KSO3}	0		250	ns	
$\overline{\text{SCK}}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY3}			ns	
SB0, SB1 ↓to $\overline{\text{SCK}}$ ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	

SBI Mode (μPD75P308/P316) (see figures 18, 23)

$\overline{\text{SCK}}$ = external clock input (slave)
T_A = −10 to +70°C; V_{DD} = 5 V ± 5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY4}	800			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KH4}/ t_{KL4}	400			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK4}	100			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI4}	0.5 t_{KCY4}			ns	
$\overline{\text{SCK}}$ ↓ to SB0, SB1 output delay time	t_{KSO4}	0		300	ns	
$\overline{\text{SCK}}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY4}			ns	
SB0, SB1 ↓ to $\overline{\text{SCK}}$ ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	

Data Memory STOP Mode Low Voltage Data Retention Characteristics (μPD75P308/P316) (see figure 24)

T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		6.0	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization time (Note 2)	t_{WAIT}		(Notes 3, 4)		ms	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

(1) Excludes current in the pullup resistors.

(2) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillation is started. The interval timer can be used to delay the CPU from executing instructions using the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{xx} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{xx}$ (Approx 250 ms)
–	0	1	1	$2^{17}/f_{xx}$ (Approx 31.3 ms)
–	1	0	1	$2^{15}/f_{xx}$ (Approx 7.82 ms)
–	1	1	1	$2^{13}/f_{xx}$ (Approx 1.95 ms)

(3) Consult the manufacturer's resonator or crystal specification for this value.

(4) The interval timer will cause a delay of 2^{17}/fxx after a reset.

DC Programming Characteristics (μPD75P308/P316)

T_A = 25 ± 5°C; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	$0.7V_{DD}$		V_{DD}	V	All except X1, X2
	V_{IH2}	V_{DD}−0.5		V_{DD}	V	X1, X2
Low-level input voltage	V_{IL1}	0		$0.3V_{DD}$	V	All except X1, X2
	V_{IL2}	0		0.4	V	X1, X2
Input leakage current	I_{LI}			10	μA	V_{IN} = V_{IL} or V_{IH}
High-level output voltage	V_{OH}	V_{DD}−1.0			V	I_{OH} = −1 mA
Low-level output voltage	V_{OL}			0.4	V	I_{OL} = 1.6 mA
V_{DD} supply current	I_{DD}			30	mA	
V_{PP} supply current	I_{PP}			30	mA	MD0 = V_{IL}; MD1 = V_{IH}

Notes:

(1) V_{PP} must not exceed +13.5 V, including overshoot.

(2) V_{DD} must be applied before V_{PP} and V_{DD} should be removed after V_{PP} is removed.

AC Programming Characteristics (μPD75P308/P316) (see figures 26, 27)

T_A = 25 ± 5°C; V_{DD} = 6.0 ± 0.25 V; V_{PP} = 12.5 ± 0.3 V; V_{SS} = 0 V

Parameter	Symbol	(Note 1)	Min	Max	Unit	Conditions
Address setup time to MD0 ↓ (Note 2)	t_{AS}	t_{AS}	2		μs	
MD1 to MD0 ↓ setup	t_{M1S}	t_{OES}	2		μs	
Data to MD0 ↓ setup	t_{DS}	t_{DS}	2		μs	
Address hold from MD0 ↑ (Note 2)	t_{AH}	t_{AH}	2		μs	
Data hold from MD0 ↑	t_{DH}	t_{DH}	2		μs	
Data output float delay from MD0 ↑	t_{DF}	t_{DF}	0	130	ns	
V_{PP} setup to MD3 ↑	t_{VPS}	t_{VPS}	2		μs	
V_{DD} setup to MD3 ↑	t_{VDS}	t_{VCS}	2		μs	
Initialized program pulse width	t_{PW}	t_{PW}	0.95	1.05	ms	
Additional program pulse width	t_{OPW}	t_{OPW}	0.95	21	ms	
MD0 setup to MD1 ↑	t_{MOS}	t_{CES}	2		μs	
Data output delay from MD0 ↓	t_{DV}	t_{DV}		1	μs	MD0 = MD1 = V_{IL}
MD1 hold to MD0 ↑	t_{M1H}	t_{OEH}	2		μs	$t_{M1H} + t_{M1R} \geq 50\,\mu s$
MD1 recovery from MD0 ↓	t_{M1R}	t_{OR}	2		μs	$t_{M1H} + t_{M1R} \geq 50\,\mu s$
Program counter reset	t_{PCR}	–	10		μs	
X1 input low- and high-level width	t_{XH}, t_{XL}	–	0.125		μs	
X1 input frequency	f_X	–		4.19	MHz	
Initial mode set	t_I	–	2		μs	
MD3 setup to MD1 ↑	t_{M35}	–	2		μs	
MD3 hold to MD1 ↓	t_{M3H}	–	2		μs	
MD3 setup to MD0 ↓	t_{M3SR}	–	2		μs	During Program Read cycle
Address to data output delay time (Note 2)	t_{DAD}	t_{ACC}	2		μs	
Address to data output hold time (Note 2)	t_{HAD}	t_{OH}	0	130	ns	
MD3 output hold from MD0 ↑	t_{M3HR}	–	2		μs	
Data output float delay from MD3 ↓	t_{DFR}	–	2		μs	

Notes:

(1) These symbols correspond to those on the μPD27C256 EPROM.

(2) The internal address signal is incremented by one at the rising edge of the fourth X1 pulse; it is not connected to an external pin.

Figure 26. EPROM Program Memory Write Timing

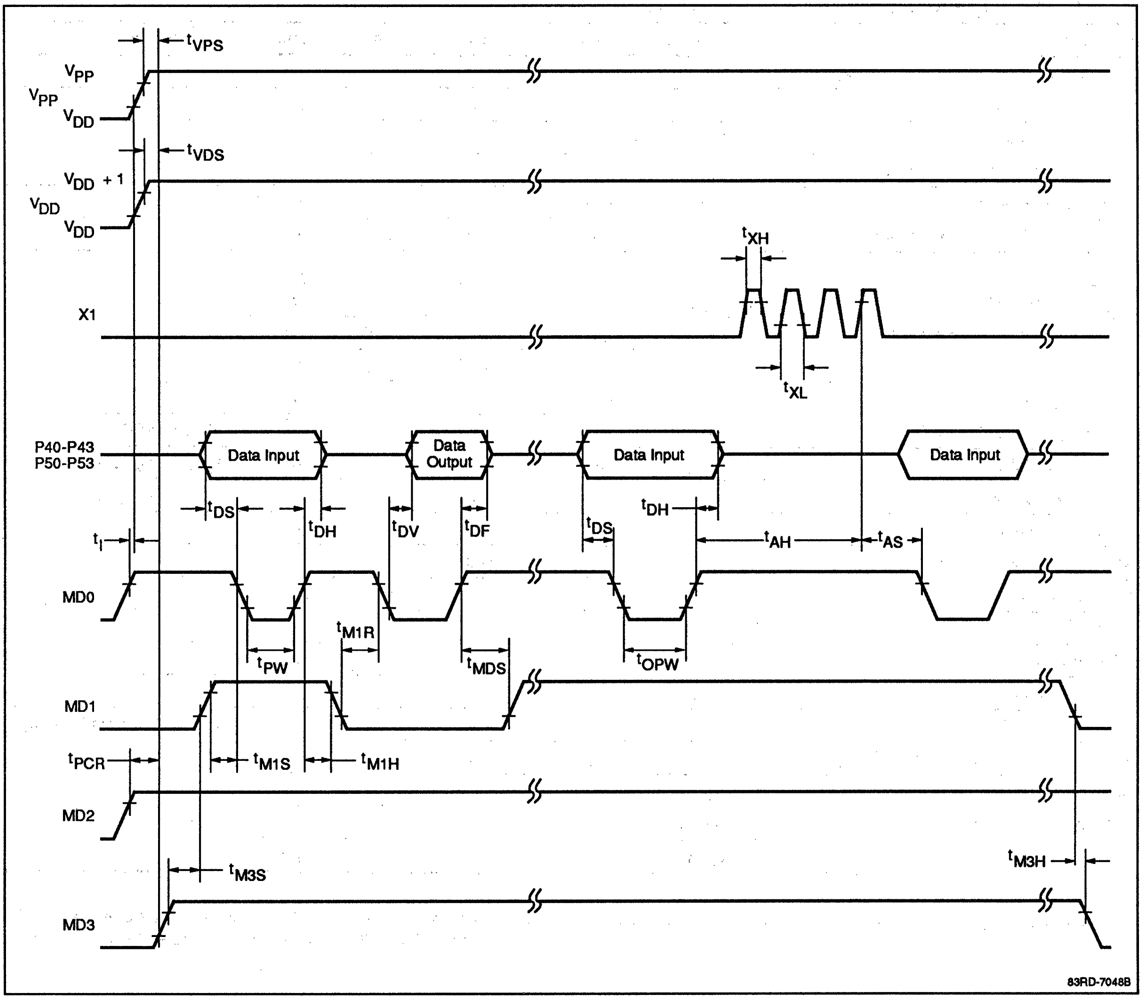
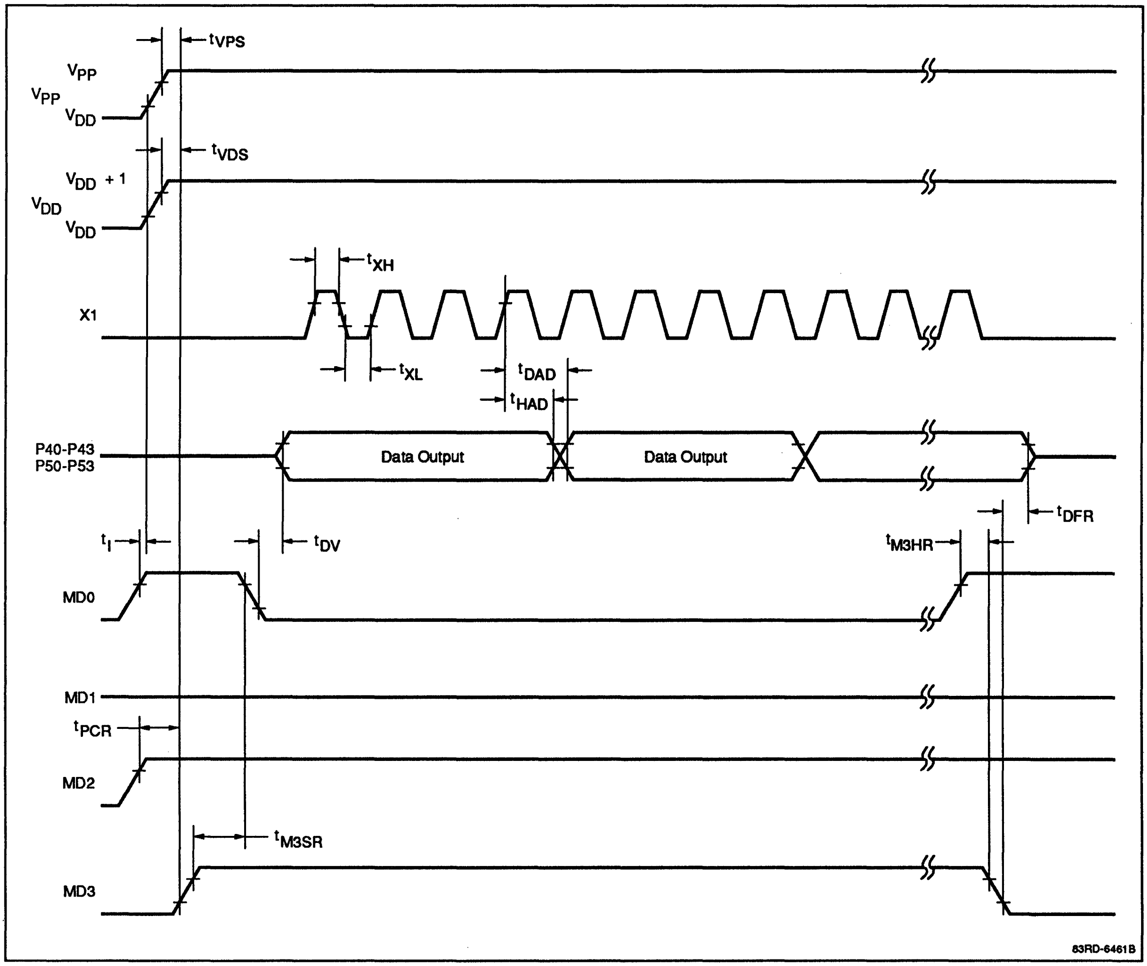


Figure 27. EPROM Program Memory Read Timing



NEC
NEC Electronics Inc.

μPD75328/75P328 4-Bit Microcomputers With LCD Controller/Driver and A/D Converter

Description

The μPD75328 and μPD75P328 are high performance single-chip CMOS microcomputers. They each contain a CPU, ROM, RAM, interval timer, timer/event counter, watch timer, LCD controller, A/D converter, subsystem clock, serial interface, I/O ports, and vectored interrupts. The instruction set allows the user to manipulate RAM data and I/O ports in one-, four-, and eight-bit units. The devices are suitable for controlling video cassette recorders (VCRs), telephones, and meters.

The μPD75P328 is a one-time programmable (OTP) version of the μPD75328.

Features

□ 103 Instructions
— Bit manipulation instructions
— Four-bit and eight-bit transfer instructions
— One-byte relative branch instructions
— GETI instruction converting 1 two-byte or three-byte instruction or 2 one-byte instructions into 1 one-byte instruction

□ Instruction execution cycles
— High-speed cycle: 0.95 μs/4.19 MHz
— Low-voltage cycle: 1.91 μs/4.19 MHz,
15.3 μs/4.19 MHz,

□ Program memory (μPD75328/75P328): 8064 bytes

□ Data memory (RAM)
— Allows operation on one, four, and eight bits
— 512 x four-bit data

□ Bit-sequential buffer
— 16-bit, bit manipulation memory

□ Eight four-bit registers

□ Accumulators
— One-bit accumulator (CY)
— Four-bit accumulator (A)
— Eight-bit accumulator (XA)

□ 24 I/O lines
— Twelve output ports that can directly drive LEDs (sink 15 mA rms)
— Eight N-channel, open-drain outputs with 10 V maximum

□ 12 Input only lines

□ One external event input

□ Timers
— One eight-bit basic interval timer
— One eight-bit timer/event counters
— One fourteen-bit watch timer

□ A/D converter
— Six-channel
— Eight-bit

□ LCD controller/driver
— Four common lines
— Twenty segment lines
— Operational modes
Static
Multiplexed 1/2 bias
Triplexed 1/2 or 1/3 bias
Quadriplexed 1/3 bias

□ Eight-bit serial interface
— Serial bus in (SBI) mode
— Two or three wire mode
Data transfer (MSB or LSB first)
Full duplex mode
Receive only mode

□ Vectored interrupts
— Three external interrupts
— Three internal interrupts
— Nine inputs generating one interrupt request

□ Standby modes
— Halt mode: stops CPU only
— Stop mode: stops main system clock

□ Mask options (Not available on the μPD75P328)
— Pull-up resistors for ports 4 and 5
— LCD resistor ladder
— Subsystem clock feed back resistor

□ Operates with oscillator or ceramic resonator

□ CMOS technology (at 5 V and 4.19 MHz)
— Normal operation: 2.5 mA (typical)
— Halt mode: 0.5 mA (typical)
— Stop mode: 0.1 μA (typical)

□ One-time programmable (OTP) version (μPD75P328) available

Ordering Information

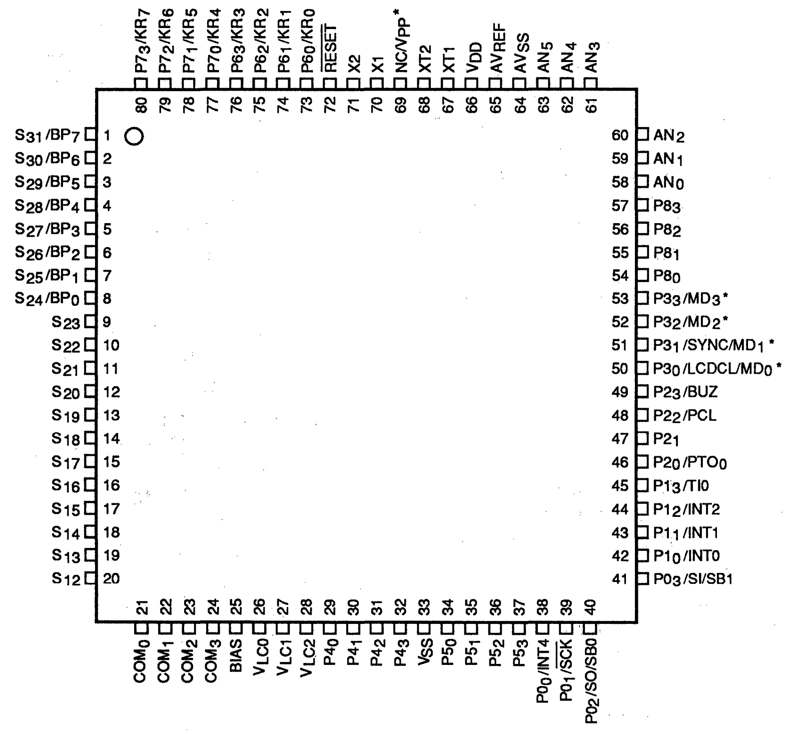
Part Number	Package	ROM
μPD75328GC-xxx-3B9	80-pin plastic QFP	Mask
μPD75P328GC-3B9	80-pin plastic QFP	OTP

Notes:

(1) xxx indicates ROM code suffix.

Pin Configuration

80-Pin Plastic QFP



* MD0-MD3 and Vpp are for programming the μPD75P328.

49NR-607B

Pin Identification

Symbol	Function
$P0_0$/INT4	Port 0 input; interrupt 4.
$P0_1$/$\overline{SCK}$	Port 0 input; serial clock ($\overline{SCK}$)
$P0_2$/SO/SB0	Port 0 input; serial data out
$P0_3$/SI/SB1	Port 0 input; serial data in
$P1_0$-P_2/INT0-INT2	Port 1 inputs; interrupts INT0-INT2
$P1_3$/TI0	Port 1 input; timer 0 input TI0
$P2_0$/PTO0	Four-bit I/O port 2 ($P2_0$-$P2_3$)/timer/event counter output 0 (PTO0)/port clock output (PCL)/buzzer output (BUZ)
$P2_1$	
$P2_2$/PCL	
$P2_3$/BUZ	
$P3_0$/LCDCL/ (MD_0)	Four-bit I/O port 3 ($P3_0$-$P3_3$)/LCD clock output (LCDCL)/sync output (SYNC)/OTP operation mode (MD_0-MD_3 for μPD75P328)
$P3_1$/SYNC/(MD_1)	
$P3_2$-$P3_3$/(MD_2-MD_3)	
$P4_0$-$P4_3$	Four-bit I/O port 4
$P5_0$-$P5_3$	Four-bit I/O port 5
$P6_0$-$P6_3$/ KR_0-KR_3	Four-bit I/O port 6 ($P6_0$-$P6_3$)/ key scan inputs 0-3 (KR_0-KR_3)
$P7_0$-$P7_3$/ KR_4-KR_7	Four-bit I/O port 7 ($P7_0$-$P7_3$)/ key scan inputs 4-7 (KR_4-KR_7)
$P8_0$-$P8_3$	Four-bit I/O port 8
S_{12}-S_{23}	LCD segment outputs 12-23
COM_0-COM_3	LCD common outputs 0-3
BP_0-BP_7/ S_{24}-S_{31}	Eight one-bit output ports (BP_0-BP_7)/ LCD segments 24-31 (S_{24}-S_{31})
V_{LC0}-V_{LC2}	LCD voltage drive level
BIAS	LCD power bias output
AN_0-AN_5	A/D converter inputs (AN_0-AN_5)
AV_{REF}	A/D converter reference voltage
AV_{SS}	A/D converter ground
NC (V_{PP})	No connection (V_{PP} for μPD75P328)
X2, X1	Main clock inputs
XT2, XT1	Subsystem clock inputs
$\overline{RESET}$	Reset input
V_{DD}	Positive power supply
V_{SS}	Ground

PIN FUNCTIONS

$P0_3$/SI/SB1, $P0_2$/SO/SB0, $P0_1$/$\overline{SCK}$, $P0_0$/INT4 (Port 0, INT4, Serial Interface)

Port 0 can be used as a four-bit input port. $P0_0$ can be used for INT4 which is an edge-triggered vectored interrupt triggered by a rising or falling edge. $P0_1$-$P0_3$ are also used for the serial interface in the SBI or 2/3 wire mode. The serial input (SI) and serial bus one (SB1), serial output (SO) and serial bus zero (SB0), and the serial clock ($\overline{SCK}$) make-up the serial interface. Port 0 is in the input mode at reset.

$P1_3$/TI0, $P1_2$/INT2, $P1_1$/INT1, $P1_0$/INT0 (Port 1, Edge-Triggered Interrupts, Timer Input)

Port 1 can be used as a four-bit input port. INT0 and INT1 are edge-triggered vectored interrupts. INT2 is an edge-triggered input which generates an input request, but does not cause an interrupt. TI0 is an input clock to the timer/event counter and is used to count external events. Port 1 is in the input mode at reset.

$P2_3$/BUZ, $P2_2$/PCL, $P2_1$, $P2_0$/PTO_0 (Port 2, Clock, Buzzer, and Timer/Event Counter Outputs)

Port 2 can be used as a four-bit I/O port. When used as an output port, the output data is latched. When used as an input port, the outputs are high-impedance. $P2_0$ is used to output the timer/event counter flip/flop signal TOUT. $P2_2$ is used to output the PCL clock from the clock generator and $P2_3$ is used to output square waves for a buzzer. Port 2 is in the input mode at reset.

$P3_3$/MD_3, $P3_2$/MD_2, $P3_1$/SYNC/MD_1, $P3_0$/LCDCL/MD_0 (Port 3, LCD Outputs, OTP Operation Mode for μPD75P328)

Port 3 is a programmable four-bit I/O port. Each bit can be independently programmed to be either an input or an output. The port has latched outputs and can directly drive LEDs. $P3_0$ and $P3_1$ can be used to output the LCD clock and LCD sync signal, respectively. MD_0 through MD_3 are used for the μPD75P328 OTP program memory write and verify mode to select the operation mode. Port 3 is in the input mode at reset.

$P4_0$-$P4_3$ (Port 4)

Port 4 is a four-bit I/O port. Ports 4 and 5 can be paired together to function as one 8-bit port. Port 4 has latched outputs and can directly drive LEDs. The outputs are N-channel, open drain, 10 V max. An internal pull-up resistor is available as a mask option. Port 4 is in the input mode at reset.

$P5_0$-$P5_3$ (Port 5)

Port 5 is a four-bit I/O port.Ports 4 and 5 can be paired together to function as one 8-bit port. Port 5 has latched outputs and can directly drive LEDs. The outputs are N-channel, open drain, 10 V max. An internal pull-up resistor is available as a mask option. Port 5 is in the input mode at reset.

$P6_3/KR_3$-$P6_0/KR_0$ (Port 6, Edge Detection of KR_0-KR_3)

Port 6 is a programmable four-bit I/O port. Port 6 has latched outputs and each bit can be independently programmed to be either an input or an output. Ports 6 and 7 can be paired together to function as one 8-bit port. Port 6 can be used to detect the falling edge of inputs KR_0- KR_3. Port 6 is in the input mode at reset.

$P7_3/KR_7$-$P7_0/KR_4$ (Port 7, Edge Detection of KR_4-KR_7)

Port 7 is a four-bit I/O port. The port has latched outputs. Ports 6 and 7 can be paired together to function as one 8-bit port. Port 7 can be used to detect the falling edge of inputs KR_4 through KR_7. Port 7 is in the input mode at reset.

$P8_0$-$P8_3$ (Port 8)

Port 8 is a four-bit I/O port with latched outputs. Port 8 is in the input mode at reset.

S_{12}-S_{23} (LCD Segment Outputs)

S_{12} through S_{23} are the LCD segment output signals which directly drive the LCD segment inputs.

COM_0-COM_3 (LCD Common Outputs)

COM_0 through COM_3 are the LCD common output signals, which directly drive the common LCD inputs.

BP_0/S_{24}-BP_7/S_{31} (One-Bit Output Ports, Segment Outputs)

BP_0 through BP_7 can be used as 8 one-bit ports or as additional LCD segment outputs. As LCD segment outputs, they are selectable in four-bit units: BP_0-BP_3/S_{24} -S_{27} or BP_4-BP_7/S_{28}-S_{31}.

V_{LC0}-V_{LC2} (LCD Voltage Levels)

V_{LC0} through V_{LC2} are used to set the LCD drive levels. These pins are outputs when the internal resistor ladder mask option is selected. When an external resistor ladder is used, the pins are inputs and must be connected to set the LCD drive levels.

BIAS (Bias Output)

BIAS output is used with V_{LC0} through V_{LC2} to set the static, 1/2 bias, or 1/3 bias levels.

AN_0-AN_5 (A/D Converter Inputs)

AN_0 through AN_5 are inputs to the six-channel eight-bit A/D converter.

AV_{REF} (A/D Converter Reference Voltage)

AV_{REF} is used to supply a reference voltage to the A/D converter.

AV_{SS} (Analog Ground)

AV_{SS} is the analog ground pin for the A/D converter.

NC/V_{PP} (No Connection, Programming Pin)

This pin is connected when using the μPD75P328 and may be left unconnected when using the μPD75328. When programming a device, the programming voltage V_{PP} is used during the EPROM write/verify cycles. When a device is not being programmed, this pin should be connected to V_{DD}.

X2, X1 (Main System Clock Inputs)

X1 and X2 are the main system clock inputs. The clock is controlled by a crystal or a ceramic oscillator. An external logic signal can be used as a clock source. See figure 1.

XT2, XT1 (Subsystem Clock Inputs)

XT1 and XT2 are the subsystem clock source inputs. These pins use a crystal, ceramic oscillator, or a logic signal as an input. See figure 2.

$\overline{\text{RESET}}$ (Reset)

System reset input pin (active low).

V_{DD} (Power Supply)

Positive power supply.

V_{SS} (Ground)

System ground.

Block Diagram

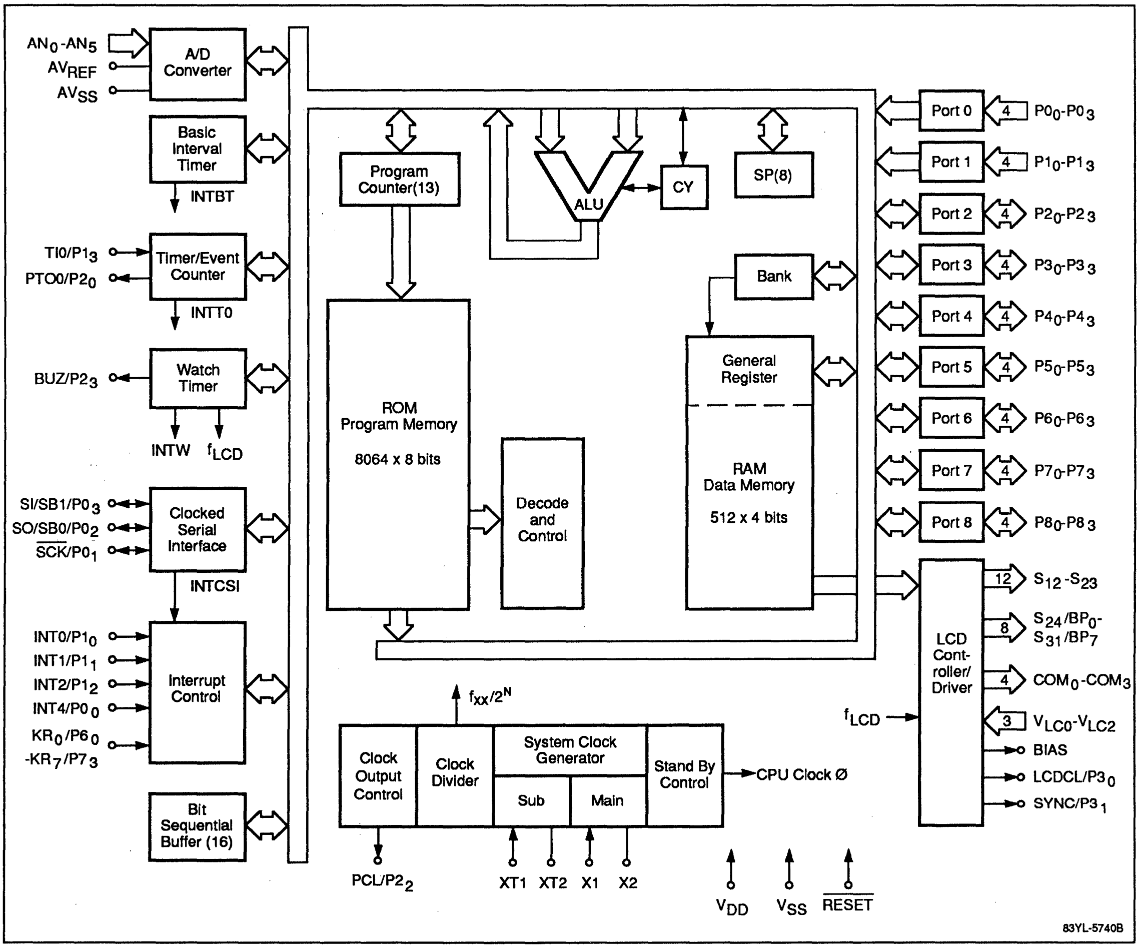
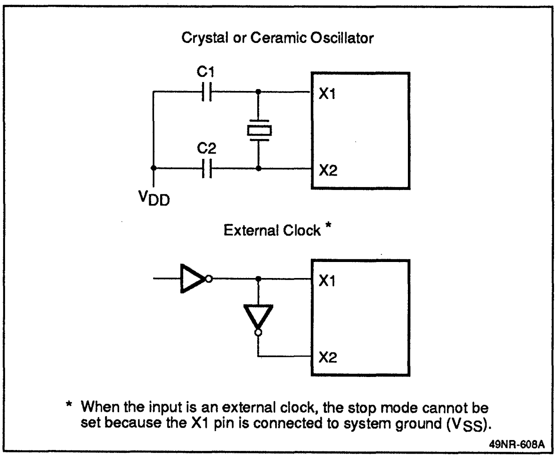
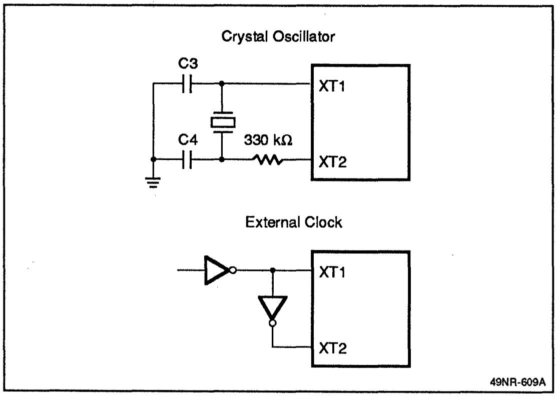


Figure 1. Main System Clock Configurations



* When the input is an external clock, the stop mode cannot be set because the X1 pin is connected to system ground (V_{SS}).

Figure 2. Subsystem Clock Configurations



μPD75328 AND μPD75P328 DIFFERENCES

The μPD75P328 contain a one-time programmable (OTP) program memory and the μPD75328 contains a mask ROM program memory. The μPD75328 and μPD75P328 differ only in their program memory and mask options, but otherwise are identical in their CPU functions and internal hardware. Their differences are shown in table 1.

Table 1. Differences between μPD75328 and μPD75P328

Item	μPD75P328	μPD75328
Program memory	One-time EPROM	Mask ROM
	8064 x 8 bits	8064 x 8 bits
	0000H - 1F7FH	0000H - 1F7FH
Ports 4 and 5 pull-up resistor	Not offered	Mask option
LCD resistor ladder	Not offered	Mask option
Subsystem clock oscillating feed back resistor	On-chip	Mask option
Programming pin connections	V_{PP} pin and one-time EPROM program pins	None
Operating supply voltage range	5 V ±5%	2.7 to 6.0 V
Package	80-pin plastic QFP with bent leads	80-pin plastic QFP with bent leads

μPD75328 AND μPD75308 COMPARISON

The μPD75328 provides 7 CMOS I/O ports; the μPD75308 has 6. The μPD75308 does not contain an A/D converter. However, the μPD75308 does have an LCD controller with 32 segment outputs versus 20 for the μPD75328. Table 2 compares the features of the two devices.

Table 2. μPD75328 and μPD75308 Features Comparison

Item	μPD75328	μPD75308
ROM	8064 bytes	Same as μPD75328
RAM	512 x 4 bits	Same as μPD75328
General purpose register	4 bits x 8 or 8 bits x 4	Same as μPD75328
Instruction cycle	Selectable from .95 μs/1.91 μs/15.3 μs (4.19 MHz) and 122 μs (32 kHz)	Same as μPD75328
Input/ output port	28 I/O lines 8 input only 8 output only	24 I/O lines 8 input 8 output only
CMOS input port	8 lines shared with INT/SIO. Can be pulled by software, except for $P0_0$	Same as μPD75328
CMOS I/O port	20 lines (4 lines can directly drive LED). Can be pulled by software, except for $P0_0$	16 lines (4 lines can directly drive LED). Can be pulled by software, except for $P0_0$
CMOS output port	4/8 lines (shared segment output and selected by software)	Same as μPD75328
N-channel I/O port	8 lines can be pulled up by mask option, directly drive LED, have continuous 10 V applied	Same as μPD75328
Timer counters	Timer/event counter, basic interval timer, clock timer	Same as μPD75328
Serial interface	NEC-SBI serial bus interface	Same as μPD75328
	Normal clock synchronized serial interface	Same as μPD75328
A/D converter	6-channel analog input, 8-bit precision	Not offered
Interrupt	6 vector interrupts (3 external and 3 internal)	Same as μPD75328
	2 test inputs (1 external and 1 internal)	Same as μPD75328
	Parallel edge detection for key scan input	Same as μPD75328
Instruction set	Bit: data set/reset/test/ boolean operation	Same as μPD75328
	4-bit: data transfer/ arithmetic/increment/ decrement/ comparison	Same as μPD75328
	8-bit: data transfer	Same as μPD75328
LCD controller/ driver	20 segment outputs	32 segment outputs
	4 common outputs	Same as μPD75328
	Display mode: Static Multiplexed Triplexed Quadriplexed	Same as μPD75328
	Resistor ladder network for LCD drive voltage supply (mask option)	Same as μPD75328
Operating voltage	2.7 to 6.0 V	Same as μPD75328
Package	80-pin plastic QFP (.65 pitch)	80-pin plastic QFP (.8 pitch)

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings, µPD75328/P328

$T_A = 25°C$

Supply voltage, V_{DD}	−0.3 to +7.0 V
Input voltage, V_{I1} (except ports 4 and 5)	−0.3 to V_{DD}+0.3 V
Input voltage, V_{I2} (µPD75328 only) (ports 4 and 5 with internal pull-up resistor)	−0.3 to V_{DD}+0.3 V
Input voltage, V_{I2} (ports 4 and 5 open drain)	−0.3 to +11 V
Output voltage, V_O	−0.3 to V_{DD}+0.3 V
High-level output current, I_{OH} (single pin)	−15 mA
High-level output current, I_{OH} (all pins)	−30 mA
Low-level output current, I_{OL} (single pin, peak value)	30 mA
Low-level output current, I_{OL} (single pin) (Note 1)	15 mA rms
Low-level output current, I_{OL} (ports 0, 2, 3, 5, and 8)	100 mA peak
Low-level output current, I_{OL} (ports 0, 2, 3, 5, and 8) (Note 1)	60 mA rms
Low-level output current, I_{OL} (ports 4, 6, and 7)	100 mA peak
Low-level output current, I_{OL} (ports 4, 6, and 7) (Note 1)	60 mA rms
Operating temperature, T_{OPT} (µPD75328 only)	−40 to +85°C
Operating temperature, T_{OPT} (µPD75P328 only)	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Notes:

(1) Root mean square (rms) = peak value x (duty factor)$^{1/2}$.

Capacitance, µPD75328/P328

T_A = +25°C; V_{DD} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_{IN}	15	pF	f = 1 MHz; unmeasured pins must be at 0 V
Output capacitance	C_{OUT}	15	pF	
Input/Output capacitance	C_{IO}	15	pF	

CLOCK OSCILLATOR SPECIFICATIONS

Main System Clock Oscillator, µPD75328/P328

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V (µPD75328); T_A = −10 to +70°C; V_{DD} = 5 V ±5% (µPD75P328)

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Figure 3)	Frequency (Note 1)	f_{XX}	1.0		5.0	MHz	
	Stabilization time (Note 2)				4.0 (Note 3)	ms	After V_{DD} reaches the minimum value of the oscillator operating voltage range.
Crystal resonator (Figure 3)	Frequency (Note 1)	f_{XX}	1.0	4.19	5.0	MHz	
	Stabilization time (Note 2)				10 (Note 3)	ms	V_{DD} = 4.5 to V_{DD} max
					30 (Note 3)	ms	µPD75328 only.
External clock (Figure 3)	X1 input frequency (Note 1)	f_{XX}	1.0		5.0	MHz	
	X1 input high/low level width	t_{XH}, t_{XL}	100		500	ns	

Notes:

(1) Oscillator and X1 input frequencies are shown only to present the oscillator characteristics.
Refer to the AC characteristics for instruction execution time.

(2) Time required for oscillator to stabilize after V_{DD} min is reached or time after release of STOP mode.

(3) Values shown pertain to the recommended oscillators. For oscillators not listed under recommended oscillators and circuit constants, consult the vendor's specification.

Subsystem Clock Oscillator, µPD75328/P328

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V (µPD75328); T_A = −10 to +70°C; V_{DD} = 5 V ±5% (µPD75P328)

Type	Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Crystal resonator (Figure 4)	Frequency	f_{XT}	32	32.768	35	kHz	
	Stabilization Time			1.0	2.0	s	V_{DD} = 4.5 toV_{DD} max
					10	s	µPD75328 only
External clock (Figure 4)	XT1 input frequency	f_{XT}	32		100	kHz	
	XT1 input high/low level width	t_{XTH}, t_{XHL}	5.0		15	µs	

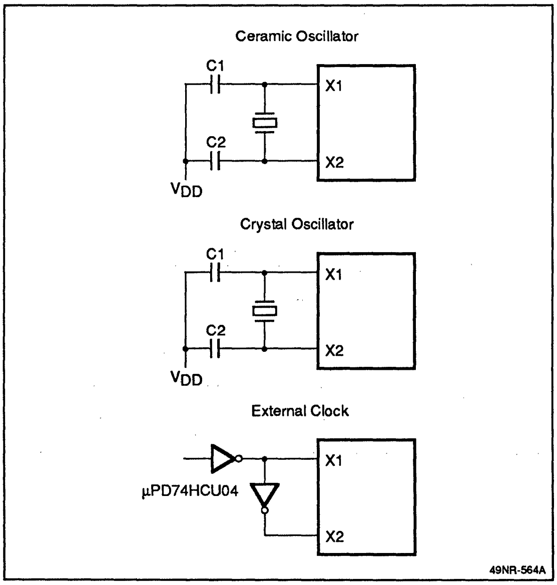


Figure 3. Recommended Main System Clock Circuit Configurations

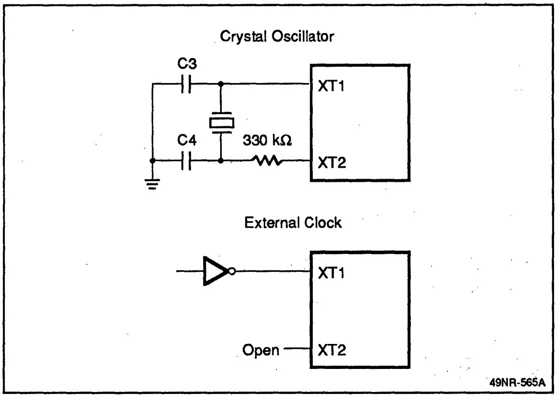


Figure 4. Recommended Subsystem Clock Circuit Configurations

Recommended Ceramic Oscillators and Circuit Constants for Main System Clock, μPD75328/P328

T_A = −40 to +85°C (Note 1)

Manufacturer	Part Number (Note 2)	Frequency (MHz)	C1 (pF)	C2 (pF)	V_{DD} Min (V)	V_{DD} Max (V)
Murata Manufacturing Company LTD. (Note 1)	CSAx.xxMG093	2.00 - 2.44	30	30	2.7	6.0
	CSTx.xxMG093		None	None	2.7	6.0
	CSAx.xxMGU	2.45 - 5.00	30	30	2.7	6.0
	CSTx.xxMGU		None	None	2.7	6.0
	CSAx.xxMG	2.00 - 5.00	30	30	3.0	6.0
	CSTx.xxMG		None	None	3.0	6.0
Kyoto Ceramic Company. LTD. (Note 1)	KBR-2.0MS	2.00	47	47	2.7	6.0
	KBR-4.0MS	4.00	33	33	2.7	6.0
	KBR-5.0M	5.00	33	33	3.0	6.0

Notes:

(1) Although the oscillators have a wider operating voltage and temperature range than the μPD75P328, the μPD75P328 is still limited in its operating voltage and temperature range to 5 V ± 5%, T_A = −10 to +70°C.

(2) x.xx indicates frequency.

Recommended Crystal Oscillators and Circuit Constants for Main System Clock, μPD75328/P328

T_A = −20 to +70°C (Note 1)

Manufacturer	Part Number	Frequency (MHz)	C1 (pF) (Note 2)	C2 (pF)	Min (V)	Max (V)
Kinseko (Note 1)	HC-18U	2.0	22	22	2.7	6.0
	HC-49U	4.19	22	22	2.7	6.0
	HC-43U	4.91	22	22	2.7	6.0

Notes:

(1) Although the oscillators have a wider operating voltage and temperature range than the μPD75P328, the μPD75P328 is still limited in its operating voltage and temperature range to 5 V ±5%, T_A = −10 to +70°C.

(2) Oscillator frequency adjustment range: C1 = 15 to 33 pF.

Recommended Crystal Oscillator and Circuit Constants for Subsystem Clock, μPD75328/P328

T_A = −10 to +60°C (Note 1)

Manufacturer	Part Number	Frequency (kHz)	C3 (pF)	C4 (pF)	V_{DD} Min (V)	V_{DD} Max (V)
Kinseki (Note 1)	P3	32.768	22 (Note 2)	22	2.7	6.0

Notes:

(1) Although the oscillators have a wider operating voltage and temperature range than the μPD75P328, the μPD75P328 is still limited in its operating voltage and temperature range to 5 V ± 5%, T_A = −10 to +70°C.

(2) Oscillator frequency adjustment range: C3 = 3 to 30 pF.

ELECTRICAL CHARACTERISTICS

DC Characteristics, μPD75328

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Ports 2, 3, and 8.
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	Ports 0, 1, 6, 7, and $\overline{\text{RESET}}$ pin.
	V_{IH3}	0.7 V_{DD}		V_{DD}	V	Ports 4 and 5 with internal pull-up resistor.
		0.7 V_{DD}		10	V	Ports 4 and 5 with open drain.
	V_{IH4}	V_{DD}−0.5		V_{DD}	V	X1, X2, and XT1.
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Ports 2, 3, 4, 5, and 8.
	V_{IL2}	0		0.2 V_{DD}	V	Ports 0, 1, 6, 7, and $\overline{\text{RESET}}$ pin.
	V_{IL3}	0		0.4	V	X1, X2, and XT1.
High-level output voltage	V_{OH1}	V_{DD}−1.0			V	Ports 0, 2, 3, 6, 7, 8, and BIAS. V_{DD} = 4.5 to 6.0 V. I_{OH} = −1 mA.
		V_{DD}−0.5			V	Ports 0, 2, 3, 6, 7, 8, and BIAS. I_{OH} = −100 μA.
	V_{OH2}	V_{DD}−2.0			V	BP_0-BP_7 (two I_{OH} outputs). V_{DD} = 4.5 to 6.0 V. I_{OH} = −100 μA.
		V_{DD}−1.0			V	BP_0-BP_7 (two I_{OH} outputs). I_{OH} = −50 μA.
Low-level output voltage	V_{OL1}		0.4	2.0	V	Ports 3, 4, and 5 (V_{DD} = 4.5 to 6.0 V, I_{OL} = 15 mA).
				0.4	V	Ports 0, 2, 3, 4, 5, 6, 7, and 8. V_{DD} = 4.5 to 6.0 V. I_{OL} = 1.6 mA.
				0.5	V	Ports 0, 2, 3, 4, 5, 6, 7, and 8. I_{OL} = 400 μA.
				0.2 V_{DD}	V	SB0 and SB1. Open drain with pull-up resistor ≥ 1 kΩ
	V_{OL2}			1.0	V	BP_0-BP_7 (two I_{OL} outputs). V_{DD} = 4.5 to 6.0 V. I_{OL} = 100 μA.
				1.0	V	BP_0-BP_7 (two I_{OL} outputs). I_{OL} = 50 μA.
High-level input leakage current	I_{LIH1}			3	μA	V_{IN} = V_{DD}. Other than X1, X2, and XT1.
	I_{LIH2}			20	μA	V_{IN} = V_{DD}. X1, X2, and XT1.
	I_{LIH3}			20	μA	V_{IN} = 10 V. Ports 4 and 5 with open drain.
Low-level input leakage current	I_{LIL1}			−3	μA	V_{IN} = 0 V. Other than X1, X2, XT1.
	I_{LIL2}			−20	μA	V_{IN} = 0 V. X1, X2, and XT1.
High-level output leakage current	I_{LOH1}			3	μA	V_{OUT} = V_{DD}. Other than ports 4 and 5 with open drain.
	I_{LOH2}			20	μA	V_{OUT} = 10 V. Ports 4 and 5 with open drain.
Low-level output leakage current	I_{LOL}			−3	μA	V_{OUT} = 0 V.
Internal pull-up resistor	R_{L1}	15	40	80	kΩ	Ports 0, 1, 2, 3, 6, 7,and 8 (except $P0_0$). V_{IN} = 0 V. V_{DD} = 5.0 V ±10%.
		30		300	kΩ	Ports 0, 1, 2, 3, 6, 7,and 8 (except $P0_0$). V_{IN} = 0 V. V_{DD} = 3.0 V ±10%.
	RL_2	15	40	70	kΩ	Ports 4 and 5. V_{OUT} = V_{DD} −2.0 V. V_{DD} = 5.0 V ±10%.
		10		60	kΩ	Ports 4 and 5. V_{OUT} = V_{DD} −2.0 V. V_{DD} = 3.0 V ±10%.
LCD drive voltage	V_{LCD}	2.5		V_{DD}	V	
LCD step-down resistor	R_{LCD}	60	100	140	kΩ	

DC Characteristics, μPD75328 (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
LCD common output voltage deviation (Note 1)	V_{ODC}	0		±0.2	V	$I_0 = \pm 5$ μA. $V_{LCD0} = V_{LCD}$, $V_{LCD1} = V_{LCD}$x2/3, $V_{LCD2} = V_{LCD}$x1/3, 2.7 V ≤ V_{LCD} ≤ V_{DD}.
LCD segment output voltage deviation (Note 1)	V_{ODS}	0		±0.2	V	$I_0 = \pm 1$ μA. $V_{LCD0} = V_{LCD}$, $V_{LCD1} = V_{LCD}$x2/3, $V_{LCD2} = V_{LCD}$x1/3, 2.7 V ≤ V_{LCD} ≤ V_{DD}.
Supply current (Note 2)	I_{DD1}		2.5	8	mA	4.19-MHz crystal oscillator (C1 = C2 = 22 pF). V_{DD} = 5 V ±10% (Note 3).
			0.35	1.2	mA	4.19-MHz crystal oscillator (C1 = C2 = 22 pF). V_{DD} = 3 V ±10% (Note 4).
	I_{DD2}		500	1500	μA	4.19-MHz crystal oscillator (C1 = C2 = 22 pF). HALT mode. V_{DD} = 5 V ±10%.
			150	450	μA	4.19-MHz crystal oscillator (C1 = C2 = 22 pF). HALT mode. V_{DD} = 3 V ±10%.
	I_{DD3}		30	90	μA	32-kHz crystal resonator. Operation mode. V_{DD} = 3 V ±10%. Main system clock stopped.
	I_{DD4}		5	15	μA	32-kHz crystal resonator. HALT mode. V_{DD} = 3 V ±10%. Main system clock stopped.
	I_{DD5}		0.5	20	μA	XT1 = 0 V. STOP mode. V_{DD} = 5 V±10%.
			0.1	10	μA	XT1 = 0 V. STOP mode. V_{DD} = 3 V ±10%.
			0.1	5	μA	XT1 = 0 V. STOP mode. V_{DD} = 3 V ±10%. T_A = 25°C.

Notes:

(1) Voltage deviation is the difference between the ideal segment or common output value VLCDn (n = 0, 1, or 2) and the output voltage.

(2) Current values for the internal pull-up resistor and the LCD step-down resistor are not included.

(3) When the processor clock control register (PCC) is set to 0011H and the CPU is operated in the high-speed mode.

(4) When the processor clock control register (PCC) is set to 0000H and the CPU is operated in the low-speed mode.

AC Characteristics, μPD75328

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

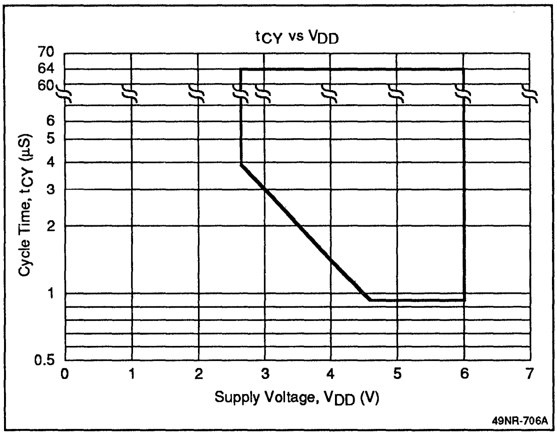
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Note 1)	t_{CY}	0.95		64	μs	Operation with main system clock. V_{DD} = 4.5 to 6.0 V
		3.8		64	μs	Operation with main system clock.
		114	122	125	μs	Operation with subsystem clock.
TIO input frequency	f_{TI}	0		1	MHz	V_{DD} = 4.5 to 6.0 V
		0		275	kHz	
TIO input low- and high-level width	t_{IL}, t_{IH}	0.48			μs	V_{DD} = 4.5 to 6.0 V
		1.8			μs	
Interrupt inputs low- and high-level width	t_{INTL}, t_{INTH}	(Note 2)			μs	INT0
		10			μs	INT1, 2, 4
		10			μs	KR_0-KR_7
$\overline{\text{RESET}}$ low-level width	t_{RSL}	10			μs	

Notes:

(1) Cycle time (minimum instruction execution time) is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See figure 5.

(2) $2t_{CY}$ or $128/f_{xx}$, depending on the setting of the interrupt mode register (IM0).

Figure 5. μPD75328 t_{CY} vs V_{DD} with Main System Clock



A/D Converter Characteristics, μPD75328

T_A = −10 to +85°C; V_{DD} = 3.5 to 6.0 V; AV_{SS} = V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8	8	8	bits	
Absolute accuracy (Note 1)				±1.5	LSB	2.5 V ≤ AV_{REF} ≤ V_{DD}
Conversion time	t_{CONV}			$168/f_X$	s	(Note 2)
Sampling time	t_{SAMP}			$44/f_X$	s	(Note 3)
Analog input voltage	V_{IAN}	AV_{SS}		AV_{REF}	V	
Analog input impedance	R_{AN}		1000		MΩ	
AV_{REF} current	I_{REF}		1.0	2.0	mA	

Notes:

(1) The absolute accuracy does not include the quantization error (±1/2 LSB).

(2) The total conversion time until EOC = 1 is 40.1 μs at f_{XX} = 4.19 MHz.

(3) The time until completion of sampling is 10.5 μs (f_{XX} = 4.19 MHz). Note that the sampling time value is included in the total conversion time value.

(4) For detailed A/D converter information refer to the user's manual.

Serial Transfer Operation, μPD75328
2-Line/3-Line Serial I/O Mode ($\overline{SCK}$, Internal Clock Output)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{SCK}$ low- and high-level width	t_{KL1}, t_{KH1}	$0.5t_{KCY1}-50$			ns	V_{DD} = 4.5 to 6.0 V
		$0.5t_{KCY1}-150$			ns	
SI vs $\overline{SCK}$ ↑ setup time	t_{SIK1}	150			ns	
SI vs $\overline{SCK}$ ↑ hold time	t_{KSI1}	400			ns	
$\overline{SCK}$ ↓ to SO output delay time	t_{KSO1}			250	ns	V_{DD} = 4.5 to 6.0 V
				1000	ns	

Serial Transfer Operation, μPD75328
2-Line/3-Line Serial I/O Mode ($\overline{\text{SCK}}$, External Clock Input)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY2}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KL2}, t_{KH2}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SI vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK2}	100			ns	
SI vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI2}	400			ns	
$\overline{\text{SCK}}$ ↓ to SO output delay time	t_{KSO2}			300	ns	V_{DD} = 4.5 to 6.0 V
				1000	ns	

SBI Mode, μPD75328
$\overline{\text{SCK}}$, Internal Clock Output (Master)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY3}	1600			ns	V_{DD} = 4.5 to 6.0 V
		3800			ns	
$\overline{\text{SCK}}$ width	t_{KL3}, t_{KH3}	$0.5t_{KCY3}$ −50			ns	V_{DD} = 4.5 to 6.0 V
		$0.5t_{KCY3}$ −150			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK3}	150			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI3}	0.5 t_{KCY3}			ns	
$\overline{\text{SCK}}$ ↓ to SBO, SB1 output delay time	t_{KSO3}	0		250	ns	V_{DD} = 4.5 to 6.0 V
		0		1000	ns	
$\overline{\text{SCK}}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY3}			ns	
SB0, SB1 ↓ to $\overline{\text{SCK}}$ ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	

SBI Mode, μPD75328
$\overline{\text{SCK}}$, External Clock Input (Slave)

T_A = −40 to +85°C; V_{DD} = 2.7 to 6.0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY4}	800			ns	V_{DD} = 4.5 to 6.0 V
		3200			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KL4}, t_{KH4}	400			ns	V_{DD} = 4.5 to 6.0 V
		1600			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK4}	100			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI4}	$0.5t_{KCY4}$			ns	
$\overline{\text{SCK}}$ ↓ to SBO, SB1 output delay time	t_{KSO4}	0		300	ns	V_{DD} = 4.5 to 6.0 V
		0		1000	ns	
$\overline{\text{SCK}}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY4}			ns	
SB0, SB1 ↓ to $\overline{\text{SCK}}$ ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	

Data Memory STOP Mode Low Voltage Data Retention Characteristics, μPD75328

T_A = −40 to +85°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		6.0	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V
Release signal set time	t_{SREL}	0			μs	
Oscillation stabilization time (Note 2)	t_{WAIT}		(Notes 3, 4)		ms	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

(1) Internal pull-up resistor current and LCD resistor ladder (mask option) current are not included.

(2) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillator is stabilizing. The interval timer can be used to delay the CPU from executing instructions by using the basic interval timer mode register (BTM) according to the following table:

BTM3	BTM2	BTM1	BTM0	WAIT time (f_{XX} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{XX}$ (approximately 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (approximately 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (approximately 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (approximately 1.95 ms)

(3) Consult the manufacturer's resonator specification sheet for this value.

(4) The interval timer will cause a delay of $2^{17}/f_{XX}$ seconds after a reset.

ELECTRICAL CHARACTERISTICS

DC Characteristics, μPD75P328

T_A = −10 to +70°C; V_{DD} = 5 V ±5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
High-level input voltage	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Ports 2, 3, and 8.
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	Ports 0, 1, 6, 7, and $\overline{\text{RESET}}$ pin.
	V_{IH3}	0.7 V_{DD}		10	V	Ports 4 and 5 with open drain.
	V_{IH4}	V_{DD}-0.5		V_{DD}	V	X1, X2, and XT1.
Low-level input voltage	V_{IL1}	0		0.3 V_{DD}	V	Ports 2, 3, 4, 5, and 8.
	V_{IL2}	0		0.2 V_{DD}	V	Ports 0, 1, 6, 7, and $\overline{\text{RESET}}$ pin.
	V_{IL3}	0		0.4	V	X1, X2, and XT1.
High-level output voltage	V_{OH1}	V_{DD} −1.0			V	Ports 0, 2, 3, 6, 7, 8, and BIAS. I_{OH} = −1 mA.
	V_{OH2}	V_{DD} −2.0			V	BP_0-BP_7. I_{OH} = −100 μA, two I_{OH} outputs
Low-level output voltage	V_{OL1}		0.4	2.0	V	Ports 3, 4, and 5 (I_{OL} = 15 mA).
				0.4	V	Ports 0, 2, 3, 4, 5, 6, 7, and 8. I_{OL} = 1.6 mA.
	V_{OL2}			0.2 V_{DD}	V	SB0 and SB1. Open drain with pull-up resistor ≥ 1 kΩ.
	V_{OL3}			1.0	V	BP_0 - BP_7. I_{OL} = 100 μA. Two I_{OL} outputs
High-level input leakage current	I_{LIH1}			3	μA	V_{IN} = V_{DD}. Other than X1, X2, XT1.
	I_{LIH2}			20	μA	V_{IN} = V_{DD}. X1, X2, and XT1.
	I_{LIH3}			20	μA	V_{IN} = 10 V. Ports 4 and 5.
Low-level input leakage current	I_{LIL1}			−3	μA	V_{IN} = 0 V. Other than X1, X2, XT1.
	I_{LIL2}			−20	μA	V_{IN} = 0 V. X1, X2, and XT1.
High-level output leakage current	I_{LOH1}			3	μA	V_{OUT} = V_{DD}. Other than ports 4 and 5.
	I_{LOH2}			20	μA	V_{OUT} = 10 V. Ports 4 and 5.
Low-level output leakage current	I_{LOL}			−3	μA	V_{OUT} = 0 V
Internal pull-up resistor	R_{L1}	15	40	80	kΩ	Ports 0, 1, 2, 3, 6, 7, and 8 (except $P0_0$). V_{IN} = 0 V.

DC Characteristics, μPD75P328 (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
LCD drive voltage	V_{LCD}	2.5		V_{DD}	V	
LCD common output voltage deviation (Note 1)	V_{ODC}	0		±0.2	V	$I_0 = \pm 5$ μA, $V_{LCD0} = V_{LCD}$, $V_{LCD1} = V_{LCD}$x2/3, $V_{LCD2} = V_{LCD}$x1/3, 2.5 ≤ V_{LCD} ≤ 5.25
LCD segment output voltage deviation (Note 1)	V_{ODS}	0		±0.2	V	$I_0 = \pm 1$ μA, $V_{LCD0} = V_{LCD}$, $V_{LCD1} = V_{LCD}$x2/3, $V_{LCD2} = V_{LCD}$x1/3, 2.5 ≤ V_{LCD} ≤ 5.25
Supply current (Note 2)	I_{DD1}		5	15	mA	4.19 MHz crystal resonator (C1 = C2 = 22 pF). (Notes 3 and 5).
	I_{DD2}		500	1500	μA	4.19 MHz crystal resonator (C1 = C2 = 22 pF). HALT mode.
	I_{DD3}		350	1000	μA	32 kHz ceramic resonator. (Note 4).
			35	100	μA	32 kHz ceramic resonator. HALT mode.
	I_{DD4}		0.5	20	μA	XT1 = 0 V. STOP mode.

Notes:

(1) Voltage deviation is the difference between the ideal segment or common output value VLCDn (n = 0, 1, or 2) and the output voltage.

(2) Current value for the internal pull-up resistor is not included.

(3) When the processor clock control register (PCC) is set to 0011H and operated in the high-speed mode.

(4) When operated with the subsystem clock and the system clock control register (SCC) is set to 1001H with the main system clock oscillator stopped.

(5) When the subsystem clock is the oscillator.

AC Characteristics, μPD75P328

$T_A = -10$ to +70°C; $V_{DD} = 5$ V ±5%

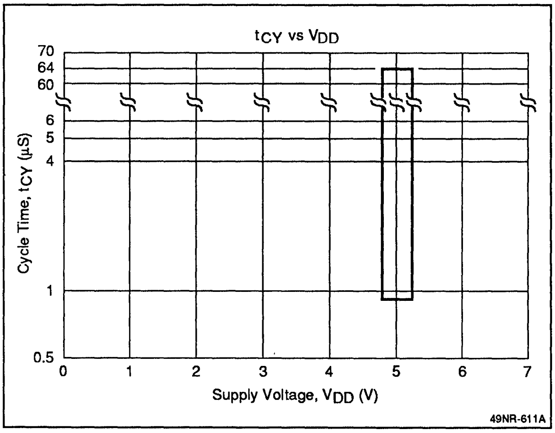
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Cycle time (Note 1)	t_{CY}	0.95		64	μs	Operation with main system clock
		114	122	125	μs	Operation with subsystem clock
TI0 input frequency	f_{TI}	0		1	MHz	
TI0 input low- and high-level width	t_{IL}, t_{IH}	0.48			μs	
Interrupt inputs low- and high-level width	t_{INTL}, t_{INTH}	(Note 2)			μs	INT0
		10			μs	KR_0-KR_7. INT1, 2, and 4
$\overline{\text{RESET}}$ low level width	t_{RSL}	10			μs	

Notes:

(1) Cycle time (minimum instruction execution time) is determined by the frequency of the oscillator connected to the microcomputer, system clock control register (SCC), and the processor clock control (PCC). See figure 6.

(2) $2t_{CY}$ or $128/f_{XX}$, depending on the setting of the interrupt mode register (IM0).

Figure 6. μPD75P328 t_{CY} vs V_{DD} with Main System Clock



A/D Converter Characteristics, μPD75P328

T_A = −10 to +70°C; V_{DD} = 5 V ±5%; AV_{SS} = V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8	8	8	bits	
Absolute accuracy (Note 1)				±1.5	LSB	2.5 V ≤ AV_{REF} ≤ V_{DD}
Conversion time	t_{CONV}			$168/f_X$	s	(Note 2)
Sampling time	t_{SAMP}			$44/f_X$	s	(Note 3)
Analog input voltage	V_{IAN}	AV_{SS}		AV_{REF}	V	
Analog input impedance	R_{AN}		1000		MΩ	
AV_{REF} current	I_{REF}		1.0	2.0	mA	

Notes:

(1) The absolute accuracy does not include the quantization error (±1/2 LSB).

(2) The total conversion time until EOC = 1 is 40.1 μs at f_{XX} = 4.19 MHz.

(3) The time until completion of sampling is 10.5 μs (f_{XX} = 4.19 MHz). Note that the sampling time value is included in the total conversion time value.

(4) For detailed A/D converter information refer to the user's manual.

Serial Transfer Operation, μPD75P328
2-Line/3-Line Serial I/O Mode ($\overline{SCK}$, Internal Clock Output)

T_A = −10 to +70°C; V_{DD} = 5 V to ±5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{SCK}$ cycle time	t_{KCY1}	1600			ns	Output
$\overline{SCK}$ low- and high-level width	t_{KH1}, t_{KL1}	$0.5t_{KCY1}$ −50			ns	Output
SI vs $\overline{SCK}$ ↑ setup time	t_{SIK1}	150			ns	
SI vs $\overline{SCK}$ ↑ hold time	t_{KSI1}	400			ns	
SO vs $\overline{SCK}$ ↓ output delay time	t_{KSO1}			250	ns	

Serial Transfer Operation, µPD75P328
2-Line/3-Line Serial I/O Mode ($\overline{\text{SCK}}$, External Clock Input)

T_A = −10 to +70°C; V_{DD} = 5 V to ±5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY2}	800			ns	Input
$\overline{\text{SCK}}$ low- and high-level width	t_{KH2}, t_{KL2}	400			ns	Input
SI vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK2}	100			ns	
SI vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI2}	400			ns	
SO vs $\overline{\text{SCK}}$ ↓ output delay time	t_{KSO2}			300	ns	

SBI Mode, µPD75P328
$\overline{\text{SCK}}$, Internal Clock Output (Master)

T_A = −10 to +70°C; V_{DD} = 5 V to ±5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY3}	1600			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KH3}, t_{KL3}	$0.5t_{KCY3}$−50			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK3}	150			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI3}	$0.5t_{KCY3}$			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↓ output delay time	t_{KSO3}	0		250	ns	
$\overline{\text{SCK}}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY3}			ns	
SB0, SB1 ↓ to $\overline{\text{SCK}}$ ↓	t_{SBK}	t_{KCY3}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY3}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY3}			ns	

SBI Mode, µPD75P328
$\overline{\text{SCK}}$, External Clock Input (Slave)

T_A = −10 to +70°C; V_{DD} = 5 V to ±5%

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{KCY4}	800			ns	
$\overline{\text{SCK}}$ low- and high-level width	t_{KH4}, t_{KL4}	400			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ setup time	t_{SIK4}	100			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↑ hold time	t_{KSI4}	$0.5t_{KCY4}$			ns	
SB0, SB1 vs $\overline{\text{SCK}}$ ↓ output delay time	t_{KSO4}	0		300	ns	
$\overline{\text{SCK}}$ ↑ to SB0, SB1 ↓	t_{KSB}	t_{KCY4}			ns	
SB0, SB1 ↓ to $\overline{\text{SCK}}$ ↓	t_{SBK}	t_{KCY4}			ns	
SB0, SB1 low-level width	t_{SBL}	t_{KCY4}			ns	
SB0, SB1 high-level width	t_{SBH}	t_{KCY4}			ns	

Data Memory STOP Mode Low Voltage Data Retention Characteristics, μPD75P328

T_A = −10 to +70°C

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention voltage	V_{DDDR}	2.0		6.0	V	
Data retention current (Note 1)	I_{DDDR}		0.1	10	μA	V_{DDDR} = 2.0 V
Release signal setup time	t_{SREL}	0			μs	
Oscillation stabilization time (Note 2)	t_{WAIT}		(Notes 3, 4)		ms	Release by $\overline{\text{RESET}}$ input
			(Note 3)		ms	Release by interrupt request

Notes:

(1) Internal pull-up resistor current is not included in the table for this item.

(2) Oscillation stabilization WAIT time is the time during which the CPU is stopped and the crystal is stabilizing. This time is required to prevent unstable operation while the oscillator is stabilizing. The interval timer can be used to delay the CPU from executing instructions by using the basic interval timer mode register (BTM) according to the following table:

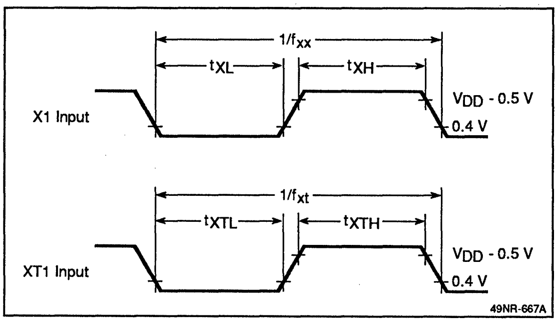
BTM3	BTM2	BTM1	BTM0	WAIT time (f_{XX} = 4.19 MHz)
–	0	0	0	$2^{20}/f_{XX}$ (approximately 250 ms)
–	0	1	1	$2^{17}/f_{XX}$ (approximately 31.3 ms)
–	1	0	1	$2^{15}/f_{XX}$ (approximately 7.82 ms)
–	1	1	1	$2^{13}/f_{XX}$ (approximately 1.95 ms)

(3) Consult the manufacturer's resonator specification sheet for this value.

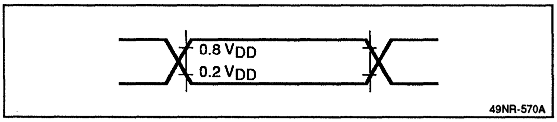
(4) The interval timer will cause a delay of $2^{17}/f_{XX}$ ms after a reset.

Timing Waveforms

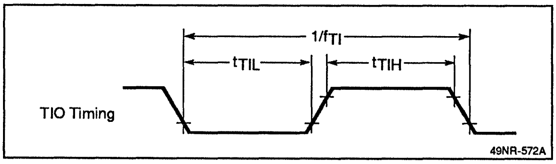
Clock Timing



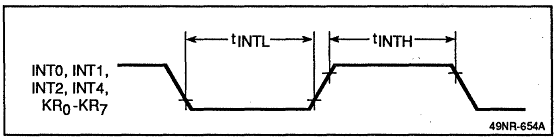
AC Timing Measurement Points (except X1 and XT1)



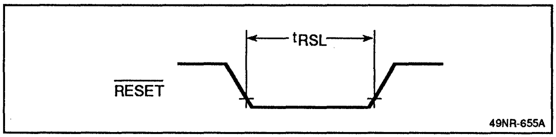
TI0 Timing



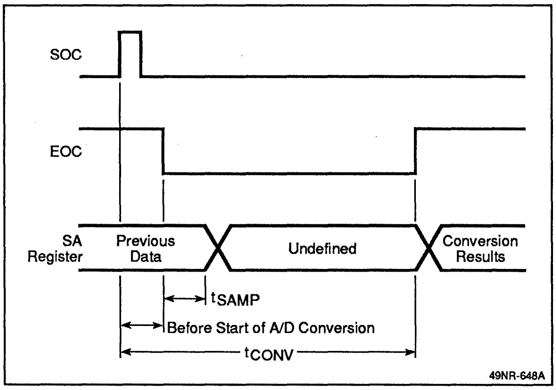
Interrupt Input Timing



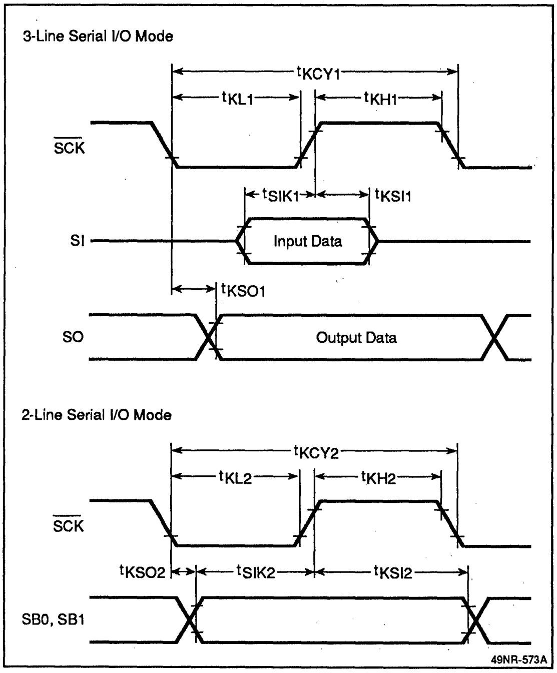
RESET Input Timing



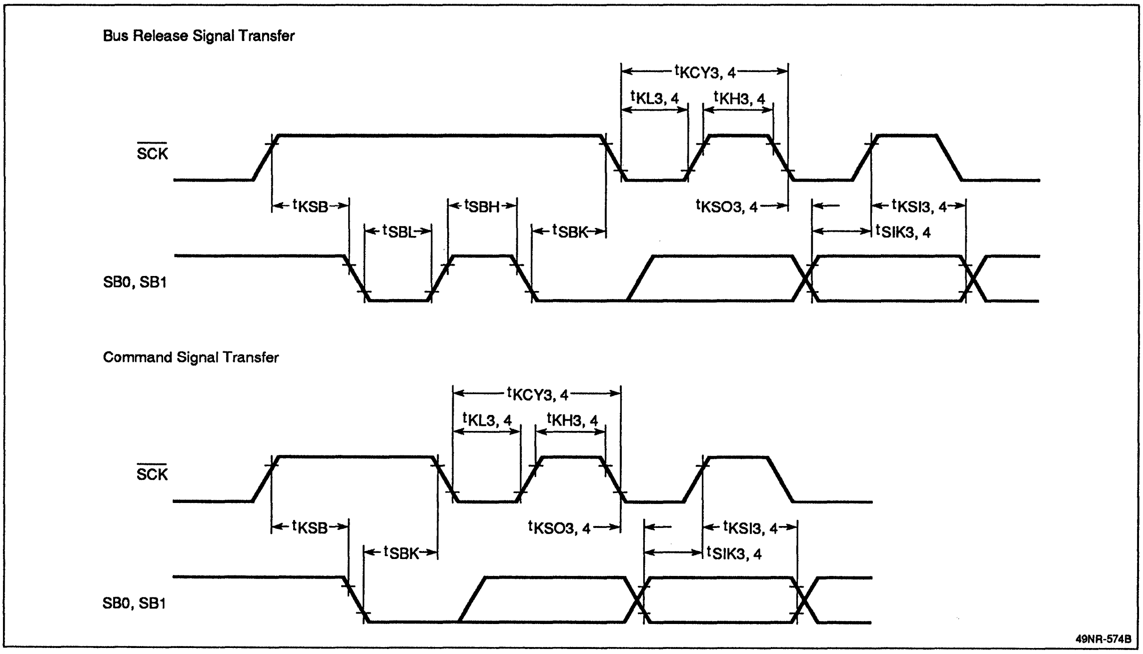
A/D Conversion Timing



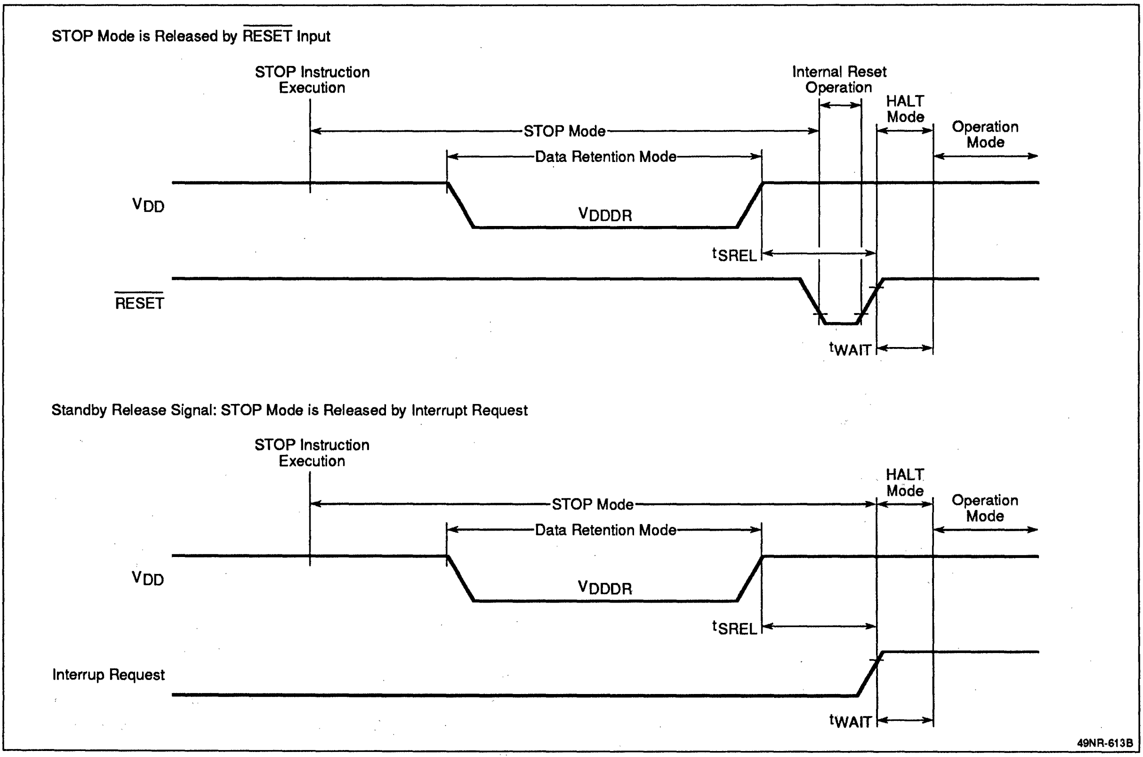
Serial Transfer Timing: 2-Line/3-Line (Serial I/O Mode)



Serial Transfer Timing (SBI Mode)



Data Retention Timing



FUNCTIONAL DESCRIPTION

Addressing and Memory Mapping

The architecture of the μPD75328 provides separate addressing spaces for program memory (ROM) and data memory (RAM).

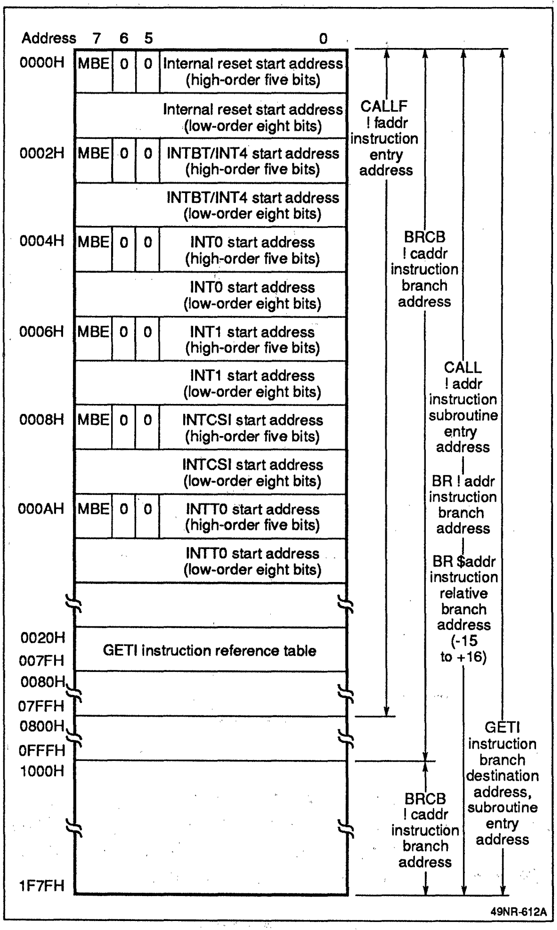
Program Memory (ROM). The program memory (ROM) is addressed by a 13-bit program counter(PC). The ROM contains program object code, interrupt vector table, GETI instruction reference table, and table data. Table data is obtained by using the table reference instruction MOVT.

Figure 7 shows the address range for a branch or subroutine call instruction. The BR PCDE and BR PCXA instructions are also used for branching, where only the low order eight-bits of the PC are changed. Program memory addresses range from 0000H to 1FFFH. All locations of the ROM except 0000H and 0001H can be used as program memory. However, if interrupts or GETI instructions are used, their corresponding locations cannot be used for program memory. Table 3 shows the ROM reserved special purpose addresses.

Table 3. ROM Reserved Addresses

Addresses	Description
0000H - 0001H	Vector addresses used for RESET. They also contain the MBE bit.
0002H - 000BH	Interrupt vector address area. Each contains an MBE bit value. Interrupts can start from any ROM location except 0000H-0001H.
0020H - 007FH	Table area for GETI instructions. GETI accesses 2 byte or 3 byte instructions using one byte of program memory. This is useful for compacting code.

Figure 7. Program Memory Map


Address 7 6 5 0
0000H
MBE 0 0 Internal reset start address (high-order five bits)
Internal reset start address (low-order eight bits)
0002H
MBE 0 0 INTBT/INT4 start address (high-order five bits)
INTBT/INT4 start address (low-order eight bits)
0004H
MBE 0 0 INT0 start address (high-order five bits)
INT0 start address (low-order eight bits)
0006H
MBE 0 0 INT1 start address (high-order five bits)
INT1 start address (low-order eight bits)
0008H
MBE 0 0 INTCSI start address (high-order five bits)
INTCSI start address (low-order eight bits)
000AH
MBE 0 0 INTT0 start address (high-order five bits)
INTT0 start address (low-order eight bits)
0020H
007FH
GETI instruction reference table
0080H
07FFH
0800H
0FFFH
1000H
1F7FH
CALLF ! faddr instruction entry address
BRCB ! caddr instruction branch address
CALL ! addr instruction subroutine entry address
BR ! addr instruction branch address
BR $addr instruction relative branch address (-15 to +16)
GETI instruction branch destination address, subroutine entry address
BRCB ! caddr instruction branch address
49NR-612A

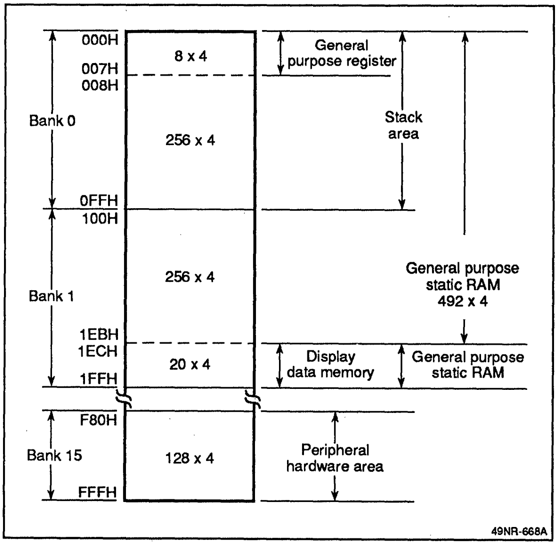
Program Counter (PC). The program counter (PC) is a 13-bit binary counter that holds the address of the current program memory location. When an instruction executes, the PC is automatically incremented by the number of bytes in the current instruction. When a branch instruction (BR, BRCB) executes, the PC bits are loaded with the branch address from a register pair or an instruction's immediate data. When a subroutine call instruction (CALL, CALLF) is executed, or an interrupt is generated, the PC is incremented to point to the next instruction. This address is saved in the stack memory. The CALL instruction or interrupt address is then loaded into the PC. When a return instruction (RET, RETS, or RETI) is executed, the contents of the stack memory is restored to the PC.

Data Memory (RAM). Figure 8 shows the memory bank for the data memory (RAM). Data memory contains three banks, banks 0, 1, and 15, and consists of a general purpose static RAM, general purpose registers, and peripheral control registers. Memory is accessed by the memory bank enable (MBE) bit and by programming the memory bank select (MBS) register. If MBE = 0, the lower 128 nibbles of memory bank 0 and the upper 128 nibbles of bank 15 are accessed. If MBE = 1, the value in the MBS register specifies the memory bank. The values can be 0H for memory bank 0, 1H for memory bank 1, and FH for memory bank 15. Both memory bank 0 and bank 1 contain 256 nibbles. Although the memory is organized in nibbles, the 75X architecture and instruction set allow data operation in bytes, nibbles, and individual bits.

Data memory is used for storing processed data, general purpose registers, and as a stack in a subroutine or interrupt service. The last 20 nibbles of bank 1 are used to store the LCD display data. If the area is not completely used by the LCD, it can be used as general purpose RAM. The RAM, because of its static nature, can retain its data when CPU operation is stopped and the chip is in the standby mode, provided it has a minimum of 2 volts applied to it.

RAM bank 0 has eight 4-bit general purpose registers starting at location 00H. These registers also can be used as four 8-bit registers. The on-chip peripheral control registers and ports reside in the upper 128 nibbles of bank 15. Addresses not assigned to a register are not available as random memory in bank 15. The lower 128 nibbles of bank 15 do not contain RAM.

Figure 8. Data Memory Map



Addressing Modes. The µPD75328 can address data memory and ports as individual bits, nibbles, or bytes. These addressing modes are as follows:

- 1-bit direct data memory
- 4-bit register indirect (@rpa)
- 4-bit direct data memory
- 8-bit register indirect (@HL)
- 8-bit direct data memory

Table 4 shows the data memory addressing modes.

Table 4. Data Memory Addressing Modes (Note 1)

Addressing Mode	Format	Address

1-bit direct addressing	mem. bit	The memory bank is:
		if MBE = 0:
		MB = 0 for addr 00H-7FH
		MB = 15 for addr 80H-FFH
		if MBE = 1:
		MB = (MBE)●(MBS Reg)
		The memory location and bit within the memory bank is:
		mem.bit

4-bit direct addressing	mem	The memory bank is:
		if MBE = 0:
		MB = 0 for addr 00H-7FH
		MB = 15 for addr 80H-FFH
		if MBE = 1:
		MB = (MBE)●(MBS Reg)
		The memory location within the memory bank is:
		mem

Table 4. Data Memory Addressing Modes (Note 1) (cont)

Addressing Mode	Format	Address
8-bit direct addressing	mem	The memory bank is if MBE = 0: MB = 0 for addr 00H-7FH 15 for addr 80H-FFH Reg) thin the memory bank is: mem mem must be an even address
4-bit register indirect addressing	@ HL	The memory bank is: MB = (MBE)●(MBS Reg) The memory location within the memory bank is: contained in register HL
	@ DE	The memory bank is always bank 0. The memory location within the memory bank is: contained in register DE
	@ DL	The memory bank is always bank 0. The memory location within the memory bank is: contained in register DL
8-bit register indirect addressing	@ HL	The memory bank is: MB = (MBE)●(MBS Reg) The memory location within the memory bank is: contained in register HL HL must contain an even address
Bit manipulation addressing	fmem. bit	The memory bank is bank 15. The memory location in bank 15 is fmem where: fmem = B0H-BFH for interrupts fmem = F0H-FFH for I/O ports The bit is specified in: fmem.bit

Table 4. Data Memory Addressing Modes (Note 1) (cont)

Addressing Mode	Format	Address
Bit manipulation addressing (cont)	pmem.@L	The memory location is independent of MBE and MBS. The upper 10 bits of the location are in the high order 10-bits of pmem and the 2 lower address bits are in the upper 2-bits of register L. The bit to be manipulated is specified by the 2 LSBs of register L.
	@H + mem. bit	The memory bank is: MB = (MBE)•(MBS Reg) The memory location within the memory bank is: 4 upper bits are in register H 4 lower bits are mem The bit is specified in: mem.bit
Stack addressing		The stack is always in bank 0 and the address is indicated by stack pointer SP

Notes:

(1) MBE: Memory bank enable bit
MB: Memory bank
MBS: Memory bank select register
mem: A memory location within a memory bank
mem.bit: A memory location and a bit at that location

Table 5 shows the peripheral control register addressing modes.

Table 5. On-Chip Peripherals Addressing Modes

Type of Manipulation	Addressing Mode	Hardware
Bit	MBE = 0 or MBE = 1 and MBS = 15; direct addressing (specified in mem.bit).	All hardware where bit manipulation can be performed
	Direct addressing regardless of the setting of MBE and MBS. (specified in fmem. bit).	ISTO, MBE; IExxx, IRQxxx, PORTn.x
	Indirect addressing regardless of the setting of MBE and MBS. (specified in pmem. @L).	BSBn.x; PORTn.x
4-bit	MBE = 0 or MBE = 1 and MBS = 15; direct addressing. (specified in mem).	All hardware where 4-bit manipulation can be performed
	MBE = 1 and MBS = 15; register indirect addressing. (specified in @HL)	

Table 5. On-Chip Peripherals Addressing Modes (cont)

Type of Manipulation	Addressing Mode	Hardware
8-bit	MBE = 0 or MBE = 1, and MBS = 15; direct addressing (specified in mem.); mem must be an even address.	All hardware where 8-bit manipulation can be performed
	MBE = 1 and MBS = 15; register indirect addressing (specified in @HL); L register must contain an even number.	

Clock Generator

The clock generator uses a crystal as a time base to generate its clocks. Figure 9 shows the generator which consists of a main and subsystem oscillator, frequency dividers, multiplexers, and three control registers (PCC, SCC, and CLOM). Registers PCC and SCC are programmed to supply frequencies derived from the crystal to the CPU at one of four speeds. Register CLOM controls the clock output to the output pin PCL. Registers PCC and SCC control the HALT and STOP logic.

The μPD75328 contains a subsystem clock with an oscillator driven by an external crystal. The clock operates from 32 kHz to 35 kHz. It can be used as a clock source for the watch timer, the LCD controller/driver, and the CPU.

Basic Interval Timer

The basic interval timer provides continuous real time interrupts. The timer consists of a multiplexer, 8-bit free running counter, and the 4-bit BTM control register. See figure 10. Every time the counter increments to FFH, it generates an interrupt, overflows to 00H, and continues to count. In addition to clearing the counter and its interrupt request, the BTM register is used to select one of four clock inputs. The counter can generate 250 ms interrupts with a 4.19 MHz crystal. It provides oscillator stabilization time when the chip leaves the STOP mode.

Timer/Event Counter

The timer/event counter consists of a binary 8-bit up-counter, 8-bit modulo register, 8-bit comparator, clock multiplexer, mode control register (TM0), and a TOUT flip-flop. See figure 11. Control logic allows the TOUT flip-flop signal to be output to port 2.

The counter operates when an 8-bit value is loaded into the modulo register. A count register clock is selected in the clock multiplexer by control register TM0. The 8-bit up-counter is incremented every time it receives a counter pulse (CP). When the count value equals the modulo register count, the 8-bit comparator outputs a signal.This toggles the TOUT flip-flop and resets the count register to 00H. The count register continues to count up unless it is stopped. Every time flip-flop TOUT changes state, it generates an interrupt signal. Signal TOUT can also be used as a clock for the serial interface.

Figure 9. Clock Generator

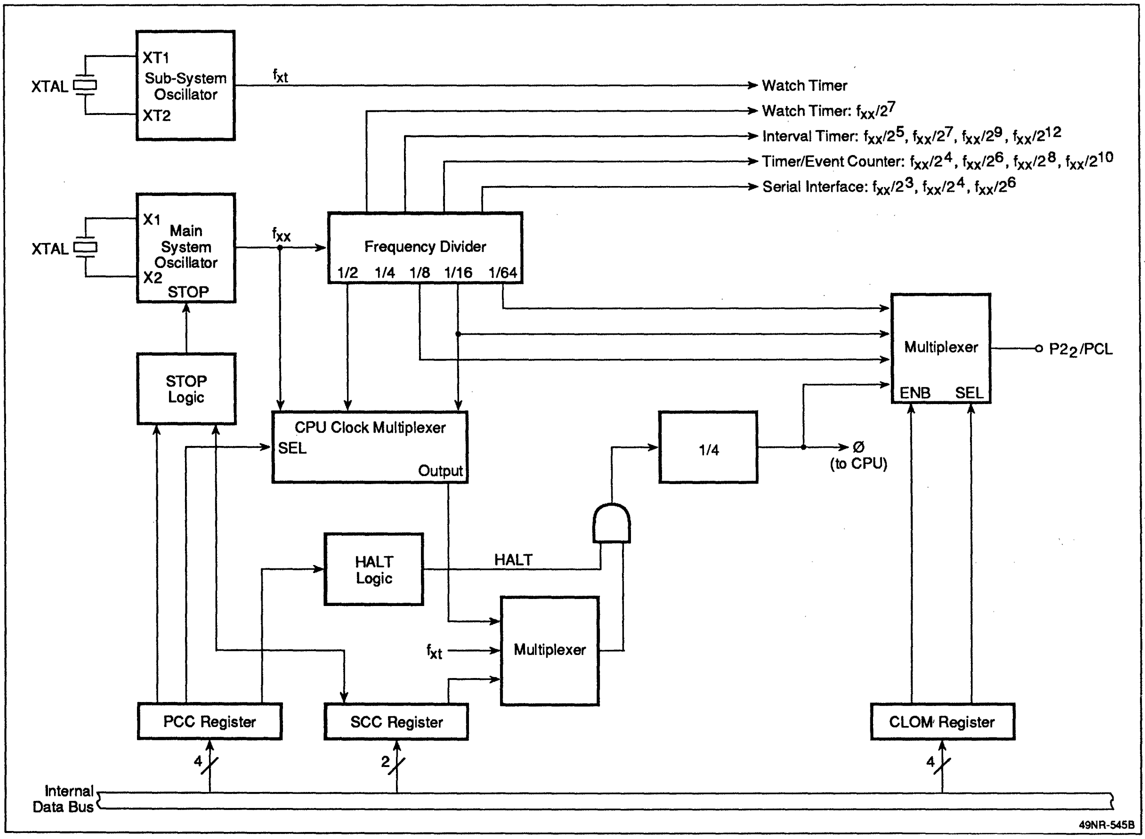
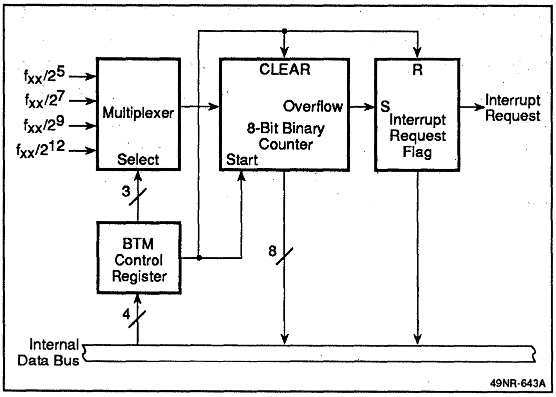


Figure 10. Basic Interval Timer



Watch Timer

The watch timer, when using a 4.19 MHz crystal, will generate interrupt requests (not interrupts) at 0.5 second intervals. The timer consists of an input multiplexer, divider, output multiplexer, control logic, and control register WM. See figure 12. It is normally used as a time source for tracking the time of day. It is also used as a clock source for the LCD controller. It can operate in the STOP mode, when a subsystem clock is present, and is capable of outputting a 2 kHz buzzer signal.

Figure 11. Timer/Event Counter

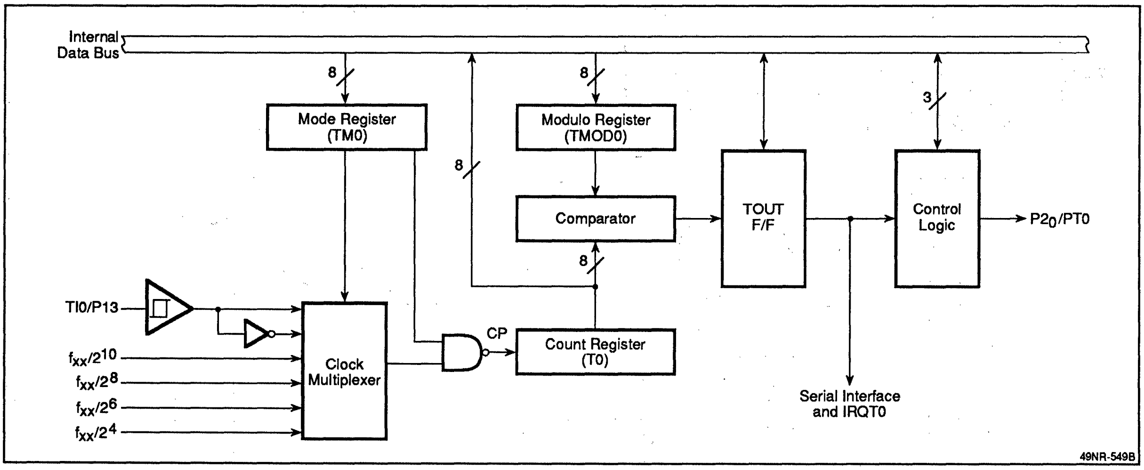
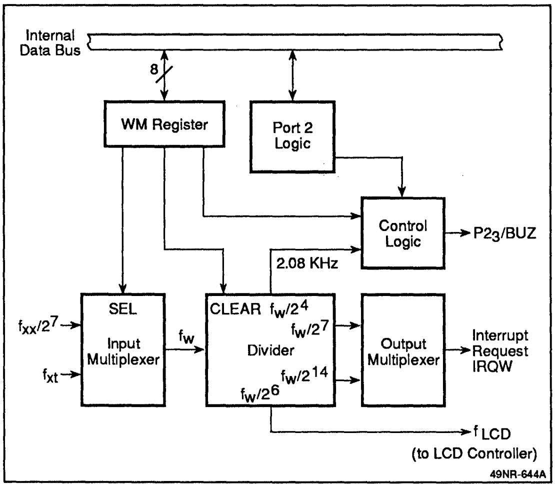


Figure 12. Watch Timer



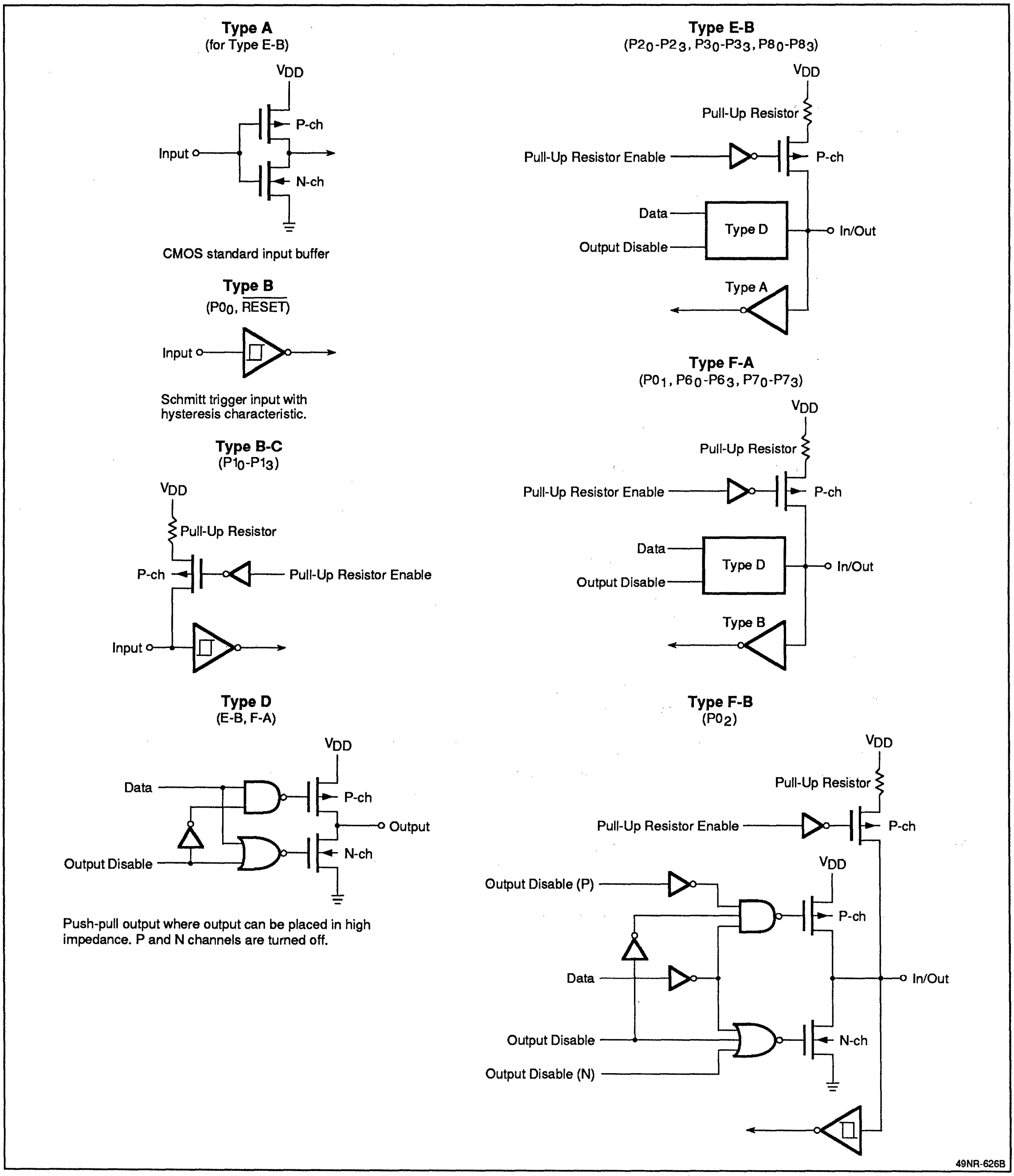
Input/Output Ports

The μPD75328 provides eight 1-bit output ports and nine 4-bit ports; seven are input/output and two are input only. Table 6 lists the function and operation of the I/O ports. Figure 13 shows the internal circuits of the ports, which are classified as types A through Z.

Table 6. Operation of the Digital I/O Ports

Port	Input/Output	Operation	Additional Pin Applications
P0, P1	4-bit input only	Can be read or tested regardless of the operation mode of the following pins: SO/SB0, SI/SB1, $\overline{\text{SCK}}$, INT0, INT1, INT2, INT4, or TI0.	SO/SB0, SI/SB1, $\overline{\text{SCK}}$, INT0, INT1, INT2, INT4, and TI0
P3	4-bit I/O	Can be set-up in input or output mode in 1-bit units.	LCDCL and SYNC
P6	4-bit I/O		KR_0-KR_3
P2	4-bit I/O	Can be set-up in input/output mode in 4-bit units. Ports 6 and 7 can be paired for data input/output in 8-bit units.	PTO0, PCL, and BUZ
P7	4-bit I/O		KR_4-KR_7
P8	4-bit I/O		
P4, P5	4-bit I/O (N-channel open drain 10 volts)	Can be set-up in input or output mode in 4-bit units. Ports 4 and 5 can be paired for data input/output in 8-bit units. An LED can be driven directly.	Internal pull-up resistor specified in 1-bit units by the mask option
BP_0-BP_7	1-bit output only	Data is output in 1-bit units. The BP_0-BP_7 pins are also used as output pins (S_{24}-S_{31}) to drive LCD segments. BP_0-BP_7 can be changed by software.	

Figure 13. Input/Output Circuits

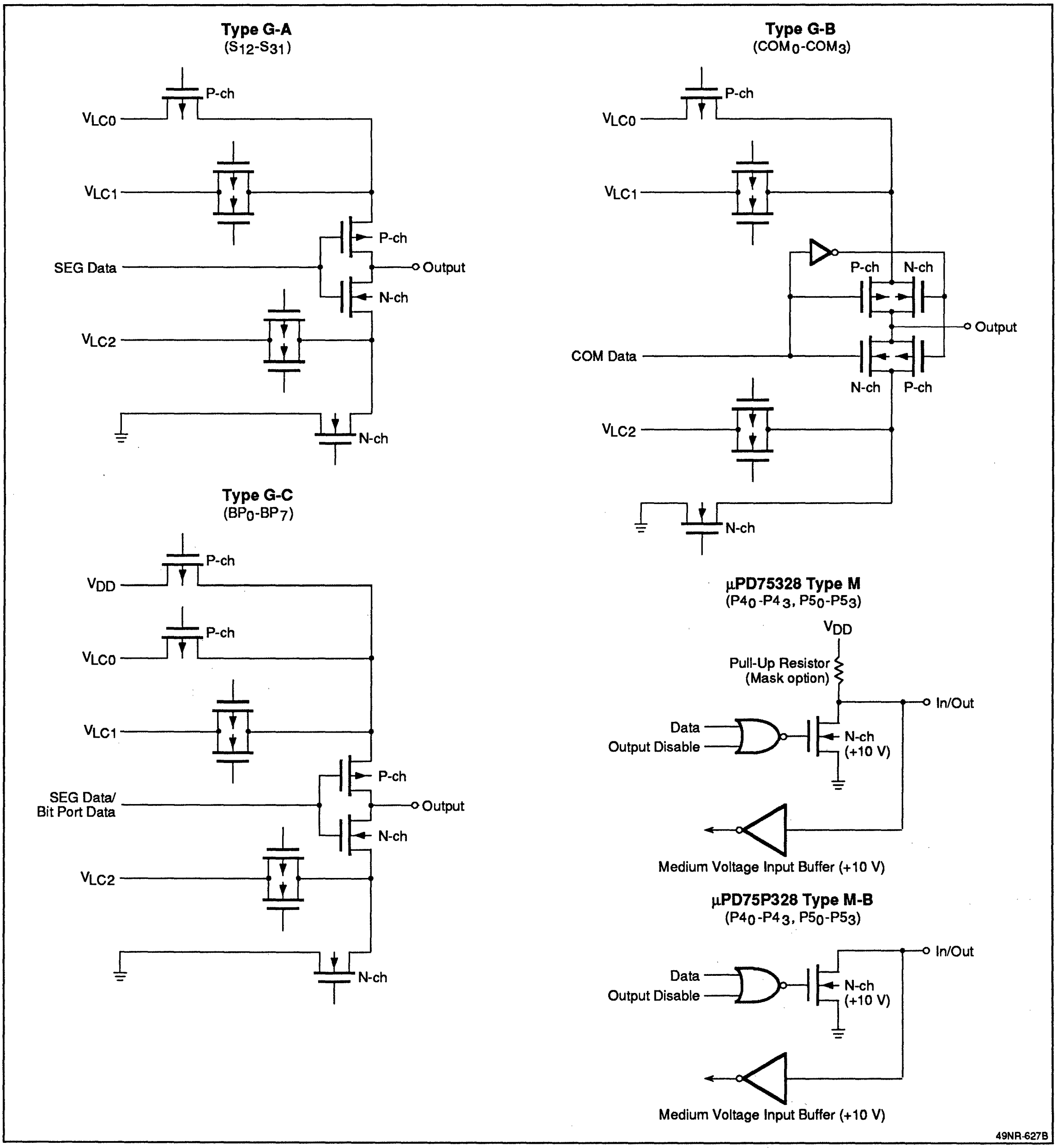


CMOS standard input buffer

Schmitt trigger input with hysteresis characteristic.

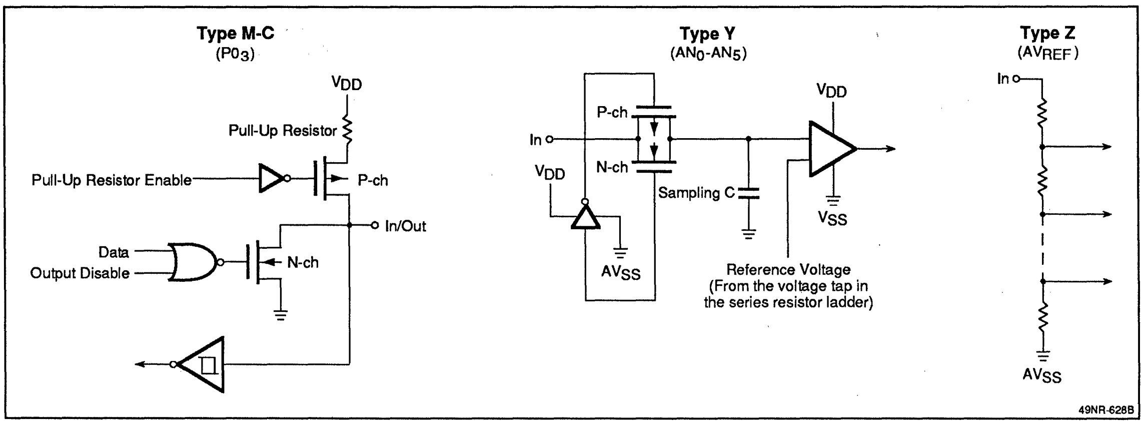
Push-pull output where output can be placed in high impedance. P and N channels are turned off.

Figure 13. Input/Output Circuits (cont)



4

Figure 13. Input/Output Circuits (cont)



Serial Interface

The 8-bit serial interface allows the μPD75328 to communicate with other NEC or NEC like serial interfaces. The serial interface consists of an 8-bit shift register (SIO), serial output latch (SO), 8-bit address comparator, slave address register (SVA), control registers (CSIM and SBIC), busy/acknowledge circuitry, and bus release/detect circuitry. See figure 14. The interface also contains a serial clock counter, clock multiplexer, and serial clock control logic. The serial interface contains a three wire interface, which consists of the following:

- Serial Data In (SI/SB1)
- Serial Data Out (SO/SB0)
- Serial Shift Clock ($\overline{SCK}$)

The three serial interface operation modes are:
- Two-wire serial mode
- Three-wire serial mode
- Two-wire SBI mode

The two or three wire serial modes are the simplest modes; the 8-bit shift register is loaded with a byte of data and eight clock pulses are generated. The pulses shift data out of the SO line and in from the SI line, thereby communicating in full duplex. When a byte of data is sent, a burst of eight clock pulses is generated and 8-bits of data are sent. The data may be sent with the LSB or MSB first. The interface can also be set to receive data only, consequently SO will be in the high impedance state. One of four internal clocks or an external clock clocks the data.

The SBI mode uses a two-wire interface with devices in a master/slave configuration. See figure 15. There is only one master device at a time; all others are slaves. The master sends addresses, commands, and data over the bus. The slaves are able to detect in hardware if their addresses were sent, a command was sent, or a portion of data were sent. There can be up to 256 slave addresses, 256 commands, and 256 data types. All commands are user definable. Commands can be sent to change slaves into masters; previous masters become slaves. Firmware performs this type of operation and thus the user decides whether the bus is simple or complex.

Figure 14. Serial Interface Block Diagram

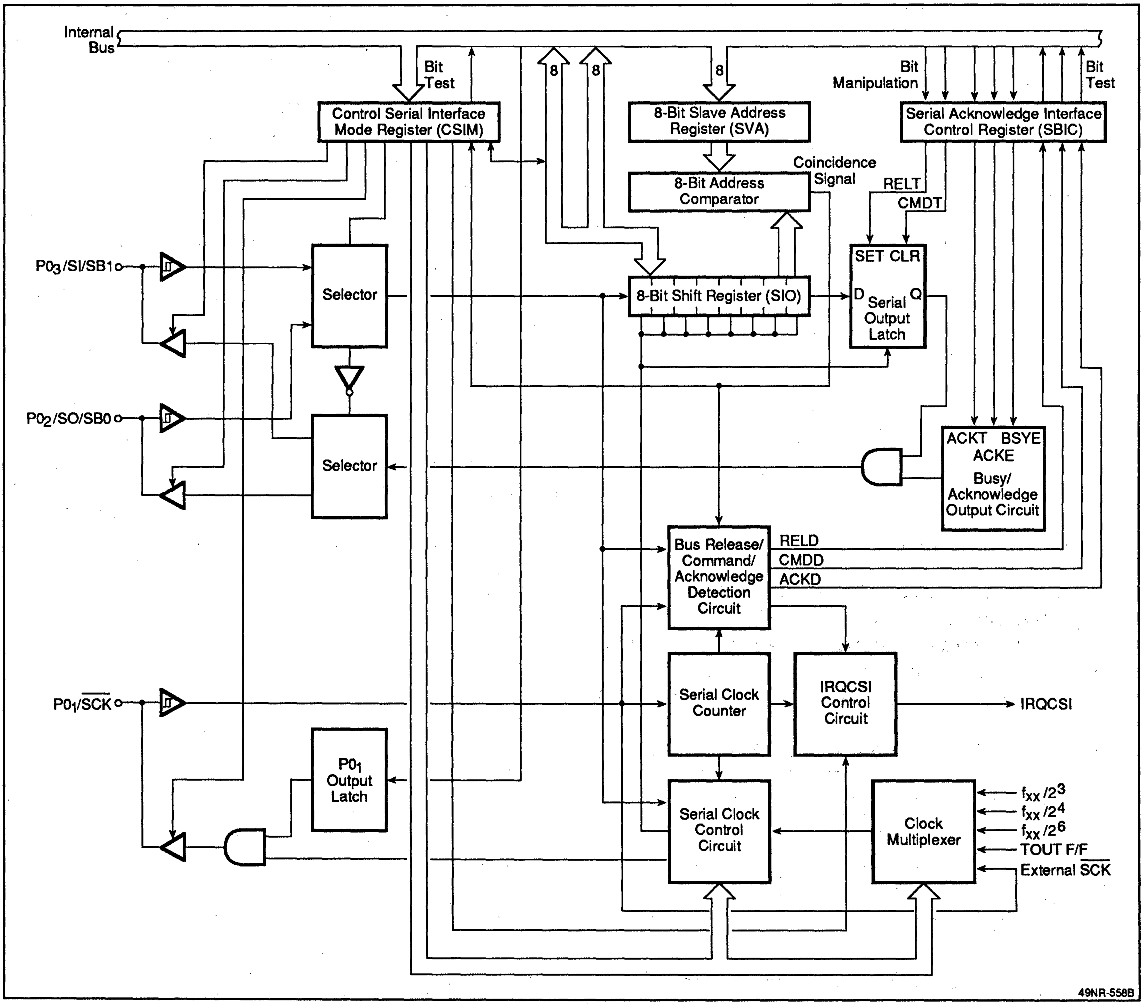
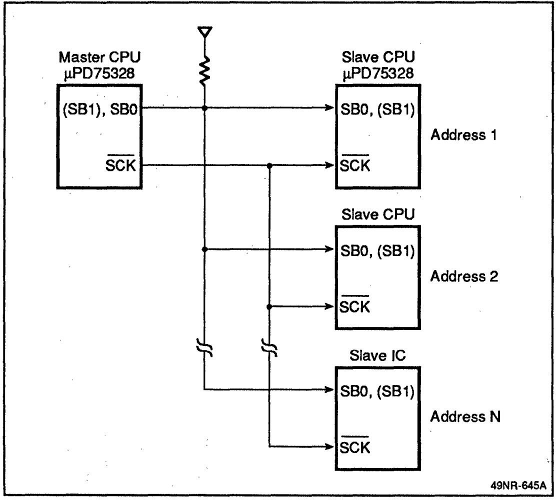


Figure 15. SBI Mode Master/Slave Configuration



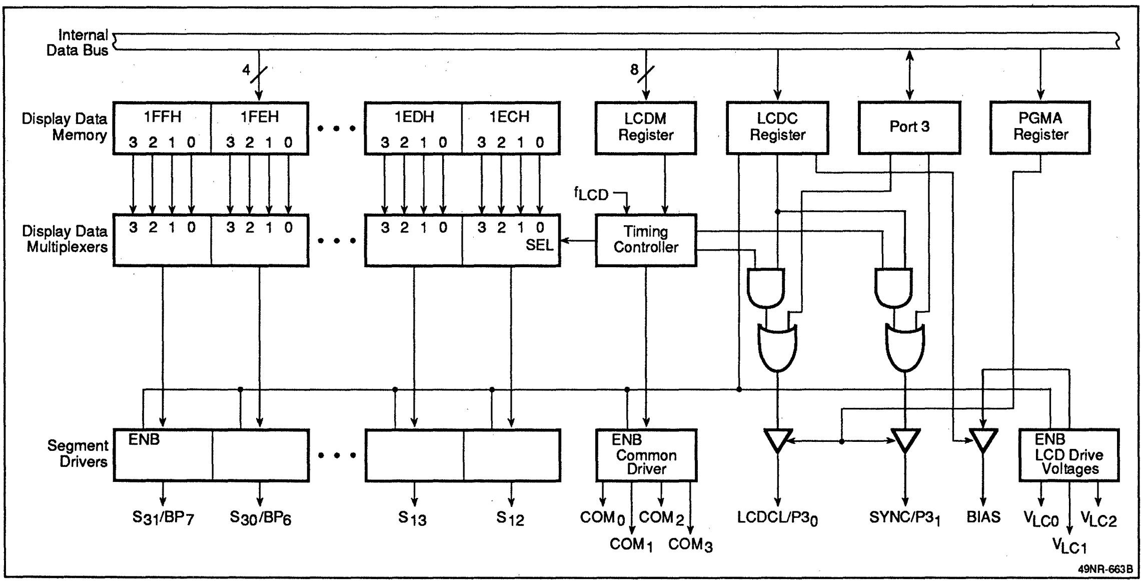
LCD Controller/Driver

The liquid-crystal display (LCD) controller/driver can directly drive up to a maximum of 80 segments. See figure 16. The controller can be programmed to operate in the static mode (drive 20 segments), multiplexed mode (drive 40 segments), triplexed mode (drive 60 segments), or the quadriplexed mode (drive 80 segments). The multiplexed mode uses 1/2 BIAS voltage; triplexed mode uses 1/2 or 1/3 BIAS voltage; and the quadriplexed mode uses 1/3 BIAS voltage.

The controller/driver automatically refreshes the LCD with data from the upper 20 nibbles of RAM memory bank one. To drive an LCD, the controller/driver uses display data multiplexers, segment drivers S_{12}-S_{31}, and common drivers COM_0-COM_3. The LCD controller/driver is controlled by registers LCDM, LCDC, and PGMA. The LCD controller/driver clock (F_{LCD}) is the main clock and is supplied by the watch timer. The watch timer operates while the chip is in the STOP mode, when it is driven from the subsystem clock. Hence the LCD controller/driver also operates in the STOP mode.

The SYNC signal and LCDCL clock are available as outputs so that additional LCD controllers can be added. The drive signals for the controller/driver can be set internally by the resistor ladder mask option (ordered as a mask option). However, the levels can also be set by using external resistors connected to pins V_{LC0}-V_{LC2}. To control the contrast of the LCD, a BIAS pin is also available.

Figure 16. LCD Controller/Driver Block Diagram



A/D Converter

The 8-bit analog to digital (A/D) converter is equipped with six inputs and uses an successive approximation routine (SAR) for the A/D conversion. See figure 17. An A/D conversion occurs when one of six inputs is selected by the ADM register. The conversion starts by setting bit 3 of the ADM register. The selected input is sampled by using the sample and hold circuit and multiplexer. Then, using the SAR with the comparator, resistor ladder, and SA register, the input value is converted. The converted value is stored in the SA register. When bit 2 of the ADM register is set, conversion is complete and can be read from the SA register.

Bit Sequential Buffer

The 16-bit sequential buffer is the only general purpose RAM in the upper half of data memory bank 15: all the other locations in this bank contain either on-chip peripheral control registers or unused addresses. A typical application for this buffer is data storage for the next serial output or input. Another application is as a port output data storage area. The bit sequential buffer can be bit, nibble, or byte manipulated.

Interrupts

The three external and three internal interrupts are all vectored interrupts and are shown in figure 18. Table 7 lists a summary of the interrupts. Input INT2 detects rising edge inputs and generates an interrupt request flag, which is testable. Inputs KR_0 through KR_7 detect a falling edge and generate the same interrupt request flag as INT2. INT2 and KR_0 through KR_7 do not cause an interrupt, but can be used to release the standby mode. Interrupt requests and all interrupts except INT0 release the standby mode.

Figure 17. A/D Converter Block Diagram

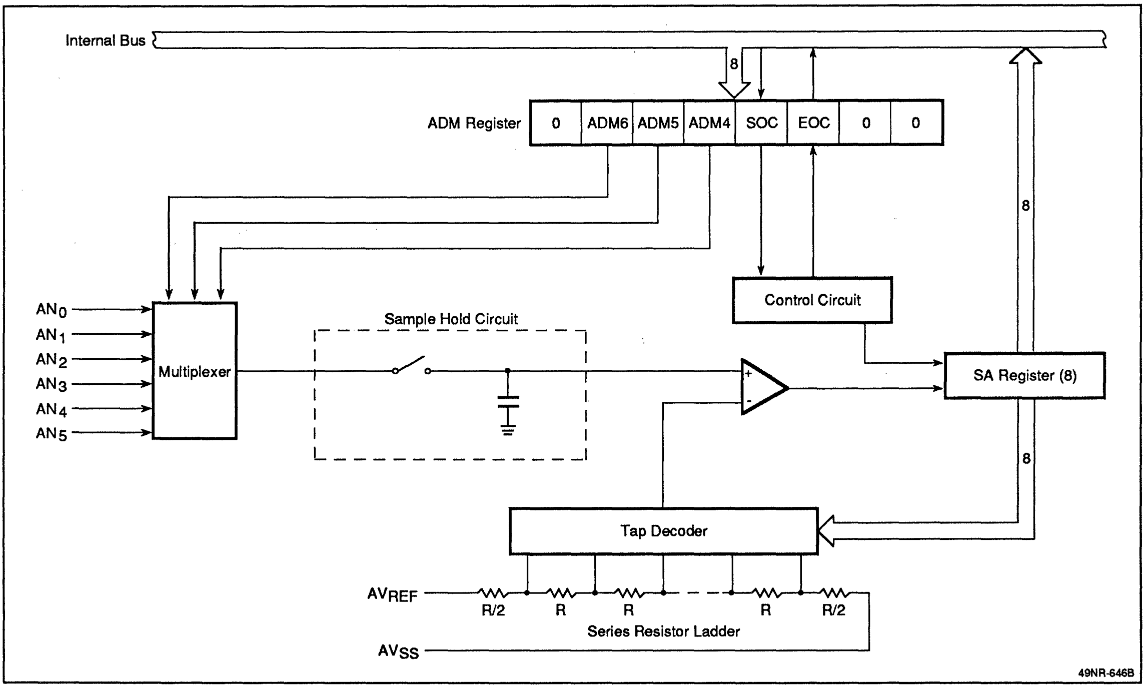


Figure 18. Interrupt Controller Block Diagram

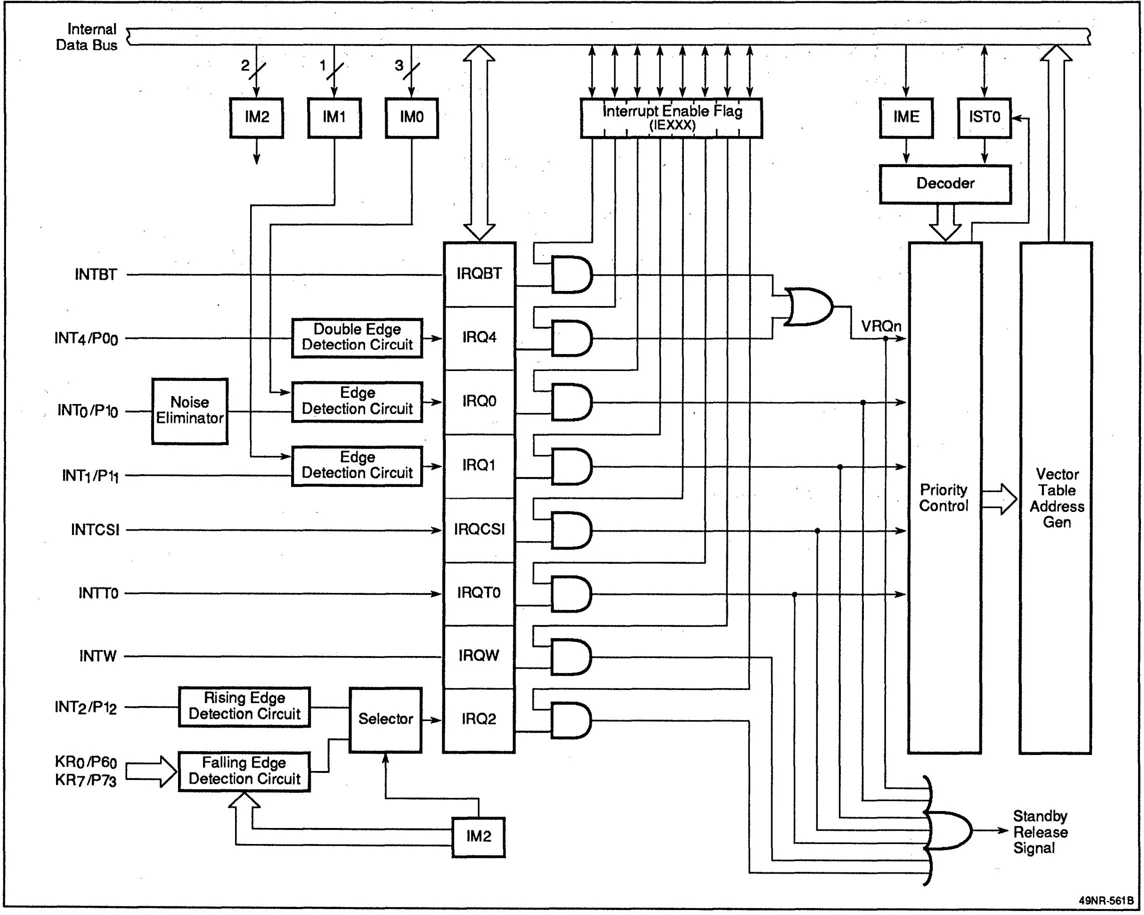


Table 7. Interrupt Sources

Interrupt Source	Internal/External	Interrupt Priority (Note 1)	Vectored Interrupt Request/ Table Address
INTBT (Time reference interval signal from the basic interval timer)	Internal	1	VRQ1/0002H
INT4 (Rising and falling edge detection)	External		
INT0 (Rising/falling edge detection)	External	2	VRQ2/0004H
INT1 (Rising/falling edge detection)	External	3	VRQ3/0006H
INTCSI (Serial data transfer end signal)	Internal	4	VRQ4/0008H
INTT0 (Signal generated when programmable timer/ counter count register and modulo register coincide)	Internal	5	VRQ5/000AH
INT2 (Rising edge input detection to INT2 pin or falling edge input detection to KR_0-KR_7)	Testable input signals (Tests if IRQ2 and IRQW are set)		
INTW (Watch timer signal)			

Notes:

(1) The interrupt priority determines the order when two or more simultaneous interrupts occur.

Standby Modes

Three standby modes, HALT, STOP, and data retention reduce power consumption during a program standby state. Table 8 summarizes the standby modes.

Execution of the HALT instruction selects the HALT mode. In the HALT mode, the CPU clock ϕ is turned off which stops the CPU. However, all other portions of the chip except interrupt INT0 are functional. Execution of the STOP instruction selects the STOP mode. In the STOP mode, the chip's main system oscillator is turned off, stopping all portions of the chip except those operating from the subsystem clock. If the subsystem clock is used, it remains on.

A RESET or any interrupt request except INT0 releases the HALT and STOP modes. The data retention mode can be selected after the STOP mode has been selected. In this mode, the supply voltage V_{DD} can be lowered to 2 volts, further reducing the power consumption. The contents of the RAM and registers are retained. The data retention mode is released by first raising the supply voltage V_{DD} to its operating level. The chip will now be in the STOP mode which may be released as described above.

Table 8. Operation of the Standby Modes

Operating State	STOP Mode	HALT mode
Mode setting instruction	STOP instruction	HALT instruction
Clock oscillator	Only the main system clock oscillator is stopped.	Only CPU clock ϕ is stopped. Main and subsystem oscillators continue to operate.
Basic interval timer	Operation stops	Operation continues (IRQBT is set at reference time intervals).
Serial interface	Operates only when external $\overline{\text{SCK}}$ input is selected for serial clock.	Operational.
Timer/event counter	Operates only when TIO pin input is selected for clock count.	Operational.
Watch timer	Operates when f_{XT} is selected for the clock count.	Operational.
LCD controller	Operates only when f_{XT} is selected for LCDCL.	Operational.
External interrupts	INT1, INT2, and INT4 are allowed to operate. Only INT0 cannot operate.	All operational except INT0
CPU	Operation stops.	Operation stops
Release signal	Enabled interrupt request signal (except INT0) with interrupt enable flag or $\overline{\text{RESET}}$ input.	Enabled interrupt request signal (except INT0) with interrupt enable flag or $\overline{\text{RESET}}$ input.

$\overline{\text{RESET}}$

Table 9 shows the status of the chip, after the $\overline{\text{RESET}}$ signal is applied.

Table 9. Chip Status after $\overline{\text{RESET}}$

Function	$\overline{\text{RESET}}$ during Standby Mode	$\overline{\text{RESET}}$ during Operational Mode
Program counter (PC)	Contents of the low-order five bits of program memory address 0000H are loaded into program counter PC12-PC8; contents of address 0001H are loaded into PC7-PC0.	
PSW - carry flag (CY)	Held	Unknown
PSW - skip flag (SK0-SK2)	0	0
PSW - interrupt status flag (IST0)	0	0
PSW - bank enable flag (MBE)	Bit 7 of program memory address 0000H sets state of MBE.	
Stack pointer (SP)	Unknown	Unknown
Data memory (RAM)	Held (note 1)	Unknown
General-purpose registers (X, A, H, L, D, E, B, C)	Held	Unknown
Bank selection register (MBS)	0	0
Basic interval timer - counter (BT)	Unknown	Unknown
Basic interval timer - mode register (BTM)	0	0
Timer/event counter - counter (T0)	0	0
Timer/event counter - modulo register (TMOD0)	FFH	FFH
Timer/event counter - mode register (TM0)	0	0
Timer/event counter - TOE0, TOUT F/F	0, 0	0, 0
Watch timer mode register (WM)	0	0
Serial interface - shift register (SIO)	Held	Unknown
Serial interface - operation mode register (CSIM)	0	0
Serial interface - SBI control register (SBIC)	0	0

Table 9. Chip Status after $\overline{\text{RESET}}$ (cont)

Function	$\overline{\text{RESET}}$ during Standby Mode	$\overline{\text{RESET}}$ during Operational Mode
Serial interface - slave address register (SVA)	Held	Unknown
Clock generator, clock output circuit -processor clock control register (PCC)	0	0
Clock generator, clock output circuit -system clock control register (SCC)	0	0
Clock generator, clock output circuit-clock output mode register (CLOM)	0	0
LCD controller - display mode register (LCDM)	0	0
LCD controller - display control register (LCDC)	0	0
A/D converter - mode register (ADM), EOC	04H (EOC = 1)	04H (EOC = 1)
A/D converter - SA register	7FH	7FH
Interrupt function - interrupt request flags (IRQXXX)	Reset to 0	Reset to 0
Interrupt function - interrupt enable flags (IEXXX)	0	0
Interrupt function - interrupt master enable flag (IME)	0	0
Interrupt function - INT0, INT1, and INT2 mode registers (IM0, IM1, and IM2)	0, 0, 0	0, 0, 0
Digital ports - output buffer	Off	Off
Digital ports - output latch	Cleared to zero	Cleared to zero
Digital ports - I/O mode registers (PMGA, PMGB, PMGC)	0	0
Digital ports - Pull-up resistor specification register (POGA, POGB)	0	0
Pin states - $P0_0$-$P0_3$, $P1_0$-$P1_3$, $P2_0$-$P2_3$, $P3_0$-$P3_3$, $P6_0$-$P6_3$, $P7_0$-$P7_3$, $P8_0$-$P8_3$	Input	Input
Pin states - $P4_0$-$P4_3$, $P5_0$-$P5_3$	Internal pull-up resistors (high level). Open drain (high impedance).	
Pin states - S_{12}-S_{23}, COM_0-COM_3	Unknown	Unknown
Pin states - BIAS	Internal resistor ladder (low level). External resistor ladder (high impedance).	
Bit sequential buffer (BSB0-BSB3)	Held	Unknown

Notes:

(1) Data in addresses 0F8H-0FDH of the data memory is undefined when the $\overline{\text{RESET}}$ signal is input.

OTP PROM (Program Memory Write and Verify)

The μPD75P328 contain 8064 x eight-bits of one-time programmable (OTP) program memory. The OTP is programmed by the pins listed in table 10. During OTP programming, addresses are incremented by applying clock pulses to the X1 input.

Table 10. OTP Access

Pin	Function
V_{PP}	OTP programming voltage pin (normally V_{DD})
X1	Address increment clock input during programming.
MD_0-MD_3	Mode selection during OTP programming
$P4_0$-$P4_3$	4-bit data I/O pins during OTP programming, low-order four bits
$P5_0$-$P5_3$	4-bit data I/O pins during OTP programming, high-order four bits
V_{DD}	Supply voltage pin: 5 V ±10% during normal operation; 6 V during OTP programming.

Notes:

(1) During OTP programming: Connect all unused pins (except XT2) to V_{SS} through a pull-down resistor. Do not connect the XT2 pin.

(2) The μPD75P328 has no erasure window. The program memory data cannot be erased with ultraviolet light.

OTP Operation Mode

The μPD75P328 operates in the program memory write/verify mode when +6 V is applied to V_{DD} and 12.5 V to V_{PP}. Mode pins MD_0-MD_3 select the operation modes shown in Table 11.

Table 11. OTP Operation Mode Selection
V_{PP} = +12.5 V; V_{DD} = +6 V

MD_0	MD_1	MD_2	MD_3	Operating Mode
H	L	H	L	Program memory address clear
L	H	H	H	Program memory write
L	L	H	H	Program memory verify
H	X	H	H	Program inhibit

Notes:

(1) X = L or H.

Program Memory Write/Verify. The program memory write/verify procedure follows (high speed write is enabled):

(1) Connect unused pins to V_{SS} through a pull-down resistor. Hold X1 low.

(2) Supply 5 V to V_{DD} and V_{PP}.

(3) Wait for 10 μs.

(4) Select the program memory address clear mode.

(5) Change the voltage on V_{DD} to 6 V and on V_{PP} to 12.5 V.

(6) Select the program inhibit mode.

(7) Write data in the 1 ms write mode.

(8) Select the program inhibit mode.

(9) Select the verify mode. If data is written correctly, proceed to step 10; if data is not written correctly, repeat steps 7-9.

(10) Perform one additional write.

(11) Select the program inhibit mode.

(12) Increment the program memory address by one by inputting four pulses to X1.

(13) Repeat steps 7-12 until the end address occurs.

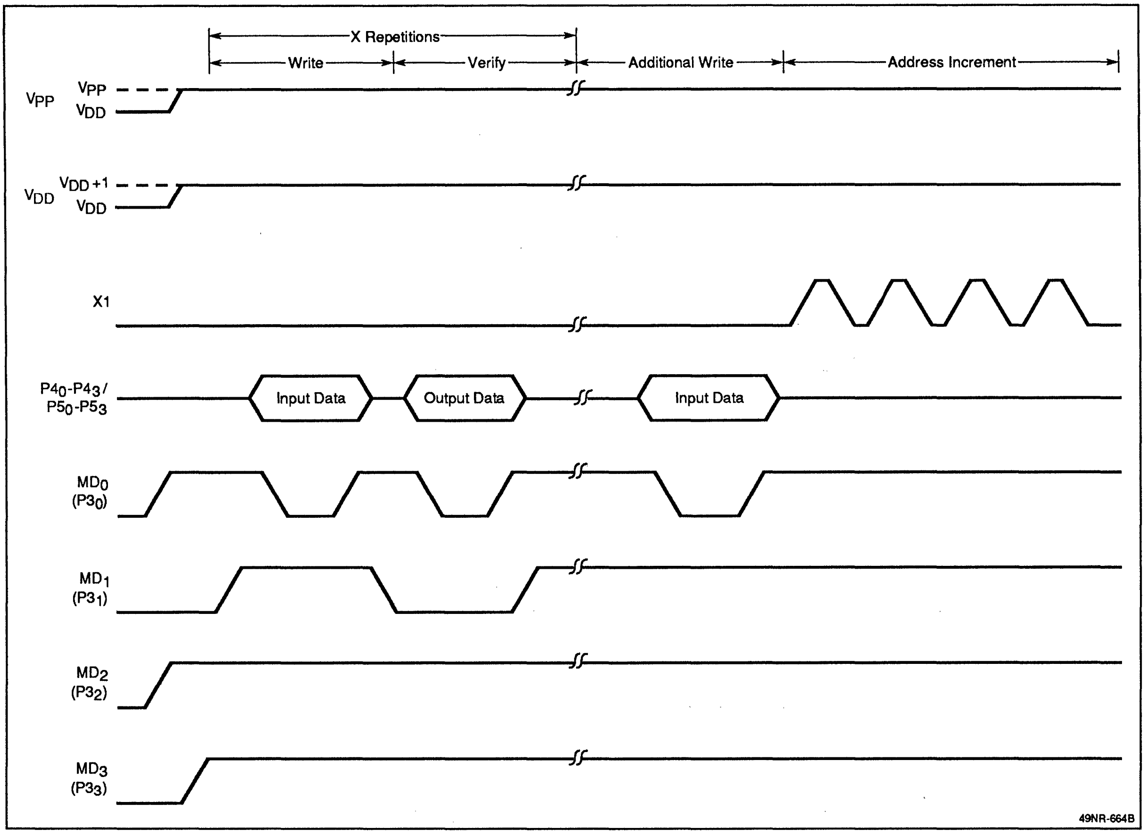
(14) Select the program memory address clear mode.

(15) Change the voltage on V_{DD} and V_{PP} to 5 V.

(16) Turn off power.

The timing for steps 2-12 is shown in figure 19.

Figure 19. Timing Diagram for Program Memory Write/Verify



Program Memory Read. The program memory read procedure follows:

(1) Connect unused pins to V_{SS} through a pull-down resistor. Hold X1 low.

(2) Supply 5 V to V_{DD} and V_{PP}.

(3) Wait for 10 µs.

(4) Select the program memory address clear mode.

(5) Change the voltage on V_{DD} to 6 V and on V_{PP} to 12.5 V.

(6) Select the program inhibit mode.

(7) Select the verify mode. When four clock pulses are input to X1, one address of data is output.

(8) Select the program inhibit mode.

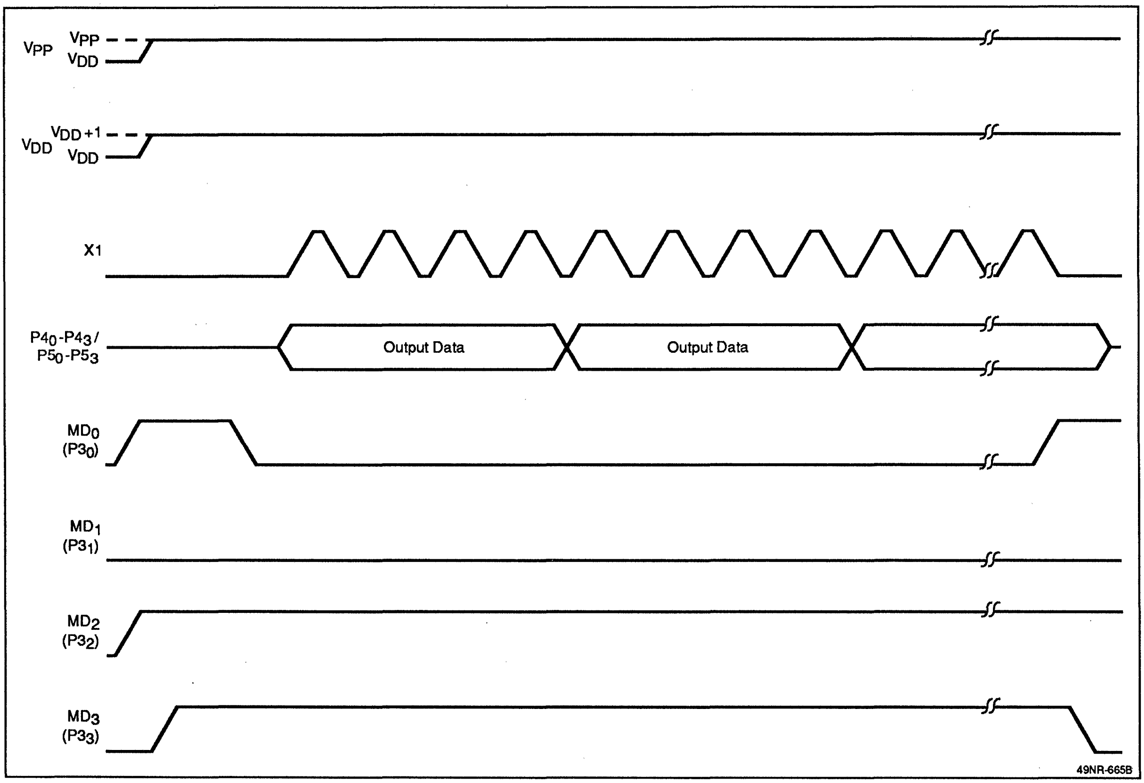
(9) Select the program memory address clear mode.

(10) Change the voltage on V_{DD} and V_{PP} to 5 V.

(11) Turn off power.

The timing for steps 2-9 is shown in figure 20.

Figure 20. Timing Diagram for Program Memory Read



INSTRUCTIONS

The μPD75328 provides a powerful set of 103 instructions.

Instruction Timing

The minimum instruction execution time is 0.95 μs with a crystal frequency of 4.19 MHz. The processor clock control (PCC) register is used to program the CPU instruction execution time to 0.95 μs, 1.91 μs, or 15.3 μs (assuming a 4.19 MHz crystal). Power consumption can be reduced by lowering the CPU speed.

Instruction Set

The instruction set contains the following features:

- Versatile bit manipulation instructions
- Four-bit manipulation instructions
- Eight-bit data transfer instructions
- GETI instruction to reduce program size
- Vertically stored and base correction instructions
- Table reference instructions
- One-byte relative branch instructions

Organization. Tables 12-15 define the operands, symbols, and addressing symbols found in table 16. Table 16 lists the instruction set encodings by instruction groups.

Clock Cycles. One machine cycle equals one CPU clock cycle ϕ. The PCC selects one of four available CPU cycle speeds.

Skip Cycles. S equals the number of extra machine cycles required for skip operation when executing a skip instruction:

- S = 0, No skip
- S = 1, one- or two-byte instruction or GETI instruction is skipped
- S = 2, three-byte instruction is skipped (BR !addr, CALL !addr instruction)

Table 12. Operand Formats and Values

Format	Values
reg	X, A, B, C, D, E, H, L
reg1	X, B, C, D, E, H, L
rp	XA, BC, DE, HL
rp1	BC, DE, HL
rp2	BC, DE
rpa	HL, DE, DL
rpa1	DE, DL
n4	4-bit immediate data or label
n8	8-bit immediate data or label
mem (Note 1)	8-bit immediate data or label
bit	2-bit immediate data or label
fmem	FB0H-FBFH, FF0H-FFFH immediate data or label
pmem	FC0H-FFFH immediate data or label
addr	0000H-177FH immediate data or label
caddr	12-bit immediate data or label
faddr	11-bit immediate data or label
taddr	20H-7EH immediate data (bit 0 = 0) or label
PORTn	PORT 0-8
IExxx	IEBT, IECSI, IET0, IE0, IE1, IE2, IE4, IEW
MBn	MB0, MB1, MB15

Notes:

(1) Only the even memory address is used in 8-bit data processing.

Table 13. Instruction Set Symbol Identifiers

Symbol	Description
A	A register (4-bit accumulator)
B	B register (4-bit accumulator)
C	C register (4-bit accumulator)
D	D register (4-bit accumulator)
E	E register (4-bit accumulator)
H	H register (4-bit accumulator)
L	L register (4-bit accumulator)
X	X register (4-bit accumulator)
XA	XA register pair (8-bit accumulator)
BC	BC register pair
DE	DE register pair
HL	HL register pair
DL	DL register pair
PC	Program counter
SP	Stack pointer
CY	Carry flag (bit accumulator)
PSW	Program status word
MBE	Memory bank enable flag
PORTn	Port 0-8
IME	Interrupt master enable flag
IExxx	Interrupt enable flag
MBS	Memory bank selection register
PCC	Processor clock control register
.	Separation between address and bit
(xx)	Contents addressed by xx
xxH	Hexadecimal data

Table 14. Instruction Code Symbols

reg,reg1

R_2	R_1	R_0	Register	
0	0	0	A	reg
0	0	1	X	reg,reg1
0	1	0	L	reg,reg1
0	1	1	H	reg,reg1
1	0	0	E	reg,reg1
1	0	1	D	reg,reg1
1	1	0	C	reg,reg1
1	1	1	B	reg,reg1

@rpa,@rpa1

Q_2	Q_1	Q_0	Addressing	
0	0	1	@HL	@rpa
1	0	0	@DE	@rpa,@rpa1
1	0	1	@DL	@rpa,@rpa1

Register Pairs

P_2	P_1	reg-pair	
0	0	XA	rp
0	1	HL	rp,rp1
1	0	DE	rp,rp1,rp2
1	1	BC	rp,rp1,rp2

IE-xxx

N_5	N_2	N_1	N_0	IExxx
0	0	0	0	IEBT
0	0	1	0	IEW
0	1	0	0	IET0
0	1	0	1	IECSI
0	1	1	0	IE0
0	1	1	1	IE2
1	0	0	0	IE4
1	1	1	0	IE1

Table 15. Addressing Symbols

Symbol	Description	Address Area
*1	MB=MBE ∧ MBS (MBS=0, 1, 15)	Data memory
*2	MB=0	

*3	MBE=0: MB=0 (00H-7FH)
	MB=15 (80H-FFH)
	MBE=1: MB=MBS (MBS=0, 1, 15)	

*4	MB=15, fmem=FB0H-FBFH, FF0H-FFFH	
*5	MB=15, pmem=FC0H-FFFH	
*6	addr=000H-1F7FH	Program memory
*7	addr= (Current PC) -15 to (Current PC) -1 (Current PC) +2 to (Current PC) +16	
*8	caddr=0000H-0FFFH (PC_{12}=0) or 1000H-1F7FH (PC_{12}=1)	
*9	faddr=0000H-07FFH	
*10	taddr=0020H-007FH	

Notes:

(1) MB = Memory bank that can be addressed.

(2) For symbol *2 (MB = 0, regardless of the status of MBE and MBS).

(3) For symbol *4 and *5 (MB = 15, regardless of the status of MBE and MBS).

(4) For symbol *6 through *10 indicates each addressable area.

(5) The addressing symbols are used in the "Addressing Area" column of the instruction set encodings. See table 16.

Table 16. Instruction Set Encodings

Mnemonic	Operand	Operation	Bytes	Machine Cycles	Addressing Area	Skip Conditions
Data Transfers						
MOV	A, #n4	A ← n4	1	1	-	String A
	reg1, #n4	reg1 ← n4	2	2	-	-
	XA, #n8	XA ← n8	2	2	-	String A
	HL, #n8	HL ← n8	2	2	-	String B
	rp2, #n8	rp2 ← n8	2	2	-	-
	A, @HL	A ← (HL)	1	1	*1	-
	A, @rpa1	A ← (rpa1)	1	1	*2	-
	XA, @HL	XA ← (HL)	2	2	*1	-
	@HL, A	(HL) ← A	1	1	*1	-
	@HL, XA	(HL) ← XA	2	2	*1	-
	A, mem	A ← (mem)	2	2	*3	-
	XA, mem	XA ← (mem)	2	2	*3	-
	mem, A	(mem) ← A	2	2	*3	-
	mem, XA	(mem) ← XA	2	2	*3	-
	A, reg1	A ← reg1	2	2	-	-
	XA, rp	XA ← rp	2	2	-	-
	reg1, A	reg1 ← A	2	2	-	-
	rp1, XA	rp1 ← XA	2	2	-	-
XCH	A, @HL	A ↔ (HL)	1	1	*1	-
	A, @rpa1	A ↔ (rpa1)	1	1	*2	-
	XA, @HL	XA ↔ (HL)	2	2	*1	-
	A, mem	A ↔ (mem)	2	2	*3	-
	XA, mem	XA ↔ (mem)	2	2	*3	-
	A, reg1	A ↔ (reg1)	1	1	-	-
	XA, rp	XA ↔ rp	2	2	-	-
MOVT	XA, @PCDE	XA ←(PC$_{12\text{-}8}$+DE)$_{ROM}$	1	3	—	—
	XA, @PCXA	XA ← (PC$_{12\text{-}8}$+XA)$_{ROM}$	1	3	—	—
Arithmetic						
ADDS	A, #n4	A ← A+n4	1	1+S	-	Carry
	A, @HL	A ← A+(HL)	1	1+S	*1	Carry
ADDC	A, @HL	A, CY ←A+(HL)+CY	1	1	*1	-
SUBS	A, @HL	A ← A-(HL)	1	1+S	*1	Borrow
SUBC	A, @HL	A, CY ← A+(HL)-CY	1	1	*1	-
AND	A, #n4	A ← A ∧ n4	2	2	-	-
	A, @HL	A ← A ∧ (HL)	1	1	*1	-
OR	A, #n4	A ← A V n4	2	2	-	-
	A, @HL	A ←A V (HL)	1	1	*1	-
XOR	A, #n4	A ← A∀ n4	2	2	-	-
	A, @HL	A ← A∀ (HL)	1	1	*1	-

Table 16. Instruction Set Encodings (cont)

Mnemonic	Operand	Operation	Bytes	Machine Cycles	Addressing Area	Skip Conditions
Accumulator						
RORC	A	CY ← A_0, A_3 ← CY, A_{n-1} ← A_n	1	1	-	-
NOT	A	A ← $\overline{A}$	2	2	-	-
Increment/Decrement						
INCS	reg	reg ← reg+1	1	1+S	-	reg = 0
	@HL	(HL) ← (HL)+1	2	2+S	*1	(HL) = 0
	mem	(mem) ← (mem)+1	2	2+S	*3	(mem) = 0
DECS	reg	reg ← reg-1	1	1+S	-	reg = FH
Comparison						
SKE	reg, #n4	Skip if reg = n4	2	2+S	-	reg = n4
	@HL, #n4	Skip if (HL) = n4	2	2+S	*1	(HL) = n4
	A, @HL	Skip if A = (HL)	1	1+S	*1	A = (HL)
	A, reg	Skip if A = reg	2	2+S	-	A = reg
Flags						
SET1	CY	CY ← 1	1	1	-	-
CLR1	CY	CY ← 0	1	1	-	-
SKT	CY	Skip if CY = 1	1	1+S	-	CY = 1
NOT1	CY	CY ← $\overline{CY}$	1	1	-	-
Memory Bits						
SET1	mem.bit	(mem.bit) ←1	2	2	*3	-
	fmem.bit	(fmem.bit) ← 1	2	2	*4	-
	pmem.@L	(pmem$_{7-2}$+ ← L$_{3-2}$.bit (L$_{1-0}$)) ← 1	2	2	*5	-
	@H+mem.bit	(H+mem$_{3-0}$.bit) ← 1	2	2	*1	-
CLR1	mem.bit	(mem.bit) ← 0	2	2	*3	-
	fmem.bit	(fmem.bit) ← 0	2	2	*4	-
	pmem.@L	(pmem$_{7-2}$+L$_{3-2}$.bit (L$_{1-0}$)) ← 0	2	2	*5	-
	@H+mem.bit	(H+mem$_{3-0}$.bit) ← 0	2	2	*1	-
SKT	mem.bit	Skip if (mem.bit) = 1	2	2+S	*3	(mem.bit) =1
	fmem.bit	Skip if (fmem.bit) = 1	2	2+S	*4	(fmem.bit) =1
	pmem.@L	Skip if (pmem$_{7-2}$+ L$_{3-2}$.bit (L$_{1-0}$)) = 1	2	2+S	*5	(pmem.@L) = 1
	@H+mem.bit	Skip if (H+mem$_{3-0}$.bit) = 1	2	2+S	*1	(@H+mem.bit) =1
SKF	mem.bit	Skip if (mem.bit) = 0	2	2+S	*3	(mem.bit) =0
	fmem.bit	Skip if (fmem.bit) = 0	2	2+S	*4	(fmem.bit) =0
	pmem.@L	Skip if (pmem$_{7-2}$+ L$_{3-2}$.bit (L$_{1-0}$)) = 0	2	2+S	*5	(pmem.@L) =0
	@H+mem.bit	Skip if (H+mem$_{3-0}$.bit) = 0	2	2+S	*1	(@H+mem.bit) =0
SKTCLR	fmem.bit	Skip if (fmem.bit) = 1 and clear	2	2+S	*4	(fmem.bit) =1
	pmem.@L	Skip if (pmem$_{7-2}$+ L$_{3-2}$.bit (L$_{1-0}$)) = 1 and clear	2	2+S	*5	(pmem.@L) =1
	@H+mem.bit	Skip if (H+mem$_{3-0}$.bit) = 1 and clear	2	2+S	*1	(@H+mem.bit) =1

Table 16. Instruction Set Encodings (cont)

Mnemonic	Operand	Operation	Bytes	Machine Cycles	Addressing Area	Skip Conditions
Memory Bits (cont)						
AND1	CY, fmem.bit	CY ← CY $\wedge$ (fmem.bit)	2	2	*4	-
	CY, pmem.@L	CY ← CY $\wedge$ (pmem$_{7\text{-}2}$+ L$_{3\text{-}2}$.bit (L$_{1\text{-}0}$))	2	2	*5	-
	CY, @H+mem.bit	CY ← CY $\wedge$ (H+mem$_{3\text{-}0}$.bit)	2	2	*1	-
OR1	CY, fmem.bit	CY ← CY V (fmem.bit)	2	2	*4	-
	CY, pmem.@L	CY ←CY V (pmem$_{7\text{-}2}$+ L$_{3\text{-}2}$.bit (L$_{1\text{-}0}$))	2	2	*5	-
	CY, @H+mem.bit	CY ← CY V (H+mem$_{3\text{-}0}$.bit)	2	2	*1	-
XOR1	CY, fmem.bit	CY ← CY $\veebar$ (fmem.bit)	2	2	*4	-
	CY, pmem.@L	CY ←CY $\veebar$ (pmem$_{7\text{-}2}$+L$_{3\text{-}2}$.bit (L$_{1\text{-}0}$))	2	2	*5	-
	CY, @H+mem.bit	CY ← CY $\veebar$ (H+mem$_{3\text{-}0}$.bit)	2	2	*1	-
BR	addr	PC$_{12\text{-}0}$ ← addr (appropriate instructions are selected from BR !addr, BRCB !caddr, and BR $addr by the assembler)	-	-	*6	-
	!addr	PC$_{12\text{-}0}$ ← addr	3	3	*6	-
	$addr	PC$_{12\text{-}0}$ ←addr	1	2	*7	-
BRCB	!caddr	PC$_{11\text{-}0}$ ←caddr$_{11\text{-}0}$	2	2	*8	-
Subroutine						
CALL	!addr	(SP-1) ← (PC$_{7\text{-}4}$), (SP-2) ← PC$_{3\text{-}0}$ (SP-3) ← (MBE, 0, 0, PC$_{12}$), (SP-4) ← PC$_{11\text{-}8}$, PC$_{12\text{-}0}$ ← addr, SP ← SP-4	3	3	*6	-
CALLF	!faddr	(SP-1) ← PC$_{7\text{-}4}$, (SP-2) ← PC$_{3\text{-}0}$ (SP-3) ← (MBE, 0, 0, PC$_{12}$), (SP-4) ← PC$_{11\text{-}8}$, SP ← (SP-4), PC ←(00, A$_{10\text{-}0}$)	2	2	*9	-
RET	-	PC$_{11\text{-}8}$ ← (SP), (MBE, PC$_{12}$) ← (SP+1), PC$_{3\text{-}0}$ ← (SP+2), PC$_{7\text{-}4}$ ← (SP+3), SP ← (SP+4)	1	3	-	-
RETS	-	PC$_{11\text{-}8}$ ← (SP), (MBE, PC$_{12}$) ← (SP+1), PC$_{3\text{-}0}$ (SP+2), PC$_{7\text{-}4}$ ← (SP+3) SP ← SP+4, then skip unconditionally	1	3+S	-	Unconditional
RETI	-	PC$_{11\text{-}8}$ ← (SP), PC$_{12}$ ← (SP+1) PC$_{3\text{-}0}$ ← (SP+2), PC$_{7\text{-}4}$ (SP+3) PSW$_L$ ← (SP+4), PSW$_H$ ← (SP+6)	1	3	-	-
PUSH	rp	(SP-1) (SP-2) ← rp, SP ← SP-2	1	1	-	-
	BS	(SP-1) ← MBS, (SP-2) ← 0, SP ← SP-2	2	2	-	-
POP	rp	rp ← (SP+1) (SP), SP ← SP+2	1	1	-	-
	BS	MBS ← (SP+1), SP ← SP+2	2	2	-	-

4

Table 16. Instruction Set Encodings (cont)

Mnemonic	Operand	Operation	Bytes	Machine Cycles	Addressing Area	Skip Conditions
Interrupt						
EI	-	IME ← 1	2	2	-	-
	IExxx	IExxx ← 1	2	2	-	-
DI	-	IME ← 0	2	2	-	-
	IExxx	IExxx ← 0	2	2	-	-
Input/Output						
IN (Note 1)	A, PORTn	A ← PORTn (n = 0-8)	2	2	-	-
	XA, PORTn	XA ← PORT$_{n+1}$, PORTn (n = 4,6)	2	2	-	-
OUT (Note 1)	PORTn, A	PORTn ← A (n = 2-8)	2	2	-	-
	PORTn, XA	PORT$_{n+1}$, PORTn ← XA (n = 4,6)	2	2	-	-
CPU Control						
HALT	-	Set HALT Mode (PCC.2 ← 1)	2	2	-	-
STOP	-	Set STOP Mode (PCC.3 ← 1)	2	2	-	-
NOP	-	No Operation	1	1	-	-
Miscellaneous						
SEL	MBn	MBS ← n (n = 0, 1, 15)	2	2	-	-
GETI	taddr	When (taddr)$_{7-6}$ = 00; PC$_{12-0}$ ← (taddr)$_{4-0}$ + (taddr+1)	1	3	*10	Depends on the referenced instruction
		When (taddr)$_{7-6}$ = 01; (SP-4) (SP-1) (SP-2) ← PC$_{11-0}$ (SP-3) ← (MBE, 0, 0, PC$_{12}$) PC$_{12-0}$ ← (taddr)$_{4-0}$ + (taddr+1) SP = SP-4				
		When (taddr)$_{7-6}$ = 10; (taddr)(taddr+1) instruction is executed				

Notes:

(1) When executing an IN or OUT instruction, MBE = 0 or MBE = 1 and MBS = 15.

Section 5
μPD7800 Series:
8-Bit, General-Purpose Microcomputers

NEC

NEC Electronics Inc.

µPD78C1x/C1xA/CG14/CP14 8-Bit CMOS Microcomputers With A/D Converter

Description

The family of single-chip microcomputers covered by this data sheet includes the following types:

µPD78C10	µPD78C10A	µPD78CG14
µPD78C11	µPD78C11A	µPD78CP14
µPD78C14	µPD78C12A	
	µPD78C14A	

These microcomputers integrate sophisticated on-chip peripheral functions normally provided by external components. Their internal 16-bit ALU and data paths, combined with a powerful instruction set and addressing, make the devices appropriate in data processing as well as control applications.

The devices integrate a 16-bit ALU, 4K-, 8K-, or 16K-byte ROM, 256-byte RAM, an eight channel A/D converter, a multifunction 16-bit timer/event counter, two 8-bit timers, a USART, and two zero-cross detect inputs on a single die, allowing their use in fast, high-end processing applications. This involves analog signal interface and processing.

The µPD78C1x/C1xA/Cx14 family includes: 4K-, 8K-, and 16K-byte mask ROM devices, embedded with a custom customer program; ROMless devices for use with up to 64K-bytes of external memory; 16K-byte piggyback EPROM device for prototyping; 16K-byte EPROM or OTP ROM devices for prototyping and low-volume production. The µPD78C11A/C12A/C14A also have mask optional pullup resistors available on ports A, B, and C.

Features

- □ CMOS technology
  - — 25 mA operating current (78C10/C10A/C11/C11A/C12A)
  - — 30 mA operating current (78C14/C14A)
- □ Complete single-chip microcomputer
  - — 16-bit ALU
  - — 4K, 8K, or 16K x 8 ROM
  - — 256-byte RAM
- □ 44 I/O lines
- □ Mask optional pullup resistors
  - — Ports A, B, and C
  - — µPD78C11A/C12A/C14A only
- □ Two zero-cross detect inputs
- □ Two 8-bit timers
- □ Expansion capabilities
  - — 8085A-like bus
  - — 60K-byte external memory address range
- □ Eight-channel, 8-bit A/D converter
  - — Autoscan mode
  - — Channel select mode
- □ Full-duplex USART
  - — Synchronous and asynchronous
- □ 159 instructions
  - — 16-bit arithmetic, multiply, and divide
  - — HALT and STOP instructions
- □ 0.8-µs instruction cycle time (15-MHz operation)
- □ Prioritized interrupt structure
  - — Three external
  - — Eight internal
- □ Standby function
- □ On-chip clock generator

Ordering Information

Part Number	Package	ROM
µPD78C10CW	64-pin plastic SDIP	ROMless
µPD78C10G-36	64-pin plastic QUIP	
µPD78C10G-1B	64-pin plastic QFP (Resin thickness 2.05 mm)	
µPD78C10GF-3BE	64-pin plastic QFP (Resin thickness 2.7 mm)	
µPD78C10L	68-pin PLCC	
µPD78C10ACW	64-pin plastic SDIP	ROMless
µPD78C10AGF-3BE	64-pin plastic QFP	
µPD78C10AGQ-36	64-pin plastic QUIP	
µPD78C10AL	68-pin PLCC	
µPD78C11CW-xxx	64-pin plastic SDIP	4K mask ROM
µPD78C11G-xxx-36	64-pin plastic QUIP	
µPD78C11G-xxx-1B	64-pin plastic QFP (Resin thickness 2.05 mm)	
µPD78C11GF-xxx-3BE	64-pin plastic QFP (Resin thickness 2.7 mm)	
µPD78C11L-xxx	68-pin PLCC	

Ordering Information (cont)

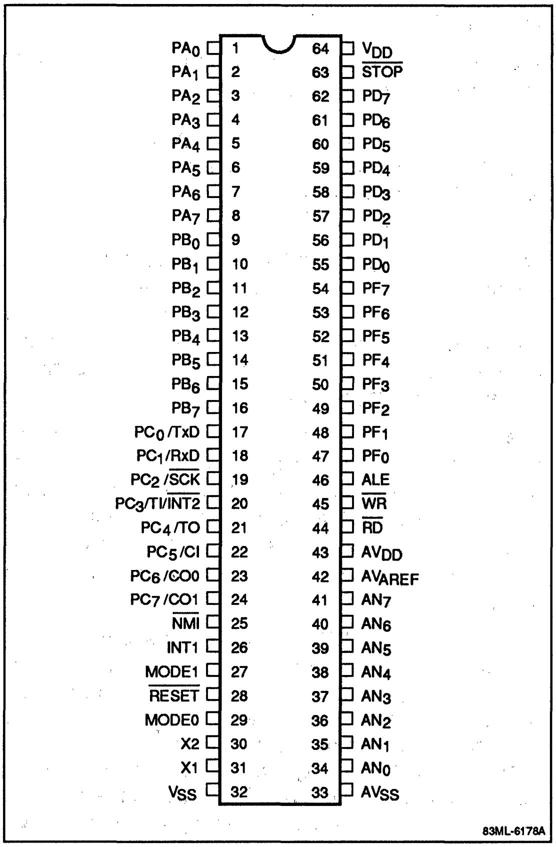
Part Number	Package	ROM
μPD78C11ACW-xxx	64-pin plastic SDIP	4K mask ROM
μPD78C11AGF-xxx-3BE	64-pin plastic QFP	
μPD78C11AGQ-xxx-36	64-pin plastic QUIP	
μPD78C11AL-xxx	68-pin PLCC	
μPD78C12ACW-xxx	64-pin plastic SDIP	8K mask ROM
μPD78C12AGF-xxx-3BE	64-pin plastic QFP	
μPD78C12AG-xxx-36	64-pin plastic QUIP	
μPD78C12AL-xxx	68-pin PLCC	
μPD78C14CW-xxx	64-pin plastic SDIP	16K mask ROM
μPD78C14G-xxx-36	64-pin plastic QUIP	
μPD78C14G-xxx-1B	64-pin plastic QFP (Resin thickness 2.05 mm)	
μPD78C14GF-xxx-3BE	64-pin plastic QFP (Resin thickness 2.7 mm)	
μPD78C14L-xxx	68-pin PLCC	
μPD78C14AG-xxx-AB8	64-pin plastic QFP (Interpin pitch 0.8 mm)	16K mask ROM
μPD78CG14E	64-pin ceramic piggyback QUIP	4/8/16K piggyback EPROM
μPD78CP14CW	64-pin plastic SDIP	16K OTP ROM
μPD78CP14G-36	64-pin plastic QUIP	
μPD78CP14GF-3BE	64-pin plastic QFP	
μPD78CP14L	68-pin PLCC	
μPD78CP14DW	64-pin ceramic SDIP with window	16K UV EPROM
μPD78CP14R	64-pin ceramic QUIP with window	

Notes:

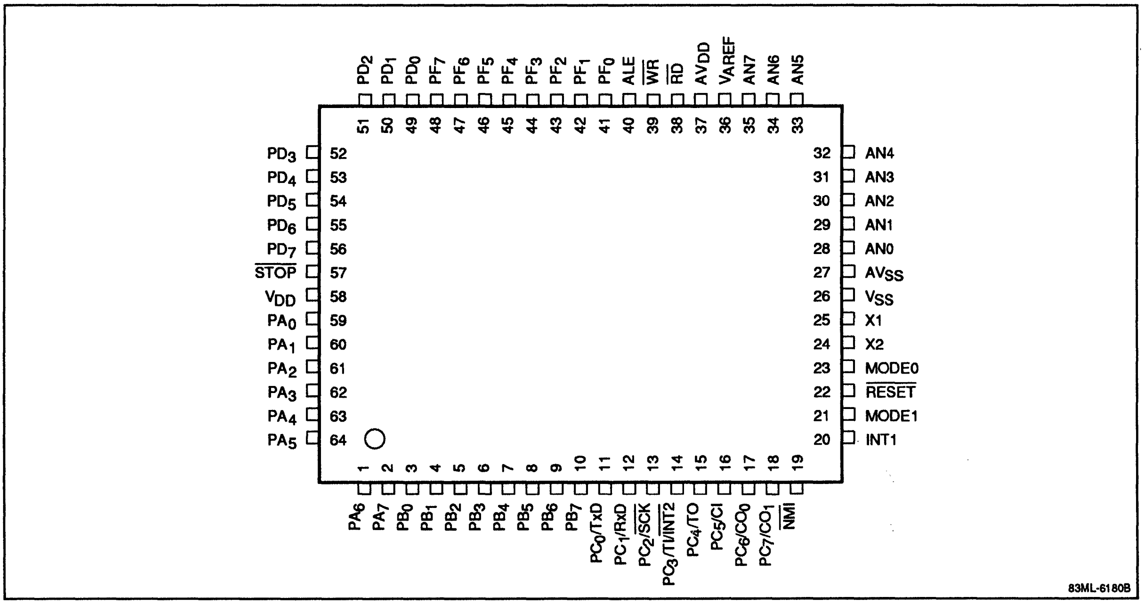
(1) xxx indicates ROM code suffix.

Pin Configurations

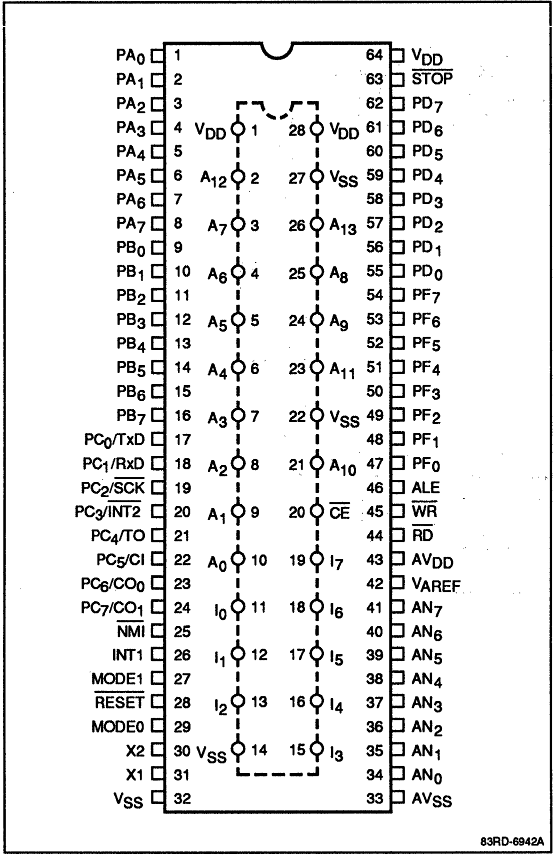
64-Pin QUIP or SDIP (Plastic or Ceramic)



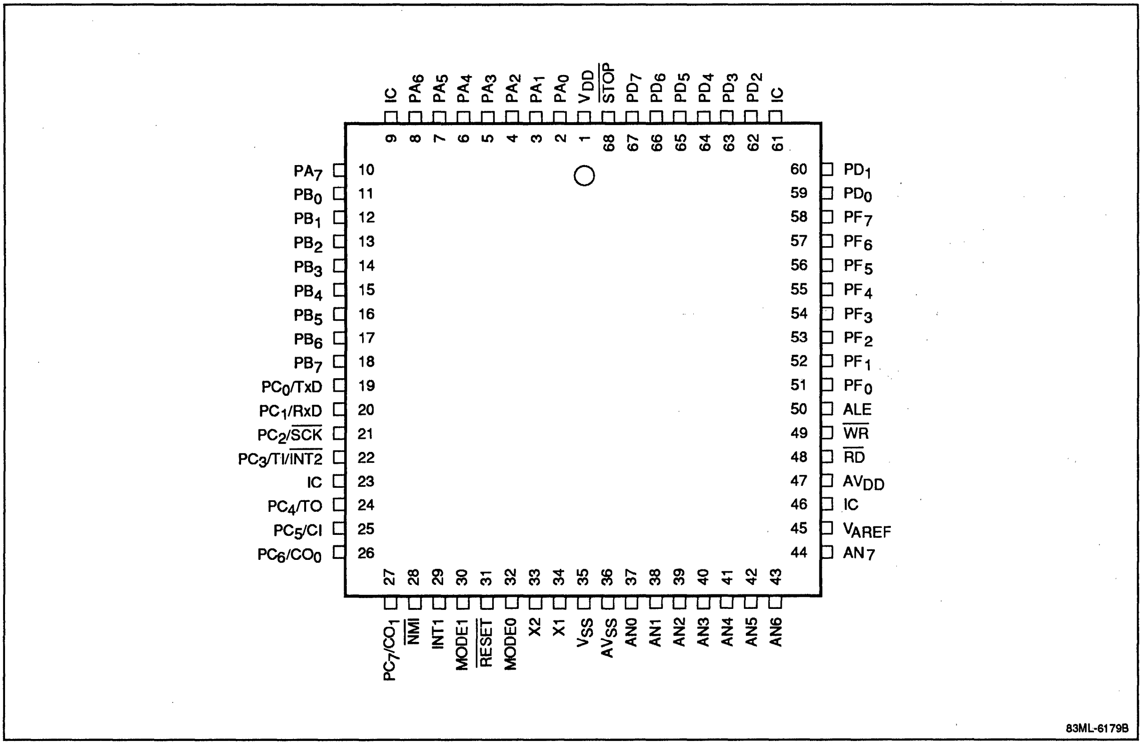
64-Pin Plastic QFP



64-Pin Ceramic Piggyback QUIP



68-Pin PLCC


9 IC
8 PA6
7 PA5
6 PA4
5 PA3
4 PA2
3 PA1
2 PA0
1 V_{DD}
68 $\overline{STOP}$
67 PD7
66 PD6
65 PD5
64 PD4
63 PD3
62 PD2
61 IC
10 PA_7
11 PB_0
12 PB_1
13 PB_2
14 PB_3
15 PB_4
16 PB_5
17 PB_6
18 PB_7
19 PC_0/TxD
20 PC_1/RxD
21 $PC_2/\overline{SCK}$
22 $PC_3/TI/\overline{INT2}$
23 IC
24 PC_4/TO
25 PC_5/CI
26 PC_6/CO_0
27 PC_7/CO_1
28 $\overline{NMI}$
29 INT1
30 MODE1
31 $\overline{RESET}$
32 MODE0
33 X2
34 X1
35 V_{SS}
36 AV_{SS}
37 AN0
38 AN1
39 AN2
40 AN3
41 AN4
42 AN5
43 AN6
44 AN_7
45 V_{AREF}
46 IC
47 AV_{DD}
48 $\overline{RD}$
49 $\overline{WR}$
50 ALE
51 PF_0
52 PF_1
53 PF_2
54 PF_3
55 PF_4
56 PF_5
57 PF_6
58 PF_7
59 PD_0
60 PD_1
83ML-6179B

Pin Identification

Symbol	Function
ALE	Address latch enable output
AN0-AN7	A/D converter analog inputs 0-7
INT1	Interrupt request 1 input
MODE0	Mode 0 input; I/O memory output
MODE1	Mode 1 input
$\overline{\text{NMI}}$	Nonmaskable interrupt input
PA_0-PA_7	Port A I/O
PB_0-PB_7	Port B I/O
PC_0/TxD	Port C I/O line 0; transmit data output
PC_1/RxD	Port C I/O line 1; receive data input
$PC_2/\overline{\text{SCK}}$	Port C I/O line 2; serial clock I/O
$PC_3/TI/\overline{\text{INT2}}$	Port C I/O line 3; timer input; interrupt request 2 input
PC_4/TO	Port C I/O line 4; timer output
PC_5/CI	Port C I/O line 5; counter input
PC_6, PC_7/ CO_0,CO_1	Port C I/O lines 6, 7; counter outputs 0, 1
PD_0-PD_7	Port D I/O; expansion memory address, data bus (bits AD_0-AD_7)
PF_0-PF_7	Port F I/O; expansion memory address, (bits AB_8-AB_{15})
$\overline{\text{RD}}$	Read strobe output
$\overline{\text{RESET}}$	Reset input
$\overline{\text{STOP}}$	Stop mode control input
V_{AREF}	A/D converter reference voltage
$\overline{\text{WR}}$	Write strobe output
X1, X2	Crystal connections 1, 2
AV_{DD}	A/D converter power supply voltage
AV_{SS}	A/D converter power supply ground
V_{DD}	5 V power supply
V_{SS}	Ground
IC	Internal connection

Pin Identification μPD78CG14E Upper EPROM Pins

Symbol	Pin	Function
A_0-A_{13}	2-10, 21 23-26	14-bit program counter (PC_0-PC_{13}) output used as 27C256/27C256A address signals
$\overline{\text{CE}}$	20	Chip enable signal for 27C256/27C256A; high-level output (during STOP or HALT), otherwise, low-level output
I_0-I_7	11-13 15-19	8-bit input of data read from 27C256/27C256A
V_{DD}	1	Same potential as lower V_{DD} pin; V_{CC} power supply line (V_{PP}) for 27C256/27C256A
V_{DD}	28	Same potential as lower V_{DD} pin; V_{CC} power supply line (V_{CC}) for 27C256/27C256A
V_{SS}	14	Same potential as lower V_{SS} pin connected to the 27C256/27C256A GND pin
V_{SS}	22	Same potential as lower V_{SS} pin; $\overline{\text{OE}}$ signal (always low) input to 27C256/27C256A
V_{SS}	27	Same potential as lower V_{SS} pin; A_{14} signal (always low) input to 27C256/27C256A

PIN FUNCTIONS

ALE (Address Latch Enable)

The ALE output is used to latch the address of PD_0-PD_7 into an external latch.

AN0-AN7 (Analog Inputs)

These are the eight analog inputs to the A/D converter. AN4-AN7 can also be used as a digital input for falling edge detection.

CI (Counter Input)

External pulse input to timer/event counter.

CO_0, CO_1 (Counter Outputs)

Programmable waveform outputs based on timer/event counter.

INT1 (Interrupt Request 1)

INT1 is a rising edge triggered, maskable interrupt input. It is also an ac-input, zero-cross detection terminal.

If the optional pullup resistor is specified for this pin on the μPD78C11A/C12A/C14A, the zero-cross detection circuitry will not function.

$\overline{\text{INT2}}$ (Interrupt Request 2)

$\overline{\text{INT2}}$ is a falling edge triggered, maskable interrupt input. It is also an ac-input, zero-cross detection terminal.

MODE0, MODE1 (Mode 0, 1)

The MODE0 and MODE1 inputs select the amount of external memory. MODE0 outputs the $\overline{\text{IO}}$ signal, and MODE1 outputs the $\overline{\text{M1}}$ signal. An external pullup resistor to V_{DD} is required if the input is to be a logic high.

The value of this pullup resistor, R, is dependent on t_{CYC} and is calculated as follows: R in KΩ is $4 \leq R \leq 0.4\, t_{CYC}$ where t_{CYC} is in ns units.

$\overline{\text{NMI}}$ (Nonmaskable Interrupt)

Falling edge, Schmitt triggered nonmaskable interrupt input.

PA_0-PA_7 (Port A)

Port A is an 8-bit three-state port. Each bit is independently programmable as either input or output. Reset makes all lines of port A inputs. Mask optional pullup resistors are available on the μPD78C11A/C12A/C14A.

PB_0-PB_7 (Port B)

Port B is an 8-bit three-state port. Each bit is independently programmable as either input or output. Reset makes all lines of port B inputs. Mask optional pullup resistors are available on the μPD78C11A/C12A/C14A.

PC_0-PC_7 (Port C)

Port C is an 8-bit three-state port. Each bit is independently programmable as either input or output. Alternatively, the lines of port C can be used as control lines for the USART, interrupts, and timer. Reset makes all lines of port C inputs. Mask optional pullup resistors are available on the μPD78C11A/C12A/C14A.

PD_0-PD_7 (Port D)

Port D is an 8-bit three-state port. It can be programmed as either 8 bits of input or 8 bits of output. When external expansion memory is used, port D acts as the multiplexed address/data bus.

PF_0-PF_7 (Port F)

Port F is an 8-bit three-state port. Each bit is independently programmable as either input or output. When external expansion memory is used, port F outputs the high-order address bits.

$\overline{\text{RD}}$ (Read Strobe)

The three-state $\overline{\text{RD}}$ output goes low to gate data from external devices onto the data bus. $\overline{\text{RD}}$ goes high during reset.

$\overline{\text{RESET}}$ (Reset)

When the Schmitt-triggered $\overline{\text{RESET}}$ input is brought low, it initializes the device.

RxD (Receive Data)

Serial data input terminal.

$\overline{\text{SCK}}$ (Serial Clock)

Output for the serial clock when internal clock is used. Input for serial clock when external clock is used.

$\overline{\text{STOP}}$ (STOP Mode Control Input)

A low-level input on $\overline{\text{STOP}}$ (Schmitt-triggered input) stops the system clock oscillator.

TI (Timer Input)

Timer input terminal.

TO (Timer Output)

The output of TO is a square wave with a frequency determined by the timer/counter.

TxD (Transmit Data)

Serial data output terminal.

V_{AREF} (A/D Converter Reference)

V_{AREF} sets the upper limit for the A/D conversion range.

$\overline{\text{WR}}$ (Write Strobe)

The three-state $\overline{\text{WR}}$ output goes low to indicate that the data bus holds valid data. It is a strobe signal for external memory or I/O write operations. $\overline{\text{WR}}$ goes high during reset.

X1, X2 (Crystal Connections)

X1 and X2 are the system clock crystal oscillator terminals. X1 is the input for an external clock.

AV_{DD} (A/D Converter Power)

This is the power supply voltage for the A/D converter.

AV_{SS} (A/D Converter Power Ground)

AV_{SS} is the ground potential for the A/D converter power supply.

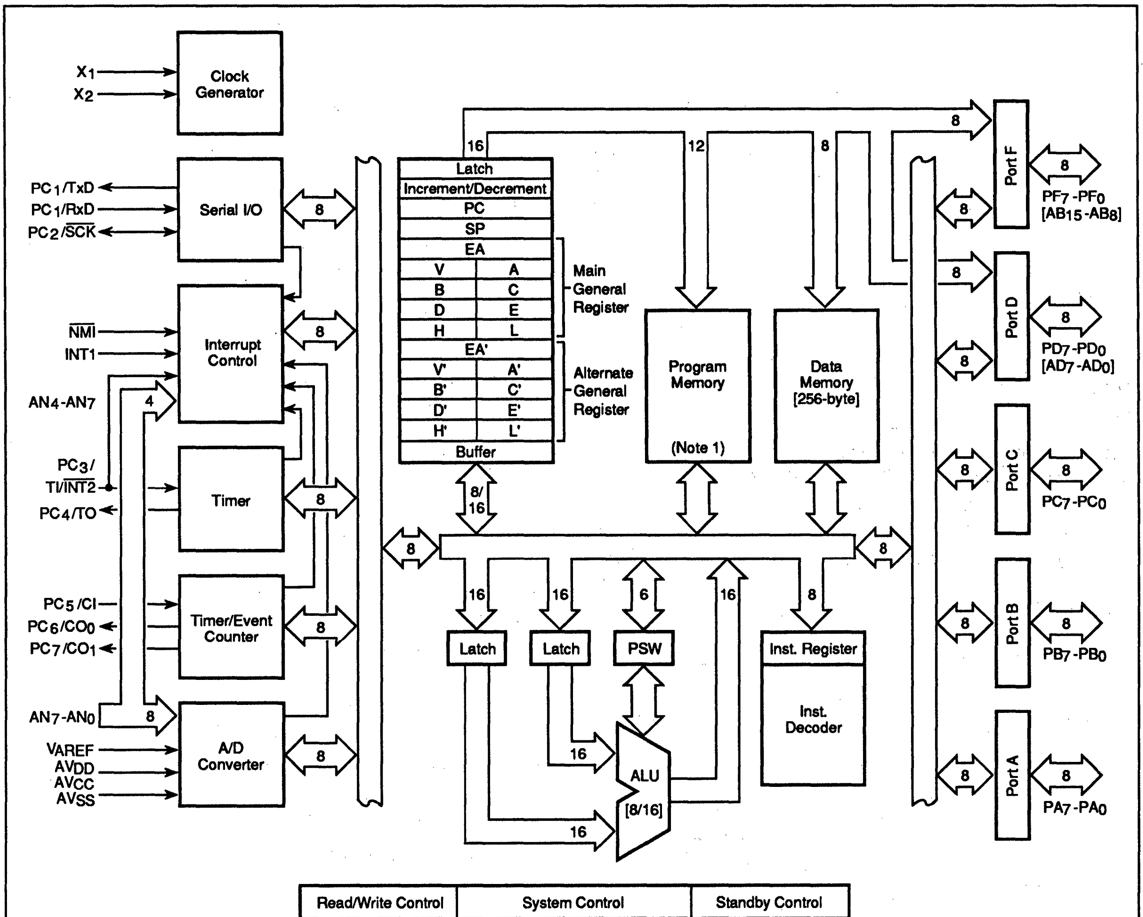
V_{DD} (Power Supply)

V_{DD} is the +5-volt power supply.

V_{SS} (Ground)

Ground potential.

Block Diagram



Note:
1. On-Chip ROM

78C10/C10A	: 0
78C11/C11A	: 4096 Bytes
78C12A	: 8192 Bytes
78C14/C14A	: 16384 Bytes
78CP14	: 16384 Bytes EPROM/OTP ROM

FUNCTIONAL DESCRIPTION

Memory Map

The μPD78C1x/C1xA/Cx14 family can directly address up to 64K bytes of memory. Except for the on-chip ROM (or PROM) and RAM (FF00H-FFFFH), any memory location can be used as ROM or RAM. The memory map, figure 1, defines the 0 to 64K-byte memory space for the μPD78C1x/C1xA/Cx14 family.

The μPD78CG14 and the μPD78CP14 can be programmed in software to have 4K, 8K, or 16K bytes of internal program memory. This programming is transparent to the ROM-based device, allowing easy transfer of code to a ROM-based device.

Input/Output

The μPD78C1x/C1xA/Cx14 family has 44 digital I/O lines, five 8-bit ports (ports A, B, C, D, F), and four digital input lines (AN4-AN7).

Analog Input Lines. AN0-AN7 are configured as analog input lines for the on-chip A/D converter. Lines AN4-AN7 can be used as digital input lines for falling edge detection.

Port A, Port B, Port C, Port F. Each line of these ports can be individually programmed as an input or output. When used as I/O ports, all have latched outputs and high-impedance inputs. On the μPD78C11A/C12A/C14A, mask optional pullup resistors are available for ports A, B, and C.

Port D. Port D can be programmed as a byte input or a byte output.

Control Lines. Under software control, each line of port C can be configured individually as a control line for the serial interface, timer, and timer/counter or as an I/O port.

Memory Expansion. In addition to the single-chip operation mode, the μPD78C1x/C1xA/Cx14 family has four memory expansion modes. Under software control, port D can provide a multiplexed low-order address and data bus; port F can provide a high-order address bus. Table 1 shows the relation between memory expansion modes and the pin configurations of port D and port F.

Table 1. Memory Expansion Modes and Port Configurations

Memory Expansion	Port	Port Configuration
None	Port D	I/O port
	Port F	I/O port
256 bytes	Port D	Multiplexed address/ data bus
	Port F	I/O port
4K bytes	Port D	Multiplexed address/ data bus
	Port F (PF_0-PF_3)	Address bus
	Port F (PF_4-PF_7)	I/O port
16K bytes	Port D	Multiplexed address/ data bus
	Port F (PF_0-PF_5)	Address bus
	Port F (PF_6-PF_7)	I/O port
60K bytes	Port D	Multiplexed address/ data bus
	Port F	Address bus

Timers

The two 8-bit timers may be programmed independently or cascaded as a 16-bit timer. The timer can be software set to increment at intervals of four machine cycles (0.8 μs at 15-MHz operation) or 128 machine cycles (25.6 μs at 15-MHz), or to increment on receipt of a pulse at TI. Figure 2 is the block diagram for the timer.

Timer/Event Counter

The 16-bit multifunctional timer/event counter (figure 3) can be used for the following operations:

- Interval timer
- External event counter
- Frequency measurement
- Pulse width measurement
- Programmable frequency and duty cycle waveform output
- Single pulse output

8-Bit A/D Converter

- 8 input channels
- 4 conversion result registers
- 2 powerful operation modes
  — Autoscan mode
  — Channel select mode
- Successive approximation technique
- Absolute accuracy: 0.6% FSR ± 1/2 LSB
- Conversion range: 0 to 5 V
- Conversion time: 38.4 μs
- Interrupt generation

Figure 1. Memory Map

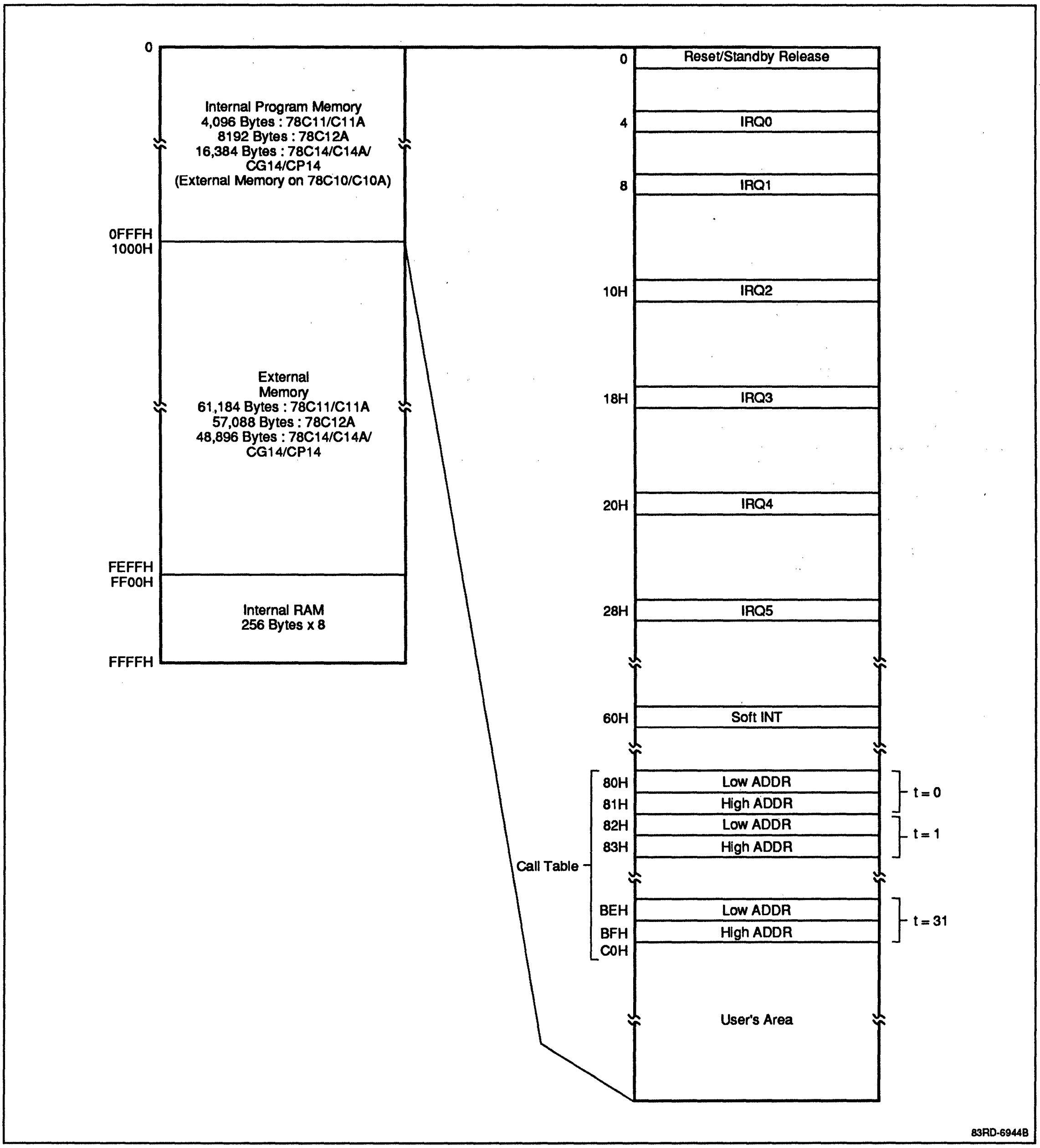


Figure 2. Timer Block Diagram

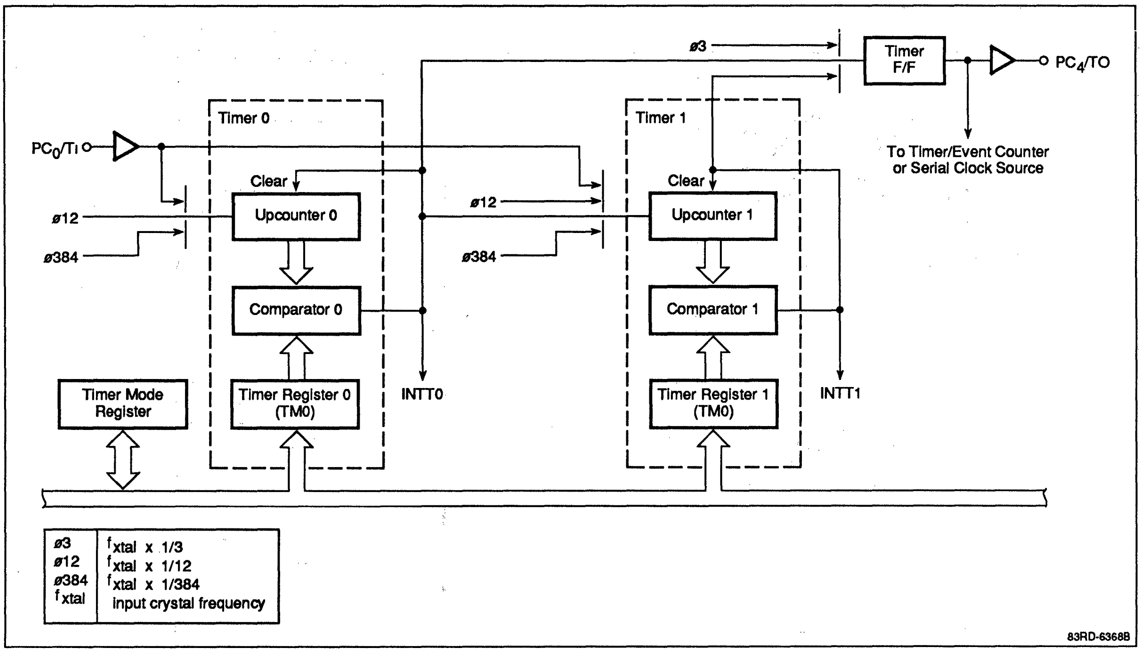
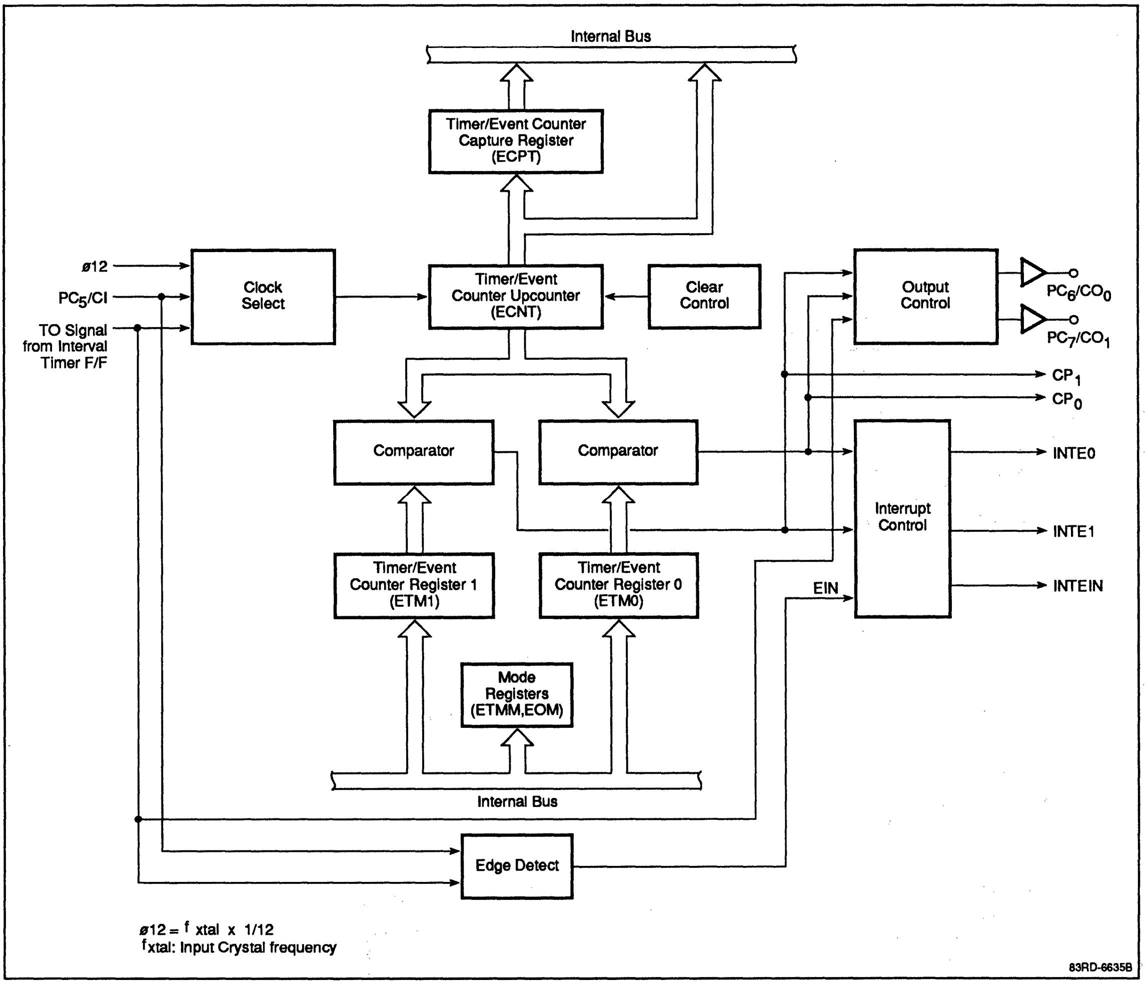


Figure 3. Block Diagram for the Timer/Event Counter


Internal Bus
Timer/Event Counter Capture Register (ECPT)
ø12
PC5/CI
TO Signal from Interval Timer F/F
Clock Select
Timer/Event Counter Upcounter (ECNT)
Clear Control
Output Control
PC6/CO0
PC7/CO1
CP1
CP0
Comparator
Comparator
Interrupt Control
INTE0
INTE1
INTEIN
EIN
Timer/Event Counter Register 1 (ETM1)
Timer/Event Counter Register 0 (ETM0)
Mode Registers (ETMM,EOM)
Internal Bus
Edge Detect
83RD-6635B

ø12 = f xtal x 1/12
fxtal: Input Crystal frequency

Analog/Digital Converter

The μPD78C1x/C1xA/Cx14 family features an 8-bit, high-speed, high accuracy A/D converter. The A/D converter is made up of a 256-resistor ladder and a successive approximation register (SAR). There are four conversion result registers (CR0-CR3).

The eight-channel analog input may be operated in either of two modes. In the select mode, the conversion value of one analog input is sequentially stored in CR0-CR3. In the scan mode, either the upper four channels or the lower four channels may be specified. Then those four channels will be consecutively selected and the conversion results stored sequentially in the four conversion result registers.

Figure 4 is the block diagram for the A/D converter. To stop the operation of the A/D converter and thus reduce power consumption, set V_{AREF} = 0 V.

Interrupt Structure

There are 12 interrupt sources in the μPD78C1x/C1xA/Cx14 family of chips. Three are external interrupts and nine are internal. Table 2 shows 11 interrupt sources divided into seven priority levels where IRQ0 is the highest and IRQ6 is the lowest. See figure 5.

Figure 4. A/D Converter Block Diagram

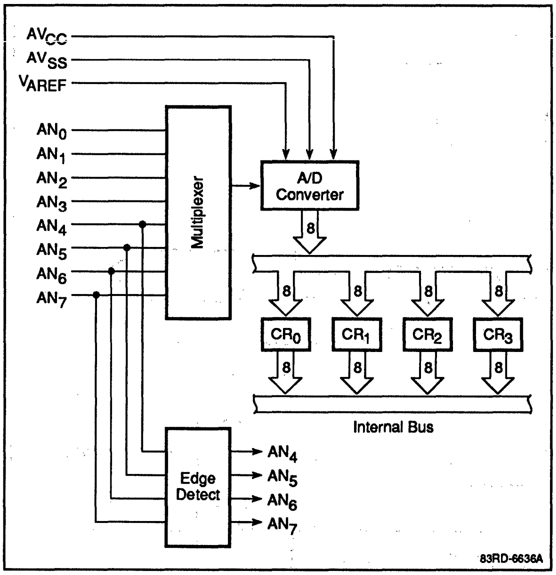
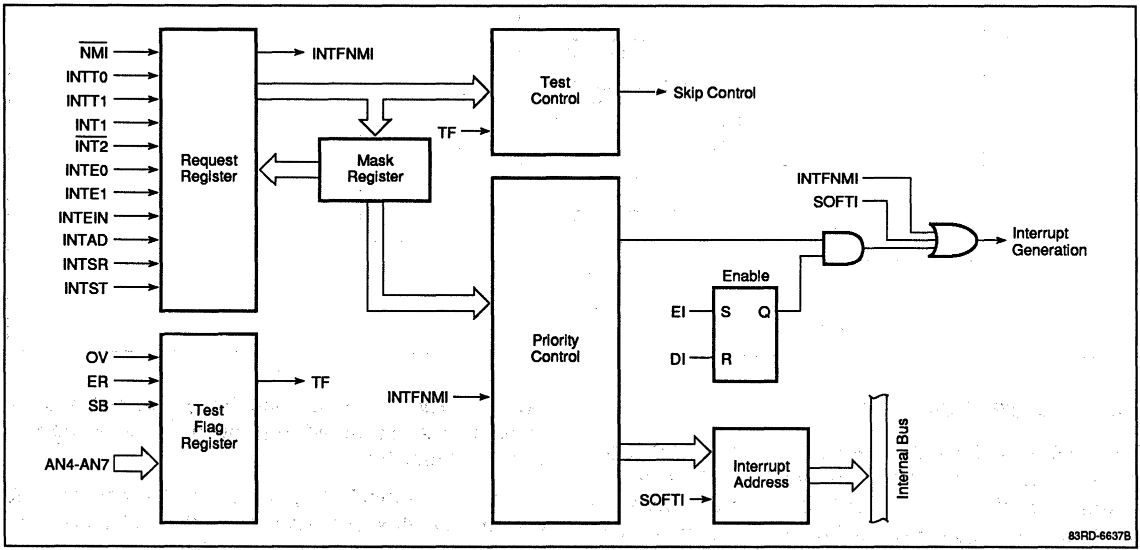


Figure 5. Interrupt Structure Block Diagram



Standby Functions

The μPD78C1x/C1xA/Cx14 family has two standby modes: HALT and STOP. The HALT mode reduces power consumption to 50% of normal operating requirements, while maintaining the contents of on-chip registers, RAM, and control status. The system clock and on-board peripherals continue to operate, but the CPU stops executing instructions. The HALT mode is initiated by executing the HLT instruction. The HALT mode can be released by any nonmasked interrupt or by $\overline{\text{RESET}}$.

The STOP mode reduces power consumption to less than 0.1% of normal operating requirements. There are two STOP modes: type A and type B.

Type A is initiated by executing a STOP instruction. If V_{DD} is held above 2.5 V, the on-board RAM is saved. The oscillator is stopped. The STOP mode can be released by an input on $\overline{\text{NMI}}$ or $\overline{\text{RESET}}$. The user can program oscillator stabilization time up to 52.4 ms via timer 1. By checking the standby flag (SB), the user can determine whether the processor has been in the standby mode or has been powered up.

Type B is initiated by inputting a low level on the $\overline{\text{STOP}}$ input. The RAM contents are saved if V_{DD} is held above 2.5 V. The oscillator is stopped. The STOP mode is released by raising $\overline{\text{STOP}}$ to a high level. The oscillator stabilization time is fixed at 52.4 ms; 52.4 ms after $\overline{\text{STOP}}$ is raised, instruction execution will automatically begin at location 0. You can increase the stabilization time by holding $\overline{\text{RESET}}$ low for the required time period.

Universal Serial Interface

The serial interface can operate in one of three modes: synchronous, asynchronous, and I/O interface. The I/O interface mode transfers data MSB first, for easy interfacing to certain NEC peripheral devices. Synchronous and asynchronous modes transfer data LSB first. Synchronous operation offers two modes of data reception: search and nonsearch. In the search mode, data is transferred one bit at a time from the serial register to the receive buffer. This allows a software search for a sync character. In the nonsearch mode, data transfer from the serial register to the transmit buffer occurs eight bits at a time. Figure 6 shows the universal serial interface block diagram.

Zero-Crossing Detector

The INT1 and $\overline{\text{INT2}}$ terminals (used common to TI and PC_3) can detect the zero-crossing point of low-frequency AC signals. When driven directly, these pins respond as a normal digital input. Figure 7 shows the zero-crossing detection circuitry.

Table 2. Interrupt Sources

Interrupt Request	Interrupt Address	Type of Interrupt	Internal/External
IRQ0	4	$\overline{\text{NMI}}$ (Nonmaskable interrupt)	External
IRQ1	8	INTT0, INTT1 (Coincidence signals from timers 0, 1)	Internal
IRQ2	16	INT1, $\overline{\text{INT2}}$ (Maskable interrupts)	External
IRQ3	24	INTE0, INTE1 (Coincidence signals from timer/event counter)	Internal
IRQ4	32	INTEIN (Falling signal of CI or TO into the timer/event counter)	Internal or External
		INTAD (A/D converter interrupt)	Internal
IRQ5	40	INTSR (Serial receive interrupt)	Internal
		INST (Serial send interrupt)	
IRQ6	96	SOFTI instruction	Internal

Figure 6. Universal Serial Interface Block Diagram

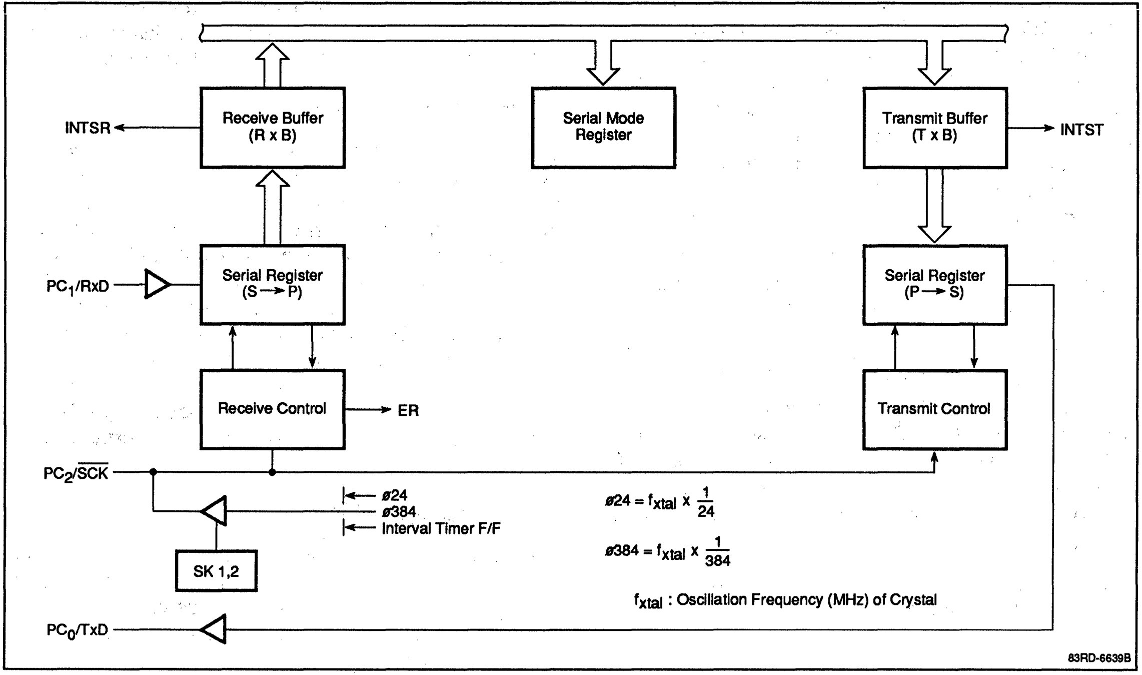
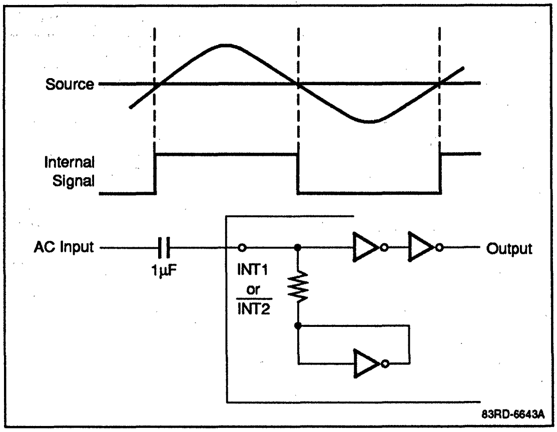


Figure 7. Zero-Crossing Detection Circuit



The zero-crossing detection capability allows you to make the 50-60 Hz power signal the basis for system timing and to control voltage phase-sensitive devices.

To use the zero-cross detection mode, an AC signal of 1.0 to 1.8 V (peak-to-peak) and a maximum frequency of 1 kHz is coupled through an external capacitor to the INT1 and $\overline{\text{INT2}}$ pins.

For the INT1 pin, the internal digital state is sensed as a 0 until the rising edge crosses the average DC level, when it becomes a 1 and an INT1 interrupt is generated.

For the $\overline{\text{INT2}}$ pin, the state is sensed as a 1 until the falling edge crosses the average DC level, when it becomes a 0 and $\overline{\text{INT2}}$ is generated.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A = 25°C

Power supply voltage, V_{DD}	−0.5 to +7.0 V
Power supply voltage, AV_{DD}	AV_{SS} to V_{DD} +0.5 V
Power supply voltage, AV_{SS}	−0.5 to +0.5 V
Power supply voltage, V_{PP} (μPD78CP14 only)	−0.5 to +13.5
Input voltage, V_I	−0.5 to V_{DD}+.5 V
$\overline{\text{STOP}}$ pin (μPD78CP14 only)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD}+.5 V
Output current, low; I_{OL}	
Each output pin	4.0 mA
Total	100 mA
Output current, high; I_{OH}	
Each output pin	−2.0 mA
Total	−50 mA
Reference input voltage, VA_{REF}	−0.5 to AV_{DD} +0.3 V
Operating temperature, T_{OPR} f_{XTAL} ≤ 15 MHz	−40 to +85°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_I	10	pF	f_C = 1 MHz; unmeasured pins returned to 0 V
Output capacitance	C_O	20	pF	
I/O capacitance	C_{IO}	20	pF	

Oscillation Characteristics

T_A = −40 to +85°C; V_{DD} = AV_{DD} = 5 V ± 10% (±5% μPD78CP14);
V_{SS} = AV_{SS} = 0 V; V_{DD} − 0.8 V ≤ AV_{DD} ≤ V_{DD}; 3.4 V ≤ V_{AREF} ≤ AV_{DD}

Resonator	Recommended Circuit	Parameter	Min	Typ	Max	Unit	Conditions
Ceramic resonator (Note 1) or XTAL (Note 2)	(Note 3)	Oscillation frequency (f_{XX})	4		15	MHz	A/D converter not used
			5.8		15	MHz	A/D converter used
			6		15	MHz	μPD78CP14 only
External clock	(Note 4)	X1 input frequency (f_X)	4		15	MHz	A/D converter not used
			5.8		15	MHz	A/D converter used
			6		15	MHz	μPD78CP14 only
		X1 input, rise, fall time (t_r, t_f)	0		20	ns	
		X_1 input low- and high-level width ($t_{\phi L}$, $t_{\phi H}$)	20		250	ns	
			20		167	ns	μPD78CP14

Notes:

(1) Refer to the Resonator and Capacitance Requirements table for the recommended ceramic resonators.

(2) For XTAL, the following external capacitances are recommended: C1 = C2 = 10 pF

(3) For XTAL, see the Recommended XTAL or Ceramic Resonator Oscillation Circuit Diagram.

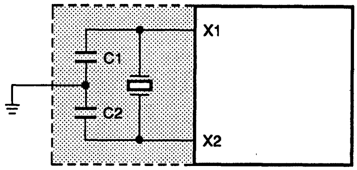
(4) See the following recommended external clock diagram.

When using an external crystal, it should be a parallel-resonant, fundamental mode, "AT cut" crystal. Capacitors C1 and C2 are required for frequency stability. The values of C1 and C2 (C1 = C2) can be calculated from the load capacitance (C_L), specified by the crystal manufacturer:

$$C_L = \frac{C1 \times C2}{C1 + C2} + C_S$$

Where C_S is any stray capacitance in parallel with the crystal such as the μPD78C10, μPD78C11, or μPD78C14 input capacitance between X1 and X2.

Recommended XTAL or Ceramic Resonator Oscillation Circuit Diagram

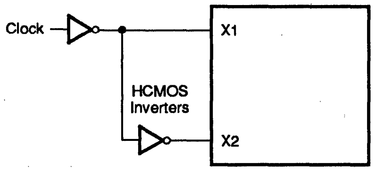


External oscillation circuit should be as close to the X1 and X2 pins as possible.

Do not place other signal lines in the shaded area.

83RD-6945A

Recommended External Clock Diagram



83RD-6946A

Resonator and Capacitance Requirements

T_A = −40 to +85°C

Manufacturer	Product Number	C1, C2 (pF)	Conditions
Murata	CSA15.0MX3	22	μPD78C10, 78C11, 78C14, 78C14A, 78CG14
	CSA10.0MT	30	
	CST10.0MT	Not required	
	CSA6.00MG	30	
	CST6.00MG	Not required	
	CSA12.0MT	30	Applies to all μPD78C1x/C1xA/CG14
	CST12.0MT	Not required	
	CSA15.00MX001	15	μPD78C10A/78C11A/78C12A
	CSA7.37MT	30	
	CST7.37MT	Not required	
TDK	FCR12.0MC	Not required	μPD78C10/78C11/78C14/ 78C14A/78CG14

DC Characteristics

T_A = −40° to +85°C; V_{DD} = +5.0 V ±10%; V_{DD} = +5.0 V ± 5% (μPD78C14 only); V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL1}	0		0.8	V	All except Note 1 inputs
	V_{IL2}	0		0.2 V_{DD}	V	Note 1 inputs
Input voltage, high	V_{IH1}	2.2		V_{DD}	V	All except X1, X2, and Note 1 inputs
	V_{IH2}	0.8 V_{DD}		V_{DD}	V	X1, X2, and Note 1 inputs
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD}−1.0			V	I_{OH} = 1.0 mA
		V_{DD}−0.5			V	I_{OH} = −100 μA
Data retention voltage	V_{DDDR}	2.5			V	STOP mode
Input current	I_{I1}			±200	μA	INT1 (Note 2); TI (PC_3) (Note 3); 0 V ≤ V_I ≤ V_{DD}
Input current (μPD78CG14 only)	I_{I2}			±200	μA	INT1 (Note 2); TI (PC_3) (Note 3); 0 V ≤ V_I ≤ V_{DD}
Input current (μPD78CG14 only)	I_{I3}			−300	μA	I_0–I_7 (upper input pin); V_I = 0
Input leakage current	I_{LI}			±10	μA	All except INT1, TI (PC_3), 0 V ≤ V_I ≤ V_{DD}
Output leakage current	I_{LO}			±10	μA	0 V ≤ V_O ≤ V_{DD}
AV_{DD} supply current	AI_{DD1}		0.5	1.3	mA	f = 15 MHz
	AI_{DD2}		10	20	μA	STOP mode
V_{DD} supply current	I_{DD1}		13	25	mA	Normal operation; f = 15 MHz; (μPD78C10/C10A/C11/C11A/C12A only)
	I_{DD2}		7	13	mA	HALT mode; f = 15 MHz; (μPD78C10/C10A/C11/C11A/C12A only)
	I_{DD3}		16	30	mA	Normal operation; f = 15 MHz (μPD78C14/C14A/CG14)
	I_{DD4}			32	mA	Normal operation; f = 15 MHz; (μPD78CP14 only)
	I_{DD5}		8	15	mA	HALT mode; f = 15 MHz; (μPD78C14/C14A/CG14/CP14 only)
Data retention current	I_{DDDR}		1	15	μA	V_{DDDR} = 2.5 V (Note 4)
				300		(μPD78CP14 only-Note 4)
			10	50	μA	V_{DDDR} = 5.0 V ±10% (Note 4)
				1	mA	(μPD78CP14 only-Note 4)
Pullup resistor	R_L	17	27	75	KΩ	Port A, B, C; 3.5 V ≤ V_{DD} ≤ 5.5 V; V_I = 0 V (μPD78C11A/C12A/C14A only)

Notes:

(1) Inputs $\overline{\text{RESET}}$, $\overline{\text{STOP}}$, $\overline{\text{NMI}}$, $\overline{\text{SCK}}$, INTP1, TI, and AN4-AN7.

(2) Assuming ZCM register is set to self-bias.

(3) Assuming ZCM register is set to self-bias and the MCC register is set to control mode.

(4) Hardware/software STOP mode and assuming ZCM register is set to self-bias not selected.

Serial Operation

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{CYK}	0.8		μs	$\overline{\text{SCK}}$ input (Notes 1, 3)
		0.4		μs	$\overline{\text{SCK}}$ input (Note 2)
		1.6		μs	$\overline{\text{SCK}}$ output (Note 3)
$\overline{\text{SCK}}$ width low	t_{KKL}	335		ns	$\overline{\text{SCK}}$ input (Notes 1, 3)
		160		ns	$\overline{\text{SCK}}$ input (Note 2)
		700		ns	$\overline{\text{SCK}}$ output (Note 3)
$\overline{\text{SCK}}$ width high	t_{KKH}	335		ns	$\overline{\text{SCK}}$ input (Notes 1, 3)
		160		ns	$\overline{\text{SCK}}$ input (Note 2)
		700		ns	$\overline{\text{SCK}}$ output (Note 3)
RxD setup time to $\overline{\text{SCK}}$ ↑	t_{RXK}	80		ns	(Note 1)
RxD hold time after $\overline{\text{SCK}}$ ↑	t_{KRX}	80		ns	(Note 1)
$\overline{\text{SCK}}$ ↓ TxD delay time	t_{KTX}		210	ns	(Note 1)

Notes:

(1) 1 x baud rate in synchronous or I/O interface mode.

(2) 16 x baud rate or 64 x baud rate in asynchronous mode.

(3) f_{XTAL} = 15 MHz.

Zero-Cross Characteristics

Parameter	Symbol	Min	Max	Unit	Condition
Zero-cross detection input	V_{ZX}	1	1.8	VAC_{p-p}	AC coupled 60 Hz sine wave
Zero-cross accuracy	A_{ZX}		±135	mV	
Zero-cross detection input frequency	f_{ZX}	0.05	1	kHz	

AC Characteristics (cont)

$T_A = -40°$ to $+85°C$; $V_{DD} = AV_{DD} = +5.0\,V \pm 10\%$ (±5% on μPD78CP14); $V_{SS} = 0\,V$

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{\text{RESET}}$ pulse width high, low	t_{RSH}, t_{RSL}	10		μs	
$\overline{\text{NMI}}$ pulse width high, low	t_{NIH}, t_{NIH}	10		μs	
X1 input cycle time	t_{CYC}	66	250	ns	
			167	ns	(Note 1)
Address setup to ALE ↓	t_{AL}	30		ns	(Notes 2, 3)
Address hold to ALE ↓	t_{LA}	35		ns	(Notes 2, 3)
Address to $\overline{\text{RD}}$ ↓ delay time	t_{AR}	100		ns	(Notes 2, 3)
$\overline{\text{RD}}$ ↓ to address floating	t_{AFR}		20	ns	(Note 2)
Address to data input	t_{AD}		250	ns	(Notes 2, 3)
ALE ↓ to data input	t_{LDR}		135	ns	(Notes 2, 3)
$\overline{\text{RD}}$ ↓ to data input	t_{RD}		120	ns	(Notes 2, 3)
ALE ↓ to $\overline{\text{RD}}$ ↓ delay time	t_{LR}	15		ns	(Notes 2, 3)
Data hold time $\overline{\text{RD}}$ ↑	t_{RDH}	0		ns	(Note 2)
$\overline{\text{RD}}$ ↑ to ALE ↑ delay time	t_{RL}	80		ns	(Notes 2, 3)
$\overline{\text{RD}}$ width low	t_{RR}	215		ns	Data read (Notes 2, 3)
		415		ns	Opcode fetch (Notes 2, 3)

AC Characteristics (cont)

Parameter	Symbol	Min	Max	Unit	Conditions
ALE width high	t_{LL}	90		ns	(Notes 2, 3)
$\overline{\text{M1}}$ setup time to ALE ↓	t_{ML}	30		ns	(Note 3)
$\overline{\text{M1}}$ hold time after ALE ↓	t_{LM}	35		ns	(Note 3)
$\overline{\text{IO}}$/M setup time to ALE ↓	t_{IL}	30		ns	(Note 3)
$\overline{\text{IO}}$/M hold time after ALE ↓	t_{LI}	35		ns	(Note 3)
Address to $\overline{\text{WR}}$ ↓ delay	t_{AW}	100		ns	(Notes 2, 3)
ALE ↓ to data output	t_{LDW}		180	ns	(Notes 2, 3)
$\overline{\text{WR}}$ ↓ to data output	t_{WD}		100	ns	(Note 2)
ALE ↓ to $\overline{\text{WR}}$ ↓ delay time	t_{LW}	15		ns	(Notes 2, 3)
Data setup time to $\overline{\text{WR}}$ ↑	t_{DW}	165		ns	(Notes 2, 3)
Data hold time to $\overline{\text{WR}}$ ↑	t_{WDH}	60		ns	(Notes 2, 3)
$\overline{\text{WR}}$ ↑ to ALE ↑ delay time	t_{WL}	80		ns	(Notes 2, 3)
$\overline{\text{WR}}$ width low	t_{WW}	215		ns	(Notes 2, 3)
Address to data input	t_{ACC}		250	ns	(Notes 2, 3)
Data hold time from address	t_{IH}	0		ns	(Note 2)

Notes:

(1) Applies to μPD78CP14 only.

(2) Load capacitance $C_L = 150$ pF.

(3) Values are for 15-MHz operation. For operation at other frequencies, refer to the table called Bus Timing Depending on t_{CYC}.

A/D Converter Characteristics

$T_A = -40°$ to $+85°C$; $V_{DD} = +5.0$ V ±10% (±5% on μPD78CP14); $V_{SS} = AV_{SS}$ 0 V;
$V_{DD} -0.5 \text{ V} \leq AV_{DD} \leq V_{DD}$; $3.4 \text{ V} \leq V_{AREF} \leq AV_{DD}$

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Resolution		8			bits	
Absolute accuracy (Note 1)				±0.4	%FSR	$T_A = -10$ to $+70°C$; $66 \text{ ns} \leq t_{CYC} \leq 170 \text{ ns}$; $4.0 \text{ V} \leq V_{AREF} \leq AV_{DD}$
				±0.6	%FSR	$66 \text{ ns} \leq t_{CYC} \leq 170 \text{ ns}$; $4.0 \text{ V} \leq V_{AREF} \leq AV_{DD}$
				±0.8	%FSR	$66 \text{ ns} \leq t_{CYC} \leq 170 \text{ ns}$; $3.4 \text{ V} \leq V_{AREF} \leq AV_{DD}$
Conversion time	t_{CONV}	576			t_{CYC}	$66 \text{ ns} \leq t_{CYC} \leq 110 \text{ ns}$
		432			t_{CYC}	$110 \text{ ns} \leq t_{CYC} \leq 170 \text{ ns}$
Sampling time	t_{SAMP}	96			t_{CYC}	$66 \text{ ns} \leq t_{CYC} \leq 110 \text{ ns}$
		72			t_{CYC}	$110 \text{ ns} \leq t_{CYC} \leq 170 \text{ ns}$
Analog input voltage	V_{IAN}	0		V_{AREF}	V	
Analog input impedance	R_{AN}		1000		MΩ	
Reference voltage	V_{AREF}	3.4		AV_{DD}	V	
V_{AREF} current	I_{AREF1}		1.5	3.0	mA	Operation mode
	I_{AREF2}		0.7	1.5	mA	STOP mode
AV_{DD} supply current	AI_{DD1}		0.5	1.3	mA	Operation mode
	AI_{DD2}		10	20	μA	STOP mode

Notes:

(1) Quantizing error (±1/2 LSB) is not included.

(2) FSR = Full-scale resolution.

Bus Timing Dependent on t_{CYK}

Symbol	Min/Max (ns)	Calculation Formula
t_{TIH}, t_{TIL}	Min	6T (TI input - PC_3)
t_{CI1H}, t_{CI1L} (Note 2)	Min	6T (TI input - PC_5)
t_{CI2H}, t_{CI2L} (Note 3)	Min	48T (TI input - PC_5)
t_{I1H}, t_{I1L}	Min	36T (INT1)
t_{I2H}, t_{I2L}	Min	36T ($\overline{INT2}$)
t_{ANH}, t_{ANL}	Min	36T (AN4-AN7)
t_{AL}	Min	2T – 100
t_{LA}	Min	T – 30
t_{AR}	Min	3T – 100
t_{AD}	Max	7T – 220
t_{LDR}	Max	5T – 200
t_{RD}	Max	4T – 150
t_{LR}	Min	T – 50
t_{RL}	Min	2T – 50
t_{RR}	Min	4T – 50 (Data read)
	Min	7T – 50 (Opcode fetch)
t_{LL}	Min	2T – 40
t_{ML}	Min	2T – 100
t_{LM}	Min	T – 30
t_{IL}	Min	2T – 100
t_{LI}	Min	T – 30
t_{AW}	Min	3T – 100
t_{LDW}	Max	T + 110
t_{LW}	Min	T – 50
t_{DW}	Min	4T – 100
t_{WDH}	Min	2T – 70
t_{WL}	Min	2T – 50
t_{WW}	Min	4T – 50
t_{CYK}	Min	12T ($\overline{SCK}$ input) (Note 1)
	Min	24T ($\overline{SCK}$ output)
t_{KKL}	Min	5T + 5 ($\overline{SCK}$ input) (Note 1)
	Min	12T – 100 ($\overline{SCK}$ output)
t_{KKH}	Min	5T + 5 ($\overline{SCK}$ input) (Note 1)
	Min	12T – 100 ($\overline{SCK}$ output)

Notes:

(1) 1 x baud rate in synchronous or I/O interface mode; T = t_{CYC} = $1/f_{XTAL}$.
The items not included in this list are independent of oscillator frequency (f_{XTAL}).

(2) Event counter mode.

(3) Pulse width measurement mode.

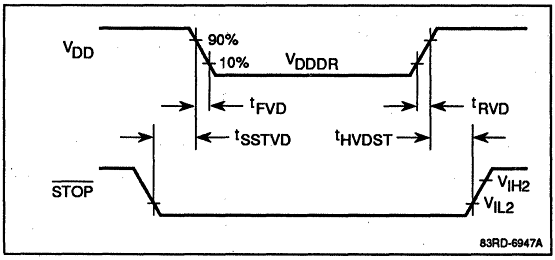
Data Memory STOP Mode Data Retention Characteristics

T_A = –40 to 85°C

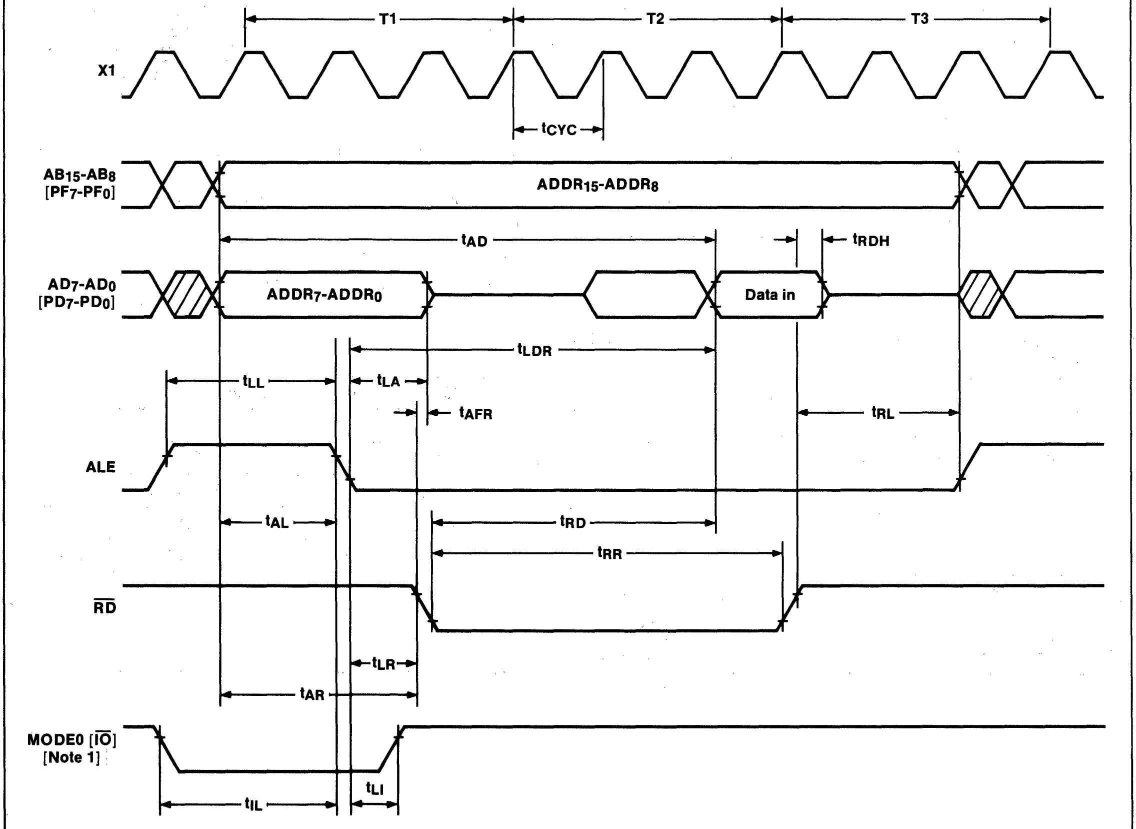
Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Data retention power supply voltage	V_{DDDR}	2.5		5.5	V	
Data retention power supply current	I_{DDDR}		1	15	µA	V_{DDDR} = 2.5 V
			15	50	µA	V_{DDDR} = 5.0 V ±10%
				300	µA	V_{DDDR} = 2.4 V (µPD78CP14)
				1	mA	V_{DDDR} = 5.0 V ±5% (µPD78CP14)
V_{DD} rise, fall time	t_{RVD}, t_{FVD}	200			µs	
$\overline{STOP}$ setup time to V_{DD}	t_{SSTVD}	12T+0.5			µs	
$\overline{STOP}$ hold time from V_{DD}	t_{HVDST}	12T+0.5			µs	

Timing Waveforms

Data Retention Timing



Read Operation

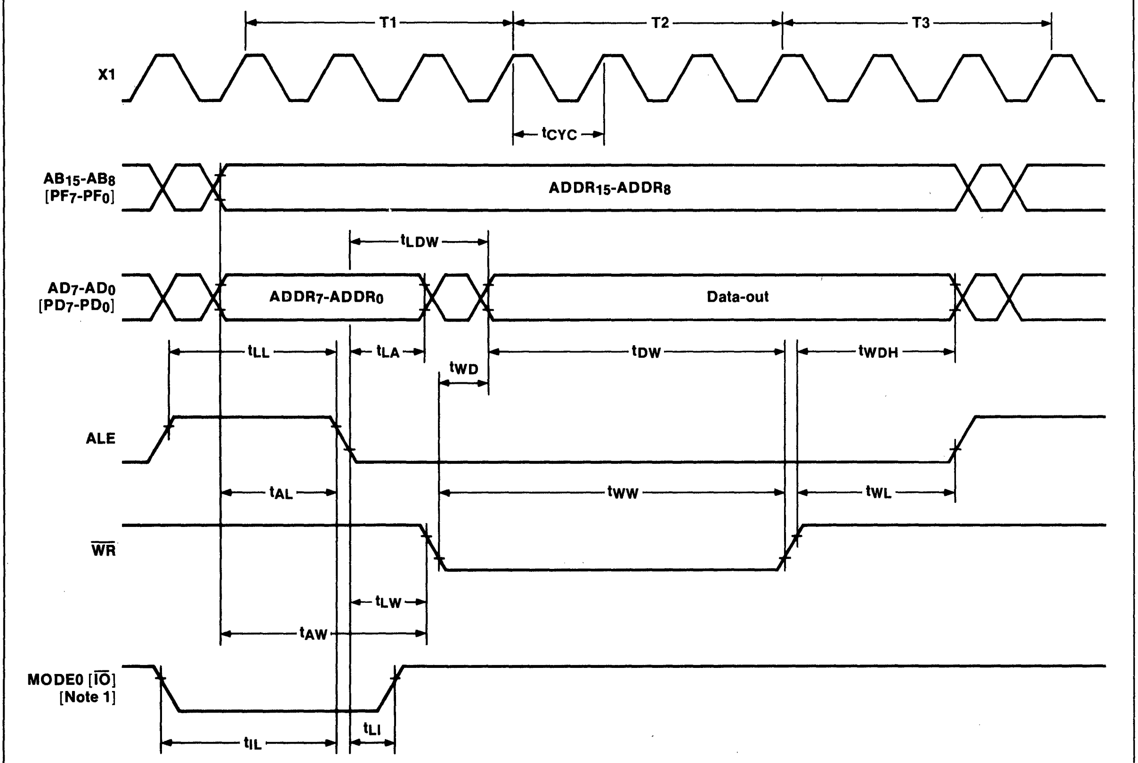


Note:

[1] $\overline{\text{IO}}$ signal is output to the MODE0 pin [if MODE0 is pulled up to V_{DD}] during a read or write of special registers sr-sr2 or a write to register MM or MF. Refer to description of Port Emulation Mode [PEM] in the User's Manual for further explanation. This signal is not output on the μPD78CP14.

83-004276B

Write Operation

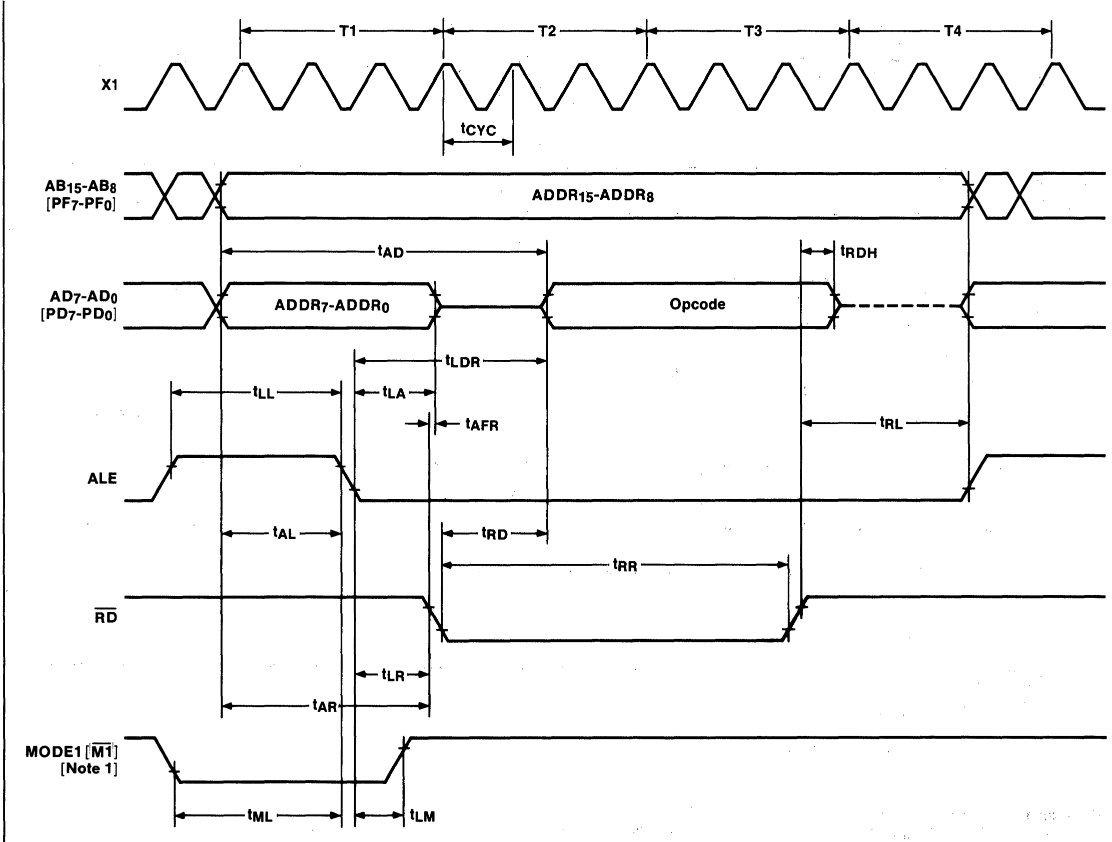


Note:

[1] $\overline{\text{IO}}$ signal is output to the MODE0 pin [if MODE0 is pulled up to V_{DD}] during a read or write of special registers sr-sr2 or a write to register MM or MF. Refer to description of Port Emulation Mode [PEM] in the User's Manual for further explanation. This signal is not output on the μPD78CP14.

83-004278B

Opcode Fetch Operation

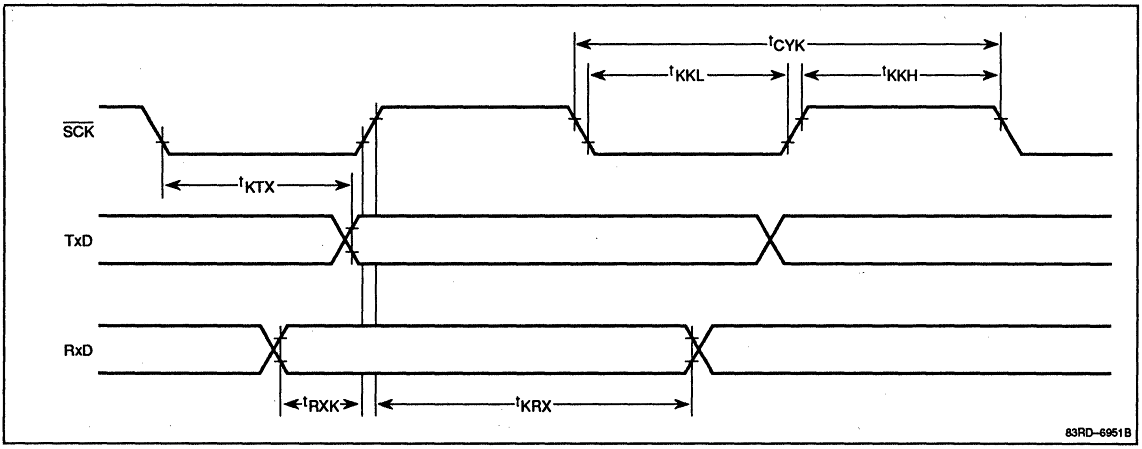


Note:

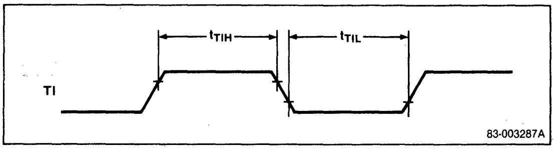
[1] $\overline{M1}$ signal is output to the MODE1 pin during every Opcode Fetch if MODE1 pin is pulled up to V_{DD}. This signal is not output on the μPD78CP14.

83-004279B

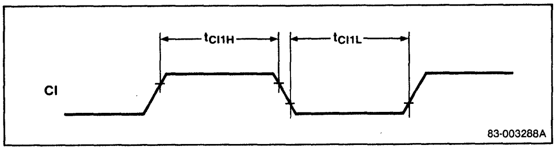
Serial Operation Transmit/Receive Timing



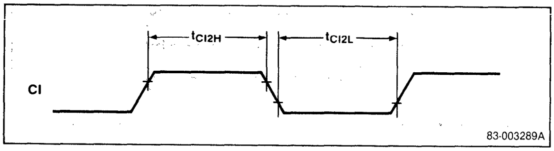
Timer Input Timing



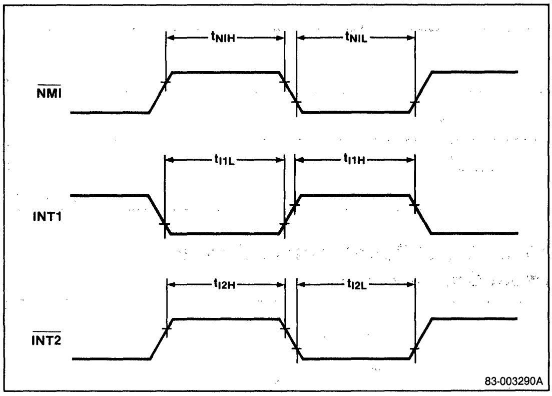
Timer/Event Counter Input Timing: Event Counter Mode



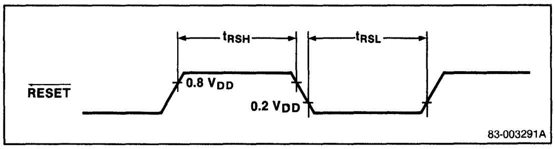
Timer/Event Counter Input Timing: Pulse Width Measurement Mode



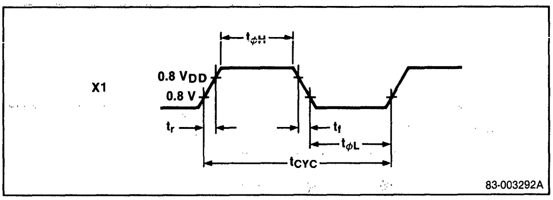
Interrupt Input Timing



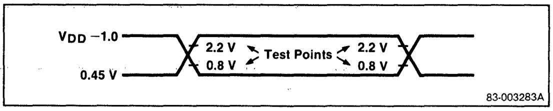
RESET Input Timing



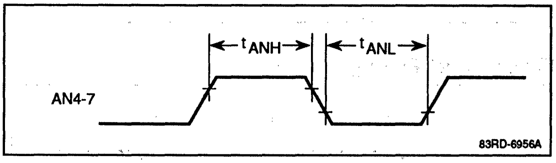
External Clock Timing



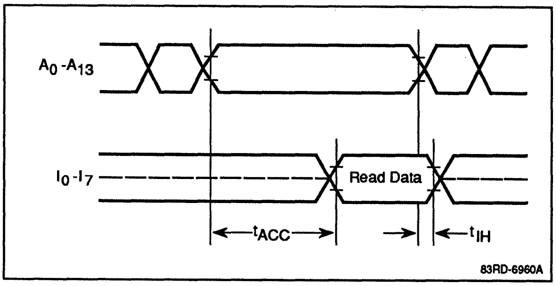
AC Timing Test Points



AN4-AN7 Edge Detection Timing



μPD78CG14E EPROM Read Timing



μPD78CP14 PROGRAMMING

In the μPD78CP14, the mask ROM of the μPD78C1X/C1XA is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 16,384 by 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode. Refer to tables 3 through 5 and the AC and DC Programming Characteristics for specific information applicable to programming the μPD78CP14.

The PA-78CP14CW/GF/GQ/L are the socket adapters used for configuring the μPD78CP14 to fit a standard μPD27C256A PROM socket.

Table 3. Pin Functions during EPROM Programming

Pin	Function	Description
PA_0-PA_7	A_0-A_7	Low-order 8-bit address
PF_0	A_8	High-order 7-bit address
$\overline{\text{NMI}}$	A_9	
PF_2-PF_6	A_{10}-A_{14}	
PD_0-PD_7	D_0-D_7	Data input/output
PB_6	$\overline{\text{CE}}$	Chip enable input
PB_7	$\overline{\text{OE}}$	Output enable input
$\overline{\text{RESET}}$	$\overline{\text{RESET}}$	PROM programming mode requires a low voltage on this pin
Mode 0	Mode 0	Enter PROM programming mode by applying a high voltage to this pin
Mode 1	Mode 1	Enter PROM programming mode by applying a low voltage to this pin
$\overline{\text{STOP}}$	V_{PP}	High-voltage input (write/verify) high level (read)

Table 4. Summary of Operation Modes for EPROM Programming

Operation Mode	$\overline{\text{CE}}$	$\overline{\text{OE}}$	V_{PP}	V_{DD}	$\overline{\text{RESET}}$	MODE0	MODE1	A_{14}
Program write	L	H	+12.5 V	+6 V	L	H	L	L
Program verify	H	L	+12.5 V	+6 V	L	H	L	L
Program inhibit	H	H	+12.5 V	+6 V	L	H	L	L
Read	L	L	+5 V	+5 V	L	H	L	L
Output disable	L	H	+5 V	+5 V	L	H	L	L
Standby	H	L/H	+5 V	+5 V	L	H	L	L

Notes:

(1) The $\overline{\text{CE}}$, $\overline{\text{OE}}$, V_{pp}, and V_{DD} pins are all compatible with the μPD27C256A pins.

Caution: When Vpp is set to +12.5 V and V_{DD} is set to +6 V, you cannot set both $\overline{\text{CE}}$ and $\overline{\text{OE}}$ to low level (L).

Table 5. Recommended Connections for Unused Pins (EPROM Programming Mode)

Pin	Recommended Connection Method
INT1	Connect to V_{SS}
X1	Connect to V_{SS}
X2	Leave this pin disconnected
AN0-AN7	Connect to V_{SS}
VA_{REF}	Connect to V_{SS}
AV_{DD}	Connect to V_{SS}
AV_{SS}	Connect to V_{SS}
Remaining pins	Connect each pin via a resistor to V_{SS}

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin, the MODE1 pin, and A_{14} pin to a low level and connect the MODE0 pin to a high level. Connect all unused pins as recommended in Table 5.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{pp} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This bit is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps 4 to 6.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) FIX the $\overline{\text{RESET}}$ pin, the MODE1 pin, and A_{14} pin to a low level and connect the MODE0 pin to a high level.

(2) Apply +5 V to the V_{DD} and V_{pp} pins.

(3) Input the address of the data to be read to pins A_0-A_{14}.

(4) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(5) Data is output to the D_0-D_7 pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15W-s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

μPD78CP14 DC Programming Characteristics

T_A = 25 ±5°C; MODE1 = V_{IL}; MODE0 = V_{IH}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.2		$V_{DDP}+0.3$	V	
Low-level input voltage	V_{IL}	V_{IL}	−0.3		0.8	V	
Input leakage current	I_{LIP}	I_{LI}			±10	μA	$0 \leq V_1 \leq V_{DDP}$
High-level output voltage	V_{OH}	V_{OH}	$V_{DD} - 1.0$			V	$I_{OH} = -1.0$ mA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	$I_{OL} = 2.0$ mA
Output leakage current	I_{LO}				±10	μA	$0 \leq V_O \leq V_{DDP}$; $\overline{OE} = V_{IH}$
V_{DDP} power voltage	V_{DDP}	V_{CC}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				$V_{PP}=V_{DDP}$		V	Program memory read mode
V_{DDP} power current	I_{DD}	I_{CC}			30	mA	Program memory write mode
					30	mA	Program memory read mode; $\overline{CE} = V_{IL}$; $V_I = V_{IH}$
V_{PP} power current	I_{PP}	I_{PP}			30	mA	Program memory read mode; $\overline{CE} = V_{IL}$; $\overline{OE} = V_{IH}$
				1	100	μA	Program memory write mode

* Corresponding symbols of the μPD27C256A.

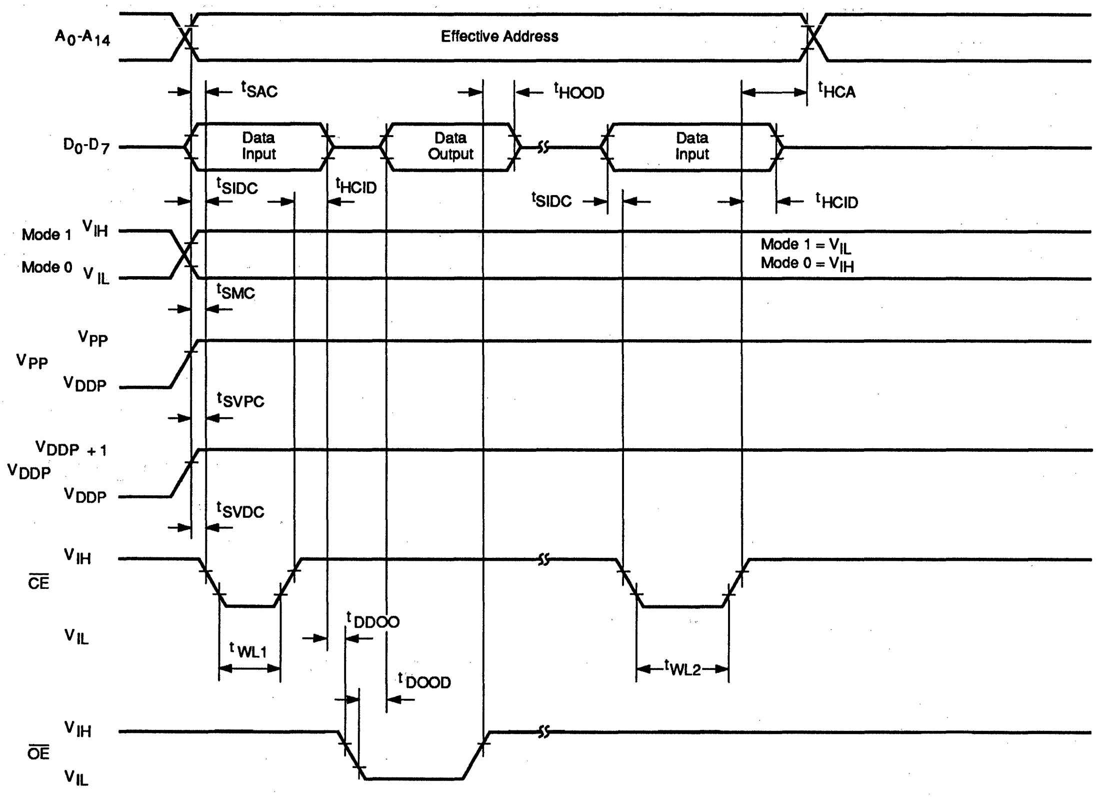
μPD78CP14 AC Programming Characteristics

T_A = 25 ±5°C; MODE1 = V_{IL}; V_{SS} = 0 V

Parameter	Symbol	Symbol*	Min	Typ	Max	Unit	Condition
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	2			μs	
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}	2			μs	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
MODE0/MODE1 setup time vs. $\overline{CE}$ ↓	t_{SMC}		2			μs	MODE1 = V_{IL} and MODE0 = V_{IH}
Address to data output time	t_{DAOD}	t_{ACC}			2	μs	$\overline{OE} = V_{IL}$
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}			1	μs	
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			1	μs	
Data hold time from $\overline{OE}$ ↑ or $\overline{CE}$ ↑	t_{HCOD}	t_{DF}	0		130	ns	
Data hold time from address	t_{HAOD}	t_{OH}	0			ns	$\overline{OE} = V_{IL}$

* Corresponding symbols of the μPD27C256A.

μPD78CP14 PROM Write Mode Timing



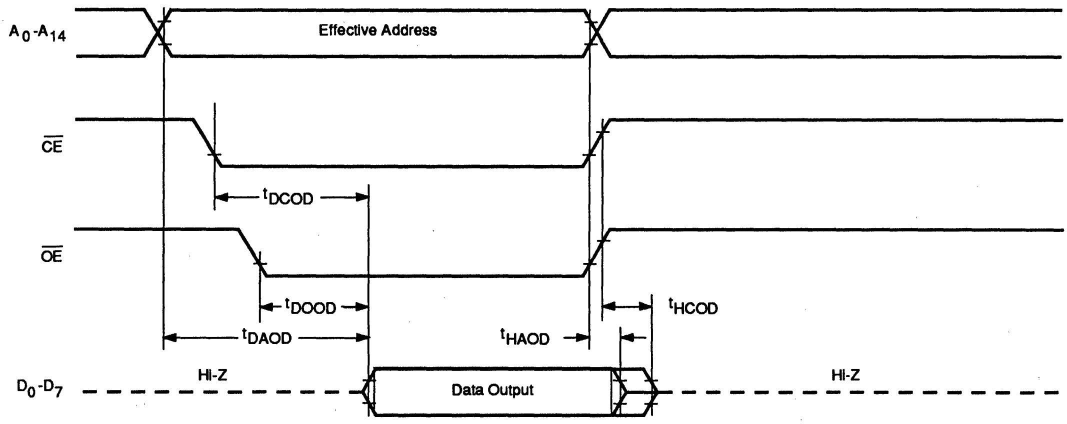
Notes:

(1) V_{DDP} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.

(2) V_{PP} must not exceed +13 V, including overshoot.

83RD-6948B

μPD78CP14 PROM Read Mode Timing



Notes:

(1) To read PROM within the t_{DAOD} range, the delay of $\overline{OE}$↓ from $\overline{CE}$↓ must be within t_{DAOD} - t_{DOOD}.

(2) t_{HCOD} is the time from the state in which either $\overline{OE}$ or $\overline{CE}$ first becomes V_{IH}.

83RD-6949B

Operand Symbols

Symbol	Allowable Operands
Registers	
r	V, A, B, C, D, E, H, L
r1	EAH, EAL, B, C, D, E, H, L
r2	A, B, C
Special Registers	
sr	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, SML, EOM, ETMM, TMM, MM, MCC, MA, MB, MC, MF, TXB, TM0, TM1, ZCM
sr1	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, EOM, TMM, RXB, CR0, CR1, CR2, CR3
sr2	PA, PB, PC, PD, PF, MKH, ANM, MKL, SMH, EOM, TMM
sr3	ETM0, ETM1
sr4	ECNT, ECPT
Register Pairs	
rp	SP, B, D, H
rp1	V, B, D, H, EA
rp2	SP, B, D, H, EA
rp3	B, D, H
Register Pair Addressing	
rpa	B, D, H, D+ , H+ , D–, H–
rpa1	B, D, H
rpa2	B, D, H, D+, H+, D–, H–, D+byte, H+A, H+B, H+EA, H+byte
rpa3	D, H, D++, H++, D+byte, H+A, H+B, H+EA, H+byte
Flags	
f	CY, HC, Z
Interrupt Flags	
irf	INTFNMI, INTFT0, INTFT1, INTF1, INTF2, INTFE0, INTFE1, INTFEIN, INTFAD, INTFSR, INTFST, ER, OV, AN4, AN5, AN6, AN7, SB
Immediate Data	
wa	8-bit immediate data (low byte of working register address)
word	16-bit immediate data
byte	8-bit immediate data
bit	3-bit immediate data (b_2, b_1, b_0)

Operand Definitions

Special Registers (sr-sr4)

PA = Port A
PB = Port B
PC = Port C
PD = Port D
PF = Port F
MA = Mode A
MB = Mode B
MC = Mode C
MCC = Mode control C
MF = Mode F
MM = Memory mapping
TM0 = Timer register 0
TM1 = Timer register 1
TMM = Timing mode
ETM0 = Timer/event counter register 0
ETM1 = Timer/event counter register 1
ZCM = Zero-cross mode control register

ECNT = Timer/event counter upcounter
ECPT = Timer/event counter capture
ETMM = Timer/event counter mode
EOM = Timer/event counter output mode
TXB = Transmit buffer
RXB = Receive buffer
SMH = Serial mode high
SML = Serial mode low
MKH = Mask high
MKL = Mask low
ANM = A/D channel mode
CR0 to CR3= A/D conversion result 0-3

Register Pairs (rp-rp3)

SP = Stack pointer
B = BC
D = DE

H = HL
V = VA
EA = Extended accumulator

Register Pair Addressing (rpa-rpa3)

B = (BC)
D = (DE)
H = (HL)
D+ = (DE)+
H+ = (HL)+
D– = (DE)–
H– = (HL)–

D++ = (DE)++
H++ = (HL)++
D+byte = (DE+byte)
H+byte = (HL+byte)
H+A = (HL+A)
H+B = (HL+B)
H+EA = (HL+EA)

Flags (f)

CY = Carry HC = Half-carry Z = Zero

Interrupt Flags (irf)

INTFNMI = NMI interrupt flag
INTFT0 = FT0
INTFT1 = FT1
INTF1 = F1
INTF2 = F2
INTFE0 = FE0
INTFE1 = FE1

INTFEIN = FEIN
INTFAD = FAD
INTFSR = FSR
INTFST = FST
ER = Error
OV = Overflow
AN4 to AN7 = Analog input 4-7
SB = Standby

Operand Codes

Registers (r, r2)

R_2	R_1	R_0	Reg	r	r2
0	0	0	V	│	
0	0	1	A	│	│
0	1	0	B	│	│
0	1	1	C	│	│
1	0	0	D	│	
1	0	1	E	│	
1	1	0	H	│	
1	1	1	L	│	

Registers (r1)

T_2	T_1	T_0	Reg
0	0	0	EAH
0	0	1	EAL
0	1	0	B
0	1	1	C
1	0	0	D
1	0	1	E
1	1	0	H
1	1	1	L

Special Registers (sr, sr1, sr2)

S_5	S_4	S_3	S_2	S_1	S_0	Special Reg	sr	sr1	sr2
0	0	0	0	0	0	PA	│	│	│
0	0	0	0	0	1	PB	│	│	│
0	0	0	0	1	0	PC	│	│	│
0	0	0	0	1	1	PD	│	│	│
0	0	0	1	0	1	PF	│	│	│
0	0	0	1	1	0	MKH	│	│	│
0	0	0	1	1	1	MKL	│	│	│
0	0	1	0	0	0	ANM	│	│	│
0	0	1	0	0	1	SMH	│	│	│
0	0	1	0	1	0	SML	│		
0	0	1	0	1	1	EOM	│	│	│
0	0	1	1	0	0	ETMM	│		
0	0	1	1	0	1	TMM	│	│	│
0	1	0	0	0	0	MM	│		
0	1	0	0	0	1	MCC	│		
0	1	0	0	1	0	MA	│		
0	1	0	0	1	1	MB	│		
0	1	0	1	0	0	MC	│		
0	1	0	1	1	1	MF	│		
0	1	1	0	0	0	TXB	│		
0	1	1	0	0	1	RXB		│	
0	1	1	0	1	0	TM0	│		
0	1	1	0	1	1	TM1	│		
1	0	0	0	0	0	CR0		│	
1	0	0	0	0	1	CR1		│	
1	0	0	0	1	0	CR2		│	
1	0	0	0	1	1	CR3		│	
1	0	1	0	0	0	ZCM	│		

Special Registers (sr3)

U_0	Special Reg
0	ETM0
1	ETM1

Special Registers (sr4)

V_0	Special Reg
0	ECNT
1	ECPT

Register Pairs (rp, rp2, rp3)

P_2	P_1	P_0	Reg Pair	rp	rp2	rp3
0	0	0	SP	│	│	
0	0	1	BC	│	│	│
0	1	0	DE	│	│	│
0	1	1	HL	│	│	│
1	0	0	EA		│	

Register Pairs (rp1)

Q_2	Q_1	Q_0	Reg Pair
0	0	0	VA
0	0	1	BC
0	1	0	DE
0	1	1	HL
1	0	0	EA

Register Pair Addressing (rpa, rpa1, rpa2)

A_3	A_2	A_1	A_0	Addressing	rpa	rpa1	rpa2
0	0	0	0	—	│	│	│
0	0	0	1	(BC)	│	│	│
0	0	1	0	(DE)	│	│	│
0	0	1	1	(HL)	│	│	│
0	1	0	0	(DE)+	│		│
0	1	0	1	(HL)+	│		│
0	1	1	0	(DE)−	│		│
0	1	1	1	(HL)−	│		│
1	0	1	1	(DE+byte)			│
1	1	0	0	(HL+A)			│
1	1	0	1	(HL+B)			│
1	1	1	0	(HL+EA)			│
1	1	1	1	(HL+byte)			│

Register Pair Addressing (rpa3)

C_3	C_2	C_1	C_0	Addressing
0	0	1	0	(DE)
0	0	1	1	(HL)
0	1	0	0	(DE)++
0	1	0	1	(HL)++
1	0	1	1	(DE+byte)
1	1	0	0	(HL+A)
1	1	0	1	(HL+B)
1	1	1	0	(HL+EA)
1	1	1	1	(HL+byte)

Operand Codes (cont)

Flags (f)

F_2	F_1	F_0	Flag
0	0	0	—
0	1	0	CY
0	1	1	HC
1	0	0	Z

Interrupt Flags (irf)

I_4	I_3	I_2	I_1	I_0	Flag
0	0	0	0	0	NMI
0	0	0	0	1	FT0
0	0	0	1	0	FT1
0	0	0	1	1	F1
0	0	1	0	0	F2
0	0	1	0	1	FE0
0	0	1	1	0	FE1
0	0	1	1	1	FEIN
0	1	0	0	0	FAD
0	1	0	0	1	FSR
0	1	0	1	0	FST
0	1	0	1	1	ER
0	1	1	0	0	OV
1	0	0	0	0	AN4
1	0	0	0	1	AN5
1	0	0	1	0	AN6
1	0	0	1	1	AN7
1	0	1	0	0	SB

Graphic Symbols

Symbol	Description
←	Transfer direction, result
∧	Logical product (logical AND)
∨	Logical sum (logical OR)
⊻	Exclusive-OR
—	Complement
•	Concatenation

Instruction Set

			Operation Code																		
			B1								B2										
			B3								B4										
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	State (Note 1)	Bytes	Skip Condition
8-Bit Data Transfer																					
MOV	r1,A	(r1) ← (A)	0	0	0	1	1	T_2	T_1	T_0									4	1	
	A, r1	(A) ← (r1)	0	0	0	0	1	T_2	T_1	T_0									4	1	
	*sr,A	(sr) ← (A)	0	1	0	0	1	1	0	1	1	1	S_5	S_4	S_3	S_2	S_1	S_0	10	2	
	*A,sr1	(A) ← (sr1)	0	1	0	0	1	1	0	0	1	1	S_5	S_4	S_3	S_2	S_1	S_0	10	2	
	r,word	(r) ← (word)	0	1	1	1	0	0	0	0	0	1	1	0	1	R_2	R_1	R_0	17	4	
			Low addr								High addr										
	word,r	(word) ← (r)	0	1	1	1	0	0	0	0	0	1	1	1	1	R_2	R_1	R_0	17	4	
			Low addr								High addr										
MVI	*r,byte	(r) ← byte	0	1	1	0	1	R_2	R_1	R_0	Data								7	2	
	sr2,byte	(sr2) ← byte	0	1	1	0	0	1	0	0	S_3	0	0	0	0	S_2	S_1	S_0	14	3	
			Data																		
MVIW	*wa, byte	((V)•(wa)) ← byte	0	1	1	1	0	0	0	1	Offset								13	3	
			Data																		
MVIX	*rpa1,byte	(rpa1) ← byte	0	1	0	0	1	0	A_1	A_0	Data								10	2	
STAW	*wa	((V)•(wa)) ← (A)	0	1	1	0	0	0	1	1	Offset								10	2	
LDAW	*wa	(A) ← ((V)•(wa))	0	0	0	0	0	0	0	1	Offset								10	2	
STAX	*rpa2	((rpa2)) ← (A)	A_3	0	1	1	1	A_2	A_1	A_0	Data (Note 2)								7/13 (Note 3)	2	
LDAX	*rpa2	(A) ← ((rpa2))	A_3	0	1	0	1	A_2	A_1	A_0	Data (Note 2)								7/13 (Note 3)	2	
EXX		(B) ↔ (B'), (C) ↔ (C'), (D) ↔ (D') (E) ↔ (E'), (H) ↔ (H'), (L) ↔ (L')	0	0	0	1	0	0	0	1									4	1	
EXA		(V) ↔ (V'), (A) ↔ (A'), (EA) ↔ (EA')	0	0	0	1	0	0	0	0									4	1	
EXH		(H) ↔ (H'), (L) ↔ (L')	0	1	0	1	0	0	0	0									4	1	
BLOCK		((DE)) ← ((HL)), (DE) ← (DE) + 1, (HL) ← (HL) + 1, (C) ← (C) – 1 End if borrow	0	0	1	1	0	0	0	1									13 x (C + 1)	1	
16-Bit Data Transfer																					
DMOV	rp3, EA	($rp3_L$) ← (EAL), ($rp3_H$) ← (EAH)	1	0	1	1	0	1	P_1	P_0									4	1	
	EA,rp3	(EAL) ← ($rp3_L$), (EAH) ← ($rp3_H$)	1	0	1	0	0	1	P_1	P_0									4	1	

Notes:

(1) For the skip condition, the idle states are as follows:

1-byte instruction: 4 states	2-byte instruction (with *): 7 states
2-byte instruction: 8 states	3-byte instruction (with *): 10 states
3-byte instruction: 11 states	4-byte instruction: 14 states

(2) B2 (Data): rpa2 = D+byte or H+byte.

(3) Right side of slash (/) in states indicates case rpa2 or rpa3 = D+byte, H+A, H+B, H+EA, or H+byte.

(4) B3 (Data): rpa3 = D+byte or H+byte.

Instruction Set (cont)

			Operation Code																		
			B1								B2										
			B3								B4								State		Skip
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	(Note 1)	Bytes	Condition
16-Bit Data Transfer (cont)																					
DMOV	sr3, EA	(sr3) ← (EA)	0	1	0	0	1	0	0	0	1	1	0	1	0	0	1	U_0	14	2	
	EA,sr4	(EA) ← (sr4)	0	1	0	0	1	0	0	0	1	1	0	0	0	0	0	V_0	14	2	
SBCD	word	(word) ← (C), (word + 1) ← (B)	0	1	1	1	0	0	0	0	0	0	0	1	1	1	1	0	20	4	
			Low addr								High addr										
SDED	word	(word) ← (E), (word + 1) ← (D)	0	1	1	1	0	0	0	0	0	0	1	0	1	1	1	0	20	4	
			Low addr								High addr										
SHLD	word	(word) ← (L), (word + 1) ← (H)	0	1	1	1	0	0	0	0	0	0	1	1	1	1	1	0	20	4	
			Low addr								High addr										
SSPD	word	(word) ← (SP_L), (word + 1) ← (SP_H)	0	1	1	1	0	0	0	0	0	0	0	0	1	1	1	0	20	4	
			Low addr								High addr										
STEAX	rpa3	((rpa3)) ← (EAL), (((rpa3)) + 1)) ← (EAH)	0	1	0	0	1	0	0	0	1	0	0	1	C_3	C_2	C_1	C_0	14/20 (Note 3)	3	
			Data (Note 4)																		
LBCD	word	(C) ← (word), (B) ← (word + 1)	0	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	20	4	
			Low addr								High addr										
LDED	word	(E) ← (word), (D) ← (word + 1)	0	1	1	1	0	0	0	0	0	0	1	0	1	1	1	1	20	4	
			Low addr								High addr										
LHLD	word	(L) ← (word), (H) ← (word + 1)	0	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	20	4	
			Low addr								High addr										
LSPD	word	(SP_L) ← (word), (SP_H) ← (word + 1)	0	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	20	4	
			Low addr								High addr										
LDEAX	rpa3	(EAL) ← ((rpa3)), (EAH) ← (((rpa3) + 1))	0	1	0	0	1	0	0	0	1	0	0	0	C_3	C_2	C_1	C_0	14/20 (Note 3)	3	
			Data (Note 4)																		
PUSH	rp1	(((SP) − 1)) ← ($rp1_H$), (((SP) − 2)) ← ($rp1_L$), (SP) ← (SP) − 2	1	0	1	1	0	Q_2	Q_1	Q_0									13	1	
POP	rp1	($rp1_L$) ← ((SP)), ($rp1_H$) ← (((SP) + 1)), (SP) ← (SP) + 2	1	0	1	0	0	Q_2	Q_1	Q_0									10	1	
LXI	*rp2,word	(rp2) ← (word)	0	P_2	P_1	P_0	0	1	0	0	Low byte								10	3	
			High byte																		
TABLE		(C) ← (((PC) + 3 + (A))), (B) ← (((PC) + 3 + (A) + 1))	0	1	0	0	1	0	0	0	1	0	1	0	1	0	0	0	17	2	
8-Bit Arithmetic (Register)																					
ADD	A,r	(A) ← (A) + (r)	0	1	1	0	0	0	0	0	1	1	0	0	0	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) + (A)	0	1	1	0	0	0	0	0	0	1	0	0	0	R_2	R_1	R_0	8	2	
ADC	A,r	(A) ← (A) + (r) + (CY)	0	1	1	0	0	0	0	0	1	1	0	1	0	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) + (A) + (CY)	0	1	1	0	0	0	0	0	0	1	0	1	0	R_2	R_1	R_0	8	2	

Instruction Set (cont)

			Operation Code																		
						B1								B2							
						B3								B4							
Mnemonic	**Operand**	**Operation**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**State (Note 1)**	**Bytes**	**Skip Condition**
8-Bit Arithmetic [Register] (cont)																					
ADDNC	A,r	(A) ← (A) + (r)	0	1	1	0	0	0	0	0	1	0	1	0	0	R_2	R_1	R_0	8	2	No carry
	r,A	(r) ← (r) + (A)	0	1	1	0	0	0	0	0	0	0	1	0	0	R_2	R_1	R_0	8	2	No carry
SUB	A,r	(A) ← (A) – (r)	0	1	1	0	0	0	0	0	1	1	1	0	0	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) – (A)	0	1	1	0	0	0	0	0	0	1	1	0	0	R_2	R_1	R_0	8	2	
SBB	A,r	(A) ← (A) – (r) – (CY)	0	1	1	0	0	0	0	0	1	1	1	1	0	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) – (A) – (CY)	0	1	1	0	0	0	0	0	0	1	1	1	0	R_2	R_1	R_0	8	2	
SUBNB	A,r	(A) ← (A) – (r)	0	1	1	0	0	0	0	0	1	0	1	1	0	R_2	R_1	R_0	8	2	No borrow
	r,A	(r) ← (r) – (A)	0	1	1	0	0	0	0	0	0	0	1	1	0	R_2	R_1	R_0	8	2	No borrow
ANA	A,r	(A) ← (A) ∧ (r)	0	1	1	0	0	0	0	0	1	0	0	0	1	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) ∧ (A)	0	1	1	0	0	0	0	0	0	0	0	0	1	R_2	R_1	R_0	8	2	
ORA	A,r	(A) ← (A) V (r)	0	1	1	0	0	0	0	0	1	0	0	1	1	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) V (A)	0	1	1	0	0	0	0	0	0	0	0	1	1	R_2	R_1	R_0	8	2	
XRA	A,r	(A) ← (A) ∀ (r)	0	1	1	0	0	0	0	0	1	0	0	1	0	R_2	R_1	R_0	8	2	
	r,A	(r) ← (r) ∀ (A)	0	1	1	0	0	0	0	0	0	0	0	1	0	R_2	R_1	R_0	8	2	
GTA	A,r	(A) – (r) – 1	0	1	1	0	0	0	0	0	1	0	1	0	1	R_2	R_1	R_0	8	2	No borrow
	r,A	(r) – (A) – 1	0	1	1	0	0	0	0	0	0	0	1	0	1	R_2	R_1	R_0	8	2	No borrow
LTA	A,r	(A) – (r)	0	1	1	0	0	0	0	0	1	0	1	1	1	R_2	R_1	R_0	8	2	Borrow
	r,A	(r) – (A)	0	1	1	0	0	0	0	0	0	0	1	1	1	R_2	R_1	R_0	8	2	Borrow
NEA	A,r	(A) – (r)	0	1	1	0	0	0	0	0	1	1	1	0	1	R_2	R_1	R_0	8	2	No zero
	r,A	(r) – (A)	0	1	1	0	0	0	0	0	0	1	1	0	1	R_2	R_1	R_0	8	2	No zero
EQA	A,r	(A) – (r)	0	1	1	0	0	0	0	0	1	1	1	1	1	R_2	R_1	R_0	8	2	Zero
	r,A	(r) – (A)	0	1	1	0	0	0	0	0	0	1	1	1	1	R_2	R_1	R_0	8	2	Zero
ONA	A,r	(A) ∧ (r)	0	1	1	0	0	0	0	0	1	1	0	0	1	R_2	R_1	R_0	8	2	No zero
OFFA	A,r	(A) ∧ (r)	0	1	1	0	0	0	0	0	1	1	0	1	1	R_2	R_1	R_0	8	2	Zero
8-Bit Arithmetic (Memory)																					
ADDX	rpa	(A) ← (A) + ((rpa))	0	1	1	1	0	0	0	0	1	1	0	0	0	A_2	A_1	A_0	11	2	
ADCX	rpa	(A) ← (A) + ((rpa)) + (CY)	0	1	1	1	0	0	0	0	1	1	0	1	0	A_2	A_1	A_0	11	2	
ADDNCX	rpa	(A) ← (A) + ((rpa))	0	1	1	1	0	0	0	0	1	0	1	0	0	A_2	A_1	A_0	11	2	No carry
SUBX	rpa	(A) ← (A) – ((rpa))	0	1	1	1	0	0	0	0	1	1	1	0	0	A_2	A_1	A_0	11	2	
SBBX	rpa	(A) ← (A) – ((rpa)) – (CY)	0	1	1	1	0	0	0	0	1	1	1	1	0	A_2	A_1	A_0	11	2	
SUBNBX	rpa	(A) ← (A) – ((rpa))	0	1	1	1	0	0	0	0	1	0	1	1	0	A_2	A_1	A_0	11	2	No borrow
ANAX	rpa	(A) ← (A) ∧ ((rpa))	0	1	1	1	0	0	0	0	1	0	0	0	1	A_2	A_1	A_0	11	2	
ORAX	rpa	(A) ← (A) V ((rpa))	0	1	1	1	0	0	0	0	1	0	0	1	1	A_2	A_1	A_0	11	2	

Instruction Set (cont)

			Operation Code																		
			B1								B2										
			B3								B4								State		Skip
Mnemonic	**Operand**	**Operation**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**(Note 1)**	**Bytes**	**Condition**
8-Bit Arithmetic (Memory) (cont)																					
XRAX	rpa	(A) ← (A)⊻((rpa))	0	1	1	1	0	0	0	0	1	0	0	1	0	A_2	A_1	A_0	11	2	
GTAX	rpa	(A) – ((rpa)) – 1	0	1	1	1	0	0	0	0	1	0	1	0	1	A_2	A_1	A_0	11	2	No borrow
LTAX	rpa	(A) – ((rpa))	0	1	1	1	0	0	0	0	1	0	1	1	1	A_2	A_1	A_0	11	2	Borrow
NEAX	rpa	(A) – ((rpa))	0	1	1	1	0	0	0	0	1	1	1	0	1	A_2	A_1	A_0	11	2	No zero
EQAX	rpa	(A) – ((rpa))	0	1	1	1	0	0	0	0	1	1	1	1	1	A_2	A_1	A_0	11	2	Zero
ONAX	rpa	(A) ∧ ((rpa))	0	1	1	1	0	0	0	0	1	1	0	0	1	A_2	A_1	A_0	11	2	No zero
OFFAX	rpa	(A) ∧ ((rpa))	0	1	1	1	0	0	0	0	1	1	0	1	1	A_2	A_1	A_0	11	2	Zero
Immediate Data																					
ADI	*A,byte	(A) ← (A) + byte	0	1	0	0	0	1	1	0	Data								7	2	
	r,byte	(r) ← (r) + byte	0	1	1	1	0	1	0	0	0	1	0	0	0	R_2	R_1	R_0	11	3	
			Data																		
	sr2, byte	(sr2) ← (sr2) + byte	0	1	1	0	0	1	0	0	S_3	1	0	0	0	S_2	S_1	S_0	20	3	
			Data																		
ACI	*A,byte	(A) ← (A) + byte + (CY)	0	1	0	1	0	1	1	0	Data								7	2	
	r,byte	(r) ← (r) + byte + (CY)	0	1	1	1	0	1	0	0	0	1	0	1	0	R_2	R_1	R_0	11	3	
			Data																		
	sr2,byte	(sr2) ← (sr2) + byte + (CY)	0	1	1	0	0	1	0	0	S_3	1	0	1	0	S_2	S_1	S_0	20	3	
			Data																		
ADINC	*A,byte	(A) ← (A) + byte	0	0	1	0	0	1	1	0	Data								7	2	No carry
	r,byte	(r) ← (r) + byte	0	1	1	1	0	1	0	0	0	0	1	0	0	R_2	R_1	R_0	11	3	No carry
			Data																		
	sr2,byte	(sr2) ← (sr2) + byte	0	1	1	0	0	1	0	0	S_3	0	1	0	0	S_2	S_1	S_0	20	3	No carry
			Data																		
SUI	*A,byte	(A) ← (A) – byte	0	1	1	0	0	1	1	0	Data								7	2	
	r,byte	(r) ← (r) – byte	0	1	1	1	0	1	0	0	0	1	1	0	0	R_2	R_1	R_0	11	3	
			Data																		
	sr2,byte	(sr2) ← (sr2) – byte	0	1	1	0	0	1	0	0	S_3	1	1	0	0	S_2	S_1	S_0	20	3	
			Data																		
SBI	*A,byte	(A) ← (A) – byte – (CY)	0	1	1	1	0	1	1	0	Data								7	2	
	r,byte	(r) ← (r) – byte – (CY)	0	1	1	1	0	1	0	0	0	1	1	1	0	R_2	R_1	R_0	11	3	
			Data																		
	sr2,byte	(sr2) ← (sr2) – byte – (CY)	0	1	1	0	0	1	0	0	S_3	1	1	1	0	S_2	S_1	S_0	20	3	
			Data																		

Instruction Set (cont)

			Operation Code																		
						B1								B2							
						B3								B4							
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	State (Note 1)	Bytes	Skip Condition
Immediate Data (cont)																					
SUINB	*A,byte	(A) ← (A) − byte	0	0	1	1	0	1	1	0				Data					7	2	No borrow
	r,byte	(r) ← (r) − byte	0	1	1	1	0	1	0	0	0	0	1	1	0	R_2	R_1	R_0	11	3	No borrow
						Data															
	sr2,byte	(sr2) ← (sr2) − byte	0	1	1	0	0	1	0	0	S_3	0	1	1	0	S_2	S_1	S_0	20	3	No borrow
						Data															
ANI	*A,byte	(A) ← (A) ∧ byte	0	0	0	0	0	1	1	1				Data					7	2	
	r,byte	(r) ← (r) ∧ byte	0	1	1	1	0	1	0	0	0	0	0	0	1	R_2	R_1	R_0	11	3	
						Data															
	sr2,byte	(sr2) ← (sr2) ∧ byte	0	1	1	0	0	1	0	0	S_3	0	0	0	1	S_2	S_1	S_0	20	3	
						Data															
ORI	*A,byte	(A) ← (A) V byte	0	0	0	1	0	1	1	1				Data					7	2	
	r,byte	(r) ← (r) V byte	0	1	1	1	0	1	0	0	0	0	0	1	1	R_2	R_1	R_0	11	3	
						Data															
	sr2,byte	(sr2) ← (sr2) V byte	0	1	1	0	0	1	0	0	S_3	0	0	1	1	S_2	S_1	S_0	20	3	
						Data															
XRI	*A,byte	(A) ← (A) ∀ byte	0	0	0	1	0	1	1	0				Data					7	2	
	r,byte	(r) ← (r) ∀ byte	0	1	1	1	0	1	0	0	0	0	0	1	0	R_2	R_1	R_0	11	3	
						Data															
	sr2,byte	(sr2) ← (sr2) ∀ byte	0	1	1	0	0	1	0	0	S_3	0	0	1	0	S_2	S_1	S_0	20	3	
						Data															
GTI	*A,byte	(A) − byte − 1	0	0	1	0	0	1	1	1				Data					7	2	No borrow
	r,byte	(r) − byte − 1	0	1	1	1	0	1	0	0	0	0	1	0	1	R_2	R_1	R_0	11	3	No borrow
						Data															
	sr2,byte	(sr2) − byte − 1	0	1	1	0	0	1	0	0	S_3	0	1	0	1	S_2	S_1	S_0	14	3	No borrow
						Data															
LTI	*A,byte	(A) − byte	0	0	1	1	0	1	1	1				Data					7	2	Borrow
	r,byte	(r) − byte	0	1	1	1	0	1	0	0	0	0	1	1	1	R_2	R_1	R_0	11	3	Borrow
						Data															
	sr2,byte	(sr2) − byte	0	1	1	0	0	1	0	0	S_3	0	1	1	1	S_2	S_1	S_0	14	3	Borrow
						Data															
NEI	*A,byte	(A) − byte	0	1	1	0	0	1	1	1				Data					7	2	No zero
	r,byte	(r) − byte	0	1	1	1	0	1	0	0	0	1	1	0	1	R_2	R_1	R_0	11	3	No zero
						Data															

Instruction Set (cont)

			Operation Code																		
			B1								B2										
			B3								B4										
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	State (Note 1)	Bytes	Skip Condition
Immediate Data (cont)																					
NEI	sr2,byte	(sr2) – byte	0	1	1	0	0	1	0	0	S_3	1	1	0	1	S_2	S_1	S_0	14	3	No zero
			Data																		
EQI	*A,byte	(A) – byte	0	1	1	1	0	1	1	1	Data								7	2	Zero
	r,byte	(r) – byte	0	1	1	1	0	1	0	0	0	1	1	1	1	R_2	R_1	R_0	11	3	Zero
			Data																		
	sr2,byte	(sr2) – byte	0	1	1	0	0	1	0	0	S_3	1	1	1	1	S_2	S_1	S_0	14	3	Zero
			Data																		
ONI	*A,byte	(A) ∧ byte	0	1	0	0	0	1	1	1	Data								7	2	No zero
	r,byte	(r) ∧ byte	0	1	1	1	0	1	0	0	0	1	0	0	1	R_2	R_1	R_0	11	3	No zero
			Data																		
	sr2,byte	(sr2) ∧ byte	0	1	1	0	0	1	0	0	S_3	1	0	0	1	S_2	S_1	S_0	14	3	No zero
			Data																		
OFFI	*A,byte	(A) ∧ byte	0	1	0	1	0	1	1	1	Data								7	2	Zero
	r,byte	(r) ∧ byte	0	1	1	1	0	1	0	0	0	1	0	1	1	R_2	R_1	R_0	11	3	Zero
			Data																		
	sr2,byte	(sr2) ∧ byte	0	1	1	0	0	1	0	0	S_3	1	0	1	1	S_2	S_1	S_0	14	3	Zero
			Data																		
Working Register																					
ADDW	wa	(A) ← (A) + ((V)•(wa))	0	1	1	1	0	1	0	0	1	1	0	0	0	0	0	0	14	3	
			Offset																		
ADCW	wa	(A) ← (A) + ((V)•(wa)) + (CY)	0	1	1	1	0	1	0	0	1	1	0	1	0	0	0	0	14	3	
			Offset																		
ADDNCW	wa	(A) ← (A) + ((V)•(wa))	0	1	1	1	0	1	0	0	1	0	1	0	0	0	0	0	14	3	No carry
			Offset																		
SUBW	wa	(A) ← (A) – ((V)•(wa))	0	1	1	1	0	1	0	0	1	1	1	0	0	0	0	0	14	3	
			Offset																		
SBBW	wa	(A) ← (A) – ((V)•(wa)) – (CY)	0	1	1	1	0	1	0	0	1	1	1	1	0	0	0	0	14	3	
			Offset																		
SUBNBW	wa	(A) ← (A) – ((V)•(wa))	0	1	1	1	0	1	0	0	1	0	1	1	0	0	0	0	14	3	No borrow
			Offset																		
ANAW	wa	(A) ← (A) ∧ ((V)•(wa))	0	1	1	1	0	1	0	0	1	0	0	0	1	0	0	0	14	3	
			Offset																		

Instruction Set (cont)

			Operation Code																		
						B1								B2							
						B3								B4							
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	State (Note 1)	Bytes	Skip Condition
Working Register (cont)																					
ORAW	wa	(A) ← (A) V ((V)•(wa))	0	1	1	1	0	1	0	0	1	0	0	1	1	0	0	0	14	3	
			Offset																		
XRAW	wa	(A) ← (A)∀((V)•(wa))	0	1	1	1	0	1	0	0	1	0	0	1	0	0	0	0	14	3	
			Offset																		
GTAW	wa	(A) – ((V)•(wa)) – 1	0	1	1	1	0	1	0	0	1	0	1	0	1	0	0	0	14	3	No borrow
			Offset																		
LTAW	wa	(A) – ((V)•(wa))	0	1	1	1	0	1	0	0	1	0	1	1	1	0	0	0	14	3	Borrow
			Offset																		
NEAW	wa	(A) – ((V)•(wa))	0	1	1	1	0	1	0	0	1	1	1	0	1	0	0	0	14	3	No zero
			Offset																		
EQAW	wa	(A) – ((V)•(wa))	0	1	1	1	1	1	0	0	1	1	1	1	1	0	0	0	14	3	Zero
			Offset																		
ONAW	wa	(A) ∧ ((V)•(wa))	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0	0	14	3	No zero
			Offset																		
OFFAW	wa	(A) ∧ ((V)•(wa))	0	1	1	1	0	1	0	0	1	1	0	1	1	0	0	0	14	3	Zero
			Offset																		
ANIW	*wa,byte	((V)•(wa)) ← ((V)•(wa)) ∧ byte	0	0	0	0	0	1	0	1	Offset								19	3	
			Data																		
ORIW	*wa,byte	((V)•(wa)) ← ((V)•(wa)) V byte	0	0	0	1	0	1	0	1	Offset								19	3	
			Data																		
GTIW	*wa,byte	((V)•(wa)) – byte – 1	0	0	1	0	0	1	0	1	Offset								13	3	No borrow
			Data																		
LTIW	*wa,byte	((V)•(wa)) – byte	0	0	1	1	0	1	0	1	Offset								13	3	Borrow
			Data																		
NEIW	*wa,byte	((V)•(wa)) – byte	0	1	1	0	0	1	0	1	Offset								13	3	No zero
			Data																		
EQIW	*wa,byte	((V)•(wa)) – byte	0	1	1	1	0	1	0	1	Offset								13	3	Zero
			Data																		
ONIW	*wa,byte	((V)•(wa)) ∧ byte	0	1	0	0	0	1	0	1	Offset								13	3	No zero
			Data																		
OFFIW	*wa,byte	((V)•(wa)) ∧ byte	0	1	0	1	0	1	0	1	Offset								13	3	Zero
			Data																		

Instruction Set (cont)

			Operation Code																		
			B1								B2										
			B3								B4								State		Skip
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	(Note 1)	Bytes	Condition
16-Bit Arithmetic																					
EADD	EA,r2	(EA) ← (EA) + (r2)	0	1	1	1	0	0	0	0	0	1	0	0	0	0	R_1	R_0	11	2	
DADD	EA,rp3	(EA) ← (EA) + (rp3)	0	1	1	1	0	1	0	0	1	1	0	0	0	1	P_1	P_0	11	2	
DADC	EA,rp3	(EA) ← (EA) + (rp3) + (CY)	0	1	1	1	0	1	0	0	1	1	0	1	0	1	P_1	P_0	11	2	
DADDNC	EA,rp3	(EA) ← (EA) + (rp3)	0	1	1	1	0	1	0	0	1	0	1	0	0	1	P_1	P_0	11	2	No carry
ESUB	EA,r2	(EA) ← (EA) − (r2)	0	1	1	1	0	0	0	0	0	1	1	0	0	0	R_1	R_0	11	2	
DSUB	EA,rp3	(EA) ← (EA) − (rp3)	0	1	1	1	0	1	0	0	1	1	1	0	0	1	P_1	P_0	11	2	
DSBB	EA,rp3	(EA) ← (EA) − (rp3) − (CY)	0	1	1	1	0	1	0	0	1	1	1	1	0	1	P_1	P_0	11	2	
DSUBNB	EA,rp3	(EA) ← (EA) − (rp3)	0	1	1	1	0	1	0	0	1	0	1	1	0	1	P_1	P_0	11	2	No borrow
DAN	EA,rp3	(EA) ← (EA) ∧ (rp3)	0	1	1	1	0	1	0	0	1	0	0	0	1	1	P_1	P_0	11	2	
DOR	EA,rp3	(EA) ← (EA) V (rp3)	0	1	1	1	0	1	0	0	1	0	0	1	1	1	P_1	P_0	11	2	
DXR	EA,rp3	(EA) ← (EA) ∀ (rp3)	0	1	1	1	0	1	0	0	1	0	0	1	0	1	P_1	P_0	11	2	
DGT	EA,rp3	(EA) − (rp3) − 1	0	1	1	1	0	1	0	0	1	0	1	0	1	1	P_1	P_0	11	2	No borrow
DLT	EA,rp3	(EA) − (rp3)	0	1	1	1	0	1	0	0	1	0	1	1	1	1	P_1	P_0	11	2	Borrow
DNE	EA,rp3	(EA) − (rp3)	0	1	1	1	0	1	0	0	1	1	1	0	1	1	P_1	P_0	11	2	No zero
DEQ	EA,rp3	(EA) − (rp3)	0	1	1	1	0	1	0	0	1	1	1	1	1	1	P_1	P_0	11	2	Zero
DON	EA,rp3	(EA) ∧ (rp3)	0	1	1	1	0	1	0	0	1	1	0	0	1	1	P_1	P_0	11	2	No zero
DOFF	EA,rp3	(EA) ∧ (rp3)	0	1	1	1	0	1	0	0	1	1	0	1	1	1	P_1	P_0	11	2	Zero
Multiply/Divide																					
MUL	r2	(EA) ← (A) x (r2)	0	1	0	0	1	0	0	0	0	0	1	0	1	1	R_1	R_0	32	2	
DIV	r2	(EA) ← (EA) ÷ (r2), (r2) ← Remainder	0	1	0	0	1	0	0	0	0	0	1	1	1	1	R_1	R_0	59	2	
Increment/Decrement																					
INR	r2	(r2) ← (r2) + 1	0	1	0	0	0	0	R_1	R_0									4	1	Carry
INRW	*wa	((V)•(wa)) ← ((V)•(wa)) + 1	0	0	1	0	0	0	0	0				Offset					16	2	Carry
INX	rp	(rp) ← (rp) + 1	0	0	P_1	P_0	0	0	1	0									7	1	
	EA	(EA) ← (EA) + 1	1	0	1	0	1	0	0	0									7	1	
DCR	r2	(r2) ← (r2) − 1	0	1	0	1	0	0	R_1	R_0									4	1	Borrow
DCRW	*wa	((V)•(wa)) ← ((V)•(wa)) − 1	0	0	1	1	0	0	0	0				Offset					16	2	Borrow
DCX	rp	(rp) ← (rp) − 1	0	0	P_1	P_0	0	0	1	1									7	1	
	EA	(EA) ← (EA) − 1	1	0	1	0	1	0	0	1									7	1	
Others																					
DAA		Decimal Adjust Accumulator	0	1	1	0	0	0	0	1									4	1	
STC		(CY) ← 1	0	1	0	0	1	0	0	0	0	0	1	0	1	0	1	1	8	2	
CLC		(CY) ← 0	0	1	0	0	1	0	0	0	0	0	1	0	1	0	1	0	8	2	

Instruction Set (cont)

			Operation Code																		
							B1 B3							B2 B4							
Mnemonic	**Operand**	**Operation**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**State (Note 1)**	**Bytes**	**Skip Condition**
Others (cont)																					
NEGA		$(A) \leftarrow (\overline{A}) + 1$	0	1	0	0	1	0	0	0	0	0	1	1	1	0	1	0	8	2	
Rotate and Shift																					
RLD		Rotate left digit $(A_{3\text{-}0}) \leftarrow ((HL))_{7\text{-}4}$, $((HL))_{7\text{-}4} \leftarrow ((HL))_{3\text{-}0}$, $((HL))_{3\text{-}0} \leftarrow (A_{3\text{-}0})$	0	1	0	0	1	0	0	0	0	0	1	1	1	0	0	0	17	2	
RRD		Rotate right digit $((HL))_{7\text{-}4} \leftarrow (A_{3\text{-}0})$, $((HL))_{3\text{-}0} \leftarrow ((HL))_{7\text{-}4}$, $(A_{3\text{-}0}) \leftarrow ((HL))_{3\text{-}0}$	0	1	0	0	1	0	0	0	0	0	1	1	1	0	0	1	17	2	
RLL	r2	$(r2_{m+1}) \leftarrow (r2_m)$, $(r2_0) \leftarrow (CY)$, $(CY) \leftarrow (r2_7)$	0	1	0	0	1	0	0	0	0	0	1	1	0	1	R_1	R_0	8	2	
RLR	r2	$(r2_{m-1}) \leftarrow (r2_m)$,$(r2_7) \leftarrow (CY)$, $(CY) \leftarrow (r2_0)$	0	1	0	0	1	0	0	0	0	0	1	1	0	0	R_1	R_0	8	2	
SLL	r2	$(r2_{m+1}) \leftarrow (r2_m)$, $(r2_0) \leftarrow 0$, $(CY) \leftarrow (r2_7)$	0	1	0	0	1	0	0	0	0	0	1	0	0	1	R_1	R_0	8	2	
SLR	r2	$(r2_{m-1}) \leftarrow (r2_m)$, $(r2_7) \leftarrow 0$, $(CY) \leftarrow (r2_0)$	0	1	0	0	1	0	0	0	0	0	1	0	0	0	R_1	R_0	8	2	
SLLC	r2	$(r2_{m+1}) \leftarrow (r2_m)$, $(r2_0) \leftarrow 0$, $(CY) \leftarrow (r2_7)$	0	1	0	0	1	0	0	0	0	0	0	0	0	1	R_1	R_0	8	2	Carry
SLRC	r2	$(r2_{m-1}) \leftarrow (r2_m)$, $(r2_7) \leftarrow 0$, $(CY) \leftarrow (r2_0)$	0	1	0	0	1	0	0	0	0	0	0	0	0	0	R_1	R_0	8	2	Carry
DRLL	EA	$(EA_{n+1}) \leftarrow (EA_n)$, $(EA_0) \leftarrow (CY)$, $(CY) \leftarrow (EA_{15})$	0	1	0	0	1	0	0	0	1	0	1	1	0	1	0	0	8	2	
DRLR	EA	$(EA_{n-1}) \leftarrow (EA_n)$, $(EA_{15}) \leftarrow (CY)$, $(CY) \leftarrow (EA_0)$	0	1	0	0	1	0	0	0	1	0	1	1	0	0	0	0	8	2	
DSLL	EA	$(EA_{n+1}) \leftarrow (EA_n)$, $(EA_0) \leftarrow 0$, $(CY) \leftarrow (EA_{15})$	0	1	0	0	1	0	0	0	1	0	1	0	0	1	0	0	8	2	
DSLR	EA	$(EA_{n-1}) \leftarrow (EA_n)$, $(EA_{15}) \leftarrow 0$, $(CY) \leftarrow (EA_0)$	0	1	0	0	1	0	0	0	1	0	1	0	0	0	0	0	8	2	
Jump																					
JMP	*word	$(PC) \leftarrow$ word	0	1	0	1	0	1	0	0				Low addr					10	3	
						High addr															
JB		$(PC_H) \leftarrow (B)$, $(PC_L) \leftarrow (C)$	0	0	1	0	0	0	0	1									4	1	
JR	word	$(PC) \leftarrow (PC) + 1 +$ jdisp 1	1	1	←		jdisp1			→									10	1	
JRE	*word	$(PC) \leftarrow (PC) + 2 +$ jdisp	0	1	0	0	1	1	1	←				jdisp				→	10	2	
JEA		$(PC) \leftarrow (EA)$	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	0	8	2	
Call																					
CALL	*word	$((SP) - 1) \leftarrow ((PC) + 3)_H$, $((SP) - 2) \leftarrow ((PC) + 3)_L$, $(PC) \leftarrow$ word, $(SP) \leftarrow (SP) - 2$	0	1	0	0	0	0	0	0				Low addr					16	3	
						High addr															
CALB		$((SP) - 1) \leftarrow ((PC) + 2)_H$, $((SP) - 2) \leftarrow ((PC) + 2)_L$, $(PC_H) \leftarrow (B)$, $(PC_L) \leftarrow (C)$, $(SP) \leftarrow (SP) - 2$	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	1	17	2	

Instruction Set (cont)

			Operation Code																		
			B1 B3								B2 B4										
Mnemonic	Operand	Operation	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	State (Note 1)	Bytes	Skip Condition
Call (cont)																					
CALF	*word	((SP) − 1) ← ((PC) + 2)$_H$, ((SP) − 2) ← ((PC) + 2)$_L$, (PC$_{15-11}$) ← 00001, (PC$_{10-0}$) ← fa, (SP) ← (SP) − 2	0	1	1	1	1	← fa →											13	2	
CALT	word	((SP) − 1) ← ((PC) + 1)$_H$, ((SP) − 2) ← ((PC) + 1)$_L$, (PC$_L$) ← (128 + 2ta), (PC$_H$) ← (129 + 2ta), (SP) ← (SP) − 2	1	0	0	← ta →													16	1	
SOFTI		((SP) − 1) ← (PSW), ((SP) − 2) ← ((PC) + 1)$_H$, ((SP) − 3) ← ((PC) + 1)$_L$, (PC) ← 0060H, (SP) ← (SP) − 3	0	1	1	1	0	0	1	0									16	1	
Return																					
RET		(PC$_L$) ← ((SP)), (PC$_H$) ← ((SP) + 1) (SP) ← (SP) + 2	1	0	1	1	1	0	0	0									10	1	
RETS		(PC$_L$) ← ((SP)), (PC$_H$) ← ((SP) + 1) (SP) ← (SP) + 2, (PC) ← (PC) + n	1	0	1	1	1	0	0	1									10	1	Unconditional Skip
RETI		(PC$_L$) ← ((SP)), (PC$_H$) ← ((SP) + 1) (PSW) ← ((SP) + 2), (SP) ← (SP) + 3	0	1	1	0	0	0	1	0									13	1	
Skip																					
BIT	*bit, wa	Skip if ((V)•(wa)) bit = 1	0	1	0	1	1	B_2	B_1	B_0	← Offset →								10	2	Bit Test
SK	f	Skip if f = 1	0	1	0	0	1	0	0	0	0	0	0	0	1	F_2	F_1	F_0	8	2	f = 1
SKN	f	Skip if f = 0	0	1	0	0	1	0	0	0	0	0	0	1	1	F_2	F_1	F_0	8	2	f = 0
SKIT	irf	Skip if irf = 1, then reset irf	0	1	0	0	1	0	0	0	0	1	0	I_4	I_3	I_2	I_1	I_0	8	2	irf = 1
SKNIT	irf	Skip if irf = 0 Reset irf if irf = 1 and don't skip	0	1	0	0	1	0	0	0	0	1	1	I_4	I_3	I_2	I_1	I_0	8	2	irf = 0
CPU Control																					
NOP		No operation	0	0	0	0	0	0	0	0									4	1	
EI		Enable interrupt	1	0	1	0	1	0	1	0									4	1	
DI		Disable interrupt	1	0	1	1	1	0	1	0									4	1	
HLT		Set HALT mode	0	1	0	0	1	0	0	0	0	0	1	1	1	0	1	1	12	2	
STOP		Set STOP mode	0	1	0	0	1	0	0	0	1	0	1	1	1	0	1	1	12	2	

Section 6
μPD78K2 Series:
8-Bit, Advanced Microcomputers

NEC

NEC Electronics Inc.

µPD7821x Advanced, 8-Bit Real-Time Control Microcomputers With A/D Converter

Description

The µPD78213, µPD78214, and µPD78P214 are high-performance, 8-bit, single-chip microcomputers. They contain extended addressing capabilities for up to 1M byte of external memory. The devices also integrate sophisticated analog and digital peripherals as well as two low-power standby modes that make them ideal for low-power/battery backup applications.

The µPD7821x family focuses on embedded control with features like hardware multiply and divide, two levels of interrupt response, four banks of main registers for multitasking, and macroservice for processor-independent peripheral and memory DMA. Augmenting this high-performance core are advanced components like a high-precision A/D converter, two independent serial interfaces, several counter/timers for PWM outputs as well as a real-time output port. On board memory includes 512 bytes of RAM and 16K bytes of mask ROM, EPROM, or OTP ROM.

The macroservice routine allows data to be transferred between any combination of memory and peripherals independent of the current program execution. The four banks of processor registers allow simplified context switching to be performed. Both features combined with powerful on-chip peripherals make this part ideal for a wide variety of embedded control applications.

Features

- □ Complete single-chip microcomputer
  - — 8-bit ALU
  - — 16K ROM
  - — 512 bytes RAM
  - — Both 1-bit and 8-bit logic
- □ Instruction prefetch queue
- □ Hardware multiply and divide
- □ Memory expansion
  - — 8085 bus-compatible
  - — 64K program address space
  - — 1M data address space
- □ Large I/O capacity: up to 54 I/O port lines
- □ Software pullup options
- □ Extensive timer/counter functions
  - — One 16-bit timer/counter/event counter
  - — Three 8-bit timer/counter/event counter
- □ Four timer-controlled PWM channels
- □ Two 4-bit real-time output ports
- □ Extensive interrupt handler
  - — Vectored interrupt handling
  - — Programmable priority
  - — Macroservice mode
- □ Two independent serial ports
- □ Refresh output for pseudostatic RAM
- □ On-chip clock generator
  - — 12-MHz maximum CPU clock frequency
  - — 0.33-µs instruction cycle
- □ CMOS silicon gate technology
- □ 5-volt power supply

Ordering Information

Part Number	ROM	Package
µPD78213CW	ROMless	64-pin plastic shrink DIP
µPD78213GQ-36		64-pin plastic QUIP
µPD78213GJ		74-pin plastic QFP
µPD78213L		68-pin PLCC
µPD78214CW	16K Mask	64-pin plastic shrink DIP
µPD78214GQ-36	ROM	64-plastic QUIP
µPD78214GJ		74-pin plastic QFP
µPD78214L		68-pin PLCC
µPD78P214CW	16K OTP	64-pin plastic shrink DIP
µPD78P214GQ-36	ROM	64-pin plastic QUIP
µPD78P214GJ		74-pin plastic QFP
µPD78P214L		68-pin PLCC
µPD78P214DW	16K UV	64-pin shrink cerdip
µPD78P214R	EPROM	64-pin ceramic QUIP

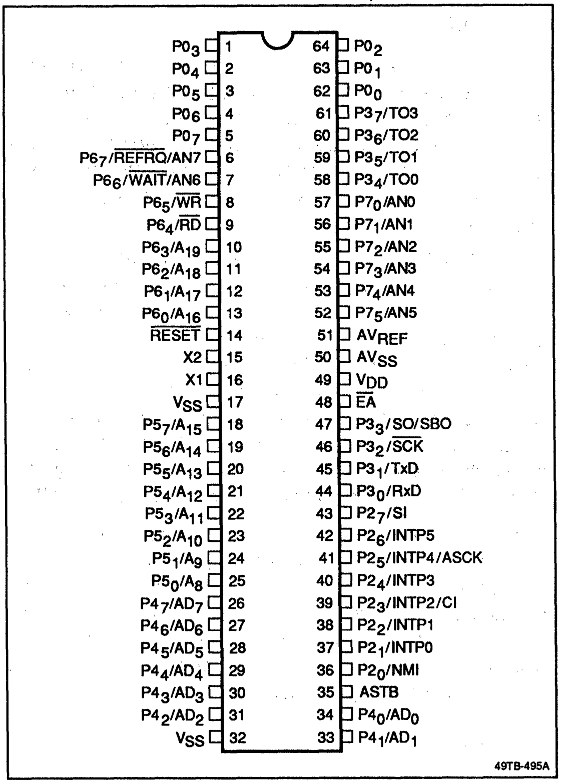
Note:
A 74-pin plastic QFP that can be reflow soldered will be available.

Pin Identification

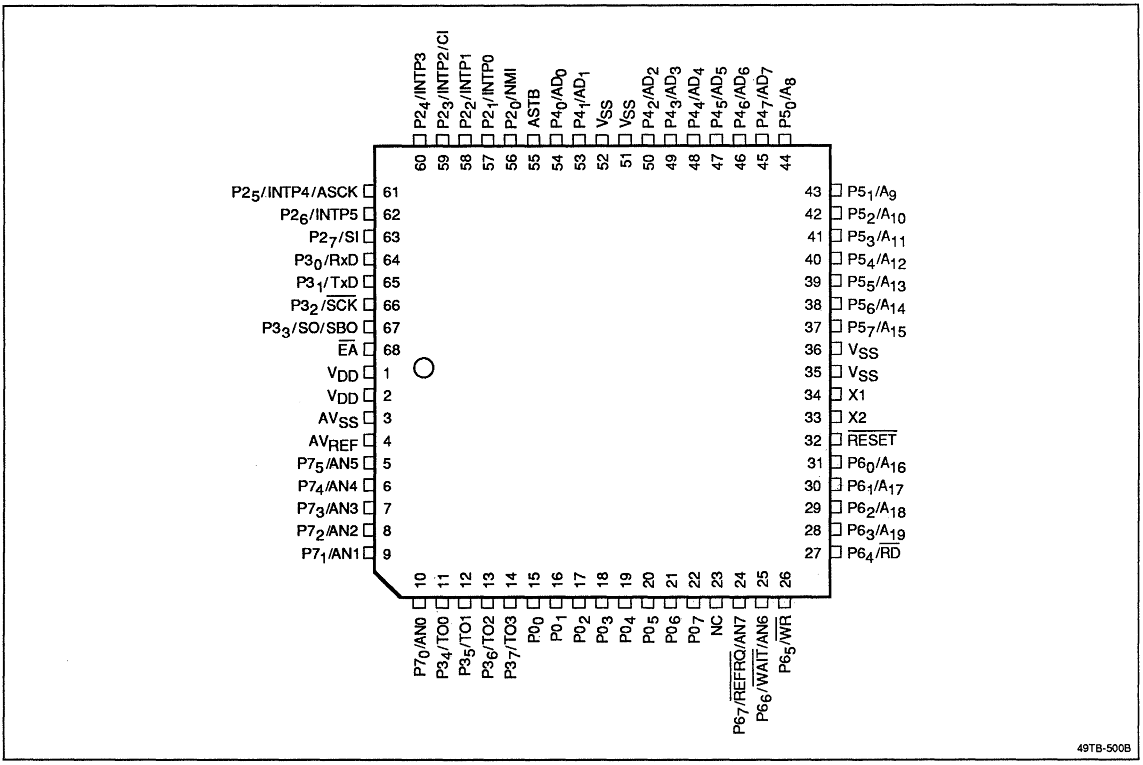
Symbol	Function
$P0_0$-$P0_7$	Output port 0
$P2_0$/NMI	Input port 2/Non-maskable interrupt input
$P2_1$-$P2_2$/INTP0-INTP1	Input port 2/External interrupt input/timer trigger
$P2_3$/INTP2/CI	Input port 2/External interrupt input/ Clock input
$P2_4$/INTP3	Input port 2/External interrupt input/timer trigger
$P2_5$/INTP4/ASCK	Input port 2/External interrupt input/ Asynchronous serial clock
$P2_6$/INTP5	Input port 2/External interrupt input
$P2_7$/SI	Input port 2/Serial input
$P3_0$/RxD	I/O port 3/Serial receive input
$P3_1$/TxD	I/O port 3/Serial transmit output
$P3_2$/$\overline{\text{SCK}}$	I/O port 3/Serial clock input/output
$P3_3$/SO/SBO	I/O port 3/Serial output/Serial bus I/O
$P3_4$-$P3_7$/TO0-TO3	I/O port 3/Timer output
$P4_0$-$P4_7$/AD_0-AD_7	I/O port 4/Lower address byte/data bus
$P5_0$-$P5_7$/A_8-A_{15}	I/O port 5/Upper address byte
$P6_0$-$P6_3$/A_{16}-A_{19}	Output port 6/Extended address nibble
$P6_4$/$\overline{\text{RD}}$	I/O port 6/Read strobe output
$P6_5$/$\overline{\text{WR}}$	I/O port 6/Write strobe output
$P6_6$/$\overline{\text{WAIT}}$/AN6	I/O port 6/Wait input/A/D converter input
$P6_7$/$\overline{\text{REFRQ}}$/AN7	I/O port 6/Refresh output/A/D converter input
$P7_0$-$P7_5$/AN0-AN5	Input port 7/A/D converter input
ASTB	Address strobe output
$\overline{\text{RESET}}$	External reset input
$\overline{\text{EA}}$	External memory access control input
X1, X2	External crystal or external clock input
AV_{REF}	A/D converter reference voltage
AV_{ss}	Analog ground
V_{DD}	Positive power supply input
V_{SS}	Power return; normally ground
NC	No connection

Pin Configurations

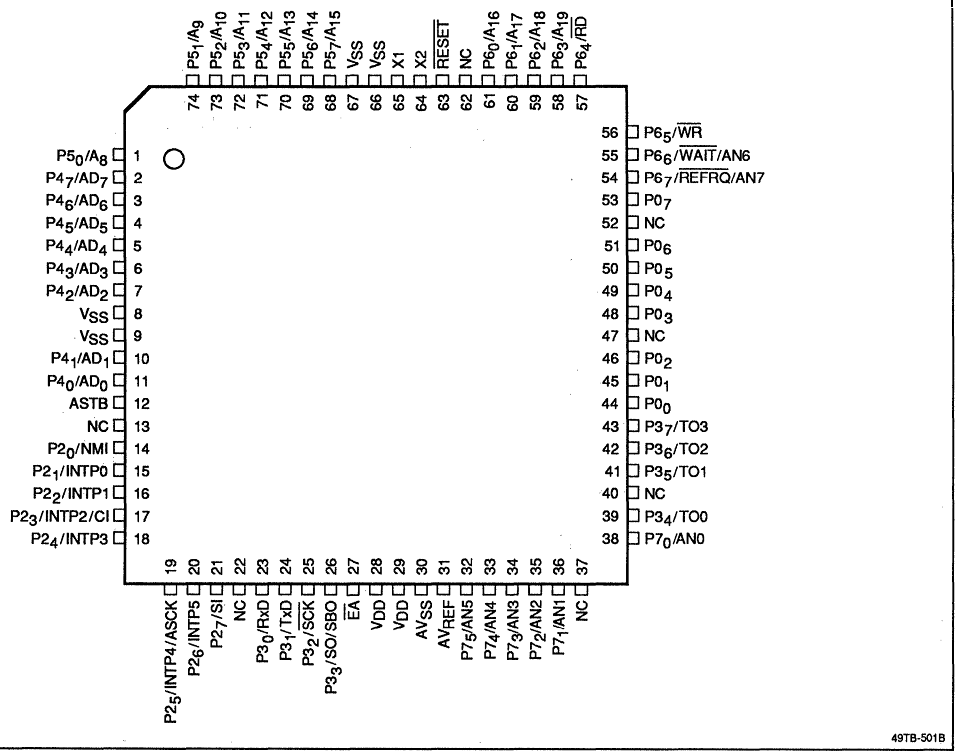
64-Pin Shrink DIP and QUIP (Plastic or Ceramic)



68-Pin PLCC


P2$_4$/INTP3 60
P2$_3$/INTP2/CI 59
P2$_2$/INTP1 58
P2$_1$/INTP0 57
P2$_0$/NMI 56
ASTB 55
P4$_0$/AD$_0$ 54
P4$_1$/AD$_1$ 53
V$_{SS}$ 52
V$_{SS}$ 51
P4$_2$/AD$_2$ 50
P4$_3$/AD$_3$ 49
P4$_4$/AD$_4$ 48
P4$_5$/AD$_5$ 47
P4$_6$/AD$_6$ 46
P4$_7$/AD$_7$ 45
P5$_0$/A$_8$ 44
P2$_5$/INTP4/ASCK 61
P2$_6$/INTP5 62
P2$_7$/SI 63
P3$_0$/RxD 64
P3$_1$/TxD 65
P3$_2$/$\overline{SCK}$ 66
P3$_3$/SO/SBO 67
$\overline{EA}$ 68
V$_{DD}$ 1
V$_{DD}$ 2
AV$_{SS}$ 3
AV$_{REF}$ 4
P7$_5$/AN5 5
P7$_4$/AN4 6
P7$_3$/AN3 7
P7$_2$/AN2 8
P7$_1$/AN1 9
43 P5$_1$/A$_9$
42 P5$_2$/A$_{10}$
41 P5$_3$/A$_{11}$
40 P5$_4$/A$_{12}$
39 P5$_5$/A$_{13}$
38 P5$_6$/A$_{14}$
37 P5$_7$/A$_{15}$
36 V$_{SS}$
35 V$_{SS}$
34 X1
33 X2
32 $\overline{RESET}$
31 P6$_0$/A$_{16}$
30 P6$_1$/A$_{17}$
29 P6$_2$/A$_{18}$
28 P6$_3$/A$_{19}$
27 P6$_4$/$\overline{RD}$
10 P7$_0$/AN0
11 P3$_4$/TO0
12 P3$_5$/TO1
13 P3$_6$/TO2
14 P3$_7$/TO3
15 P0$_0$
16 P0$_1$
17 P0$_2$
18 P0$_3$
19 P0$_4$
20 P0$_5$
21 P0$_6$
22 P0$_7$
23 NC
24 P6$_7$/$\overline{REFRQ}$/AN7
25 P6$_6$/$\overline{WAIT}$/AN6
26 P6$_5$/$\overline{WR}$
49TB-500B

74-Pin Plastic QFP



Pin Functions

P0$_0$-P0$_7$. Port 0 is an 8-bit, tristate output port with direct transistor drive capability. Port 0 can also be configured as two 4-bit, real-time (timer-controlled) output ports.

P2$_0$-P2$_7$. Port 2 is an 8-bit input port with the programmable pullup option except for P2$_0$ and P2$_1$.

NMI. Non-maskable interrupt input.

INTP0-INTP5. External interrupt inputs. INTP0, INTP1, and INTP3 are timer capture trigger inputs.

CI. External clock input to the timer.

ASCK. Asynchronous serial clock input.

SI. Serial data input for three-wire serial I/O mode.

P3$_0$-P3$_7$. Port 3 is an 8-bit tristate I/O port with the programmable pullup option.

RxD. Receive serial data input.

TxD. Transmit serial data output.

$\overline{\text{SCK}}$. Serial shift clock output.

SO. Serial data output for three-wire serial I/O mode.

SBO. I/O bus for the clocked serial interface.

TO0-TO3. Timer flip-flop outputs.

P4$_0$-P4$_7$. Port 4 is an 8-bit, bidirectional tristate port with the programmable pullup option. Port 4 has direct LED drive capability.

AD$_0$-AD$_7$. Multiplexed address/data bus used with external memory or expanded I/O.

P5$_0$-P5$_7$. Port 5 is an 8-bit, bidirectional tristate port with the programmable pullup option. Port 5 has direct LED drive capability.

A$_8$-A$_{15}$. Upper-order address bus used with external memory or expanded I/O.

P6$_0$-P6$_3$. Pins P6$_0$-P6$_3$ of port 6 are outputs.

A$_{16}$-A$_{19}$. Extended-order address bus used with external memory.

P6$_4$-P6$_7$. Pins P6$_4$-P6$_7$ of port 6 are tristate I/Os with the programmable pullup option.

$\overline{\text{RD}}$. Read strobe output used by external memory (or data registers) to place data on the I/O bus during a read operation.

$\overline{\text{WR}}$. Write strobe output used by external memory (or data registers) to latch data from the I/O bus during a write operation.

$\overline{\text{WAIT}}$. Wait signal input.

$\overline{\text{REFRQ}}$. Refresh pulse output used by external pseudostatic memory.

AN6, AN7. Analog voltage inputs to A/D converter.

P7$_0$-P7$_5$. Port 7 is a 6-bit input port.

AN0-AN5. Analog voltage inputs to A/D converter.

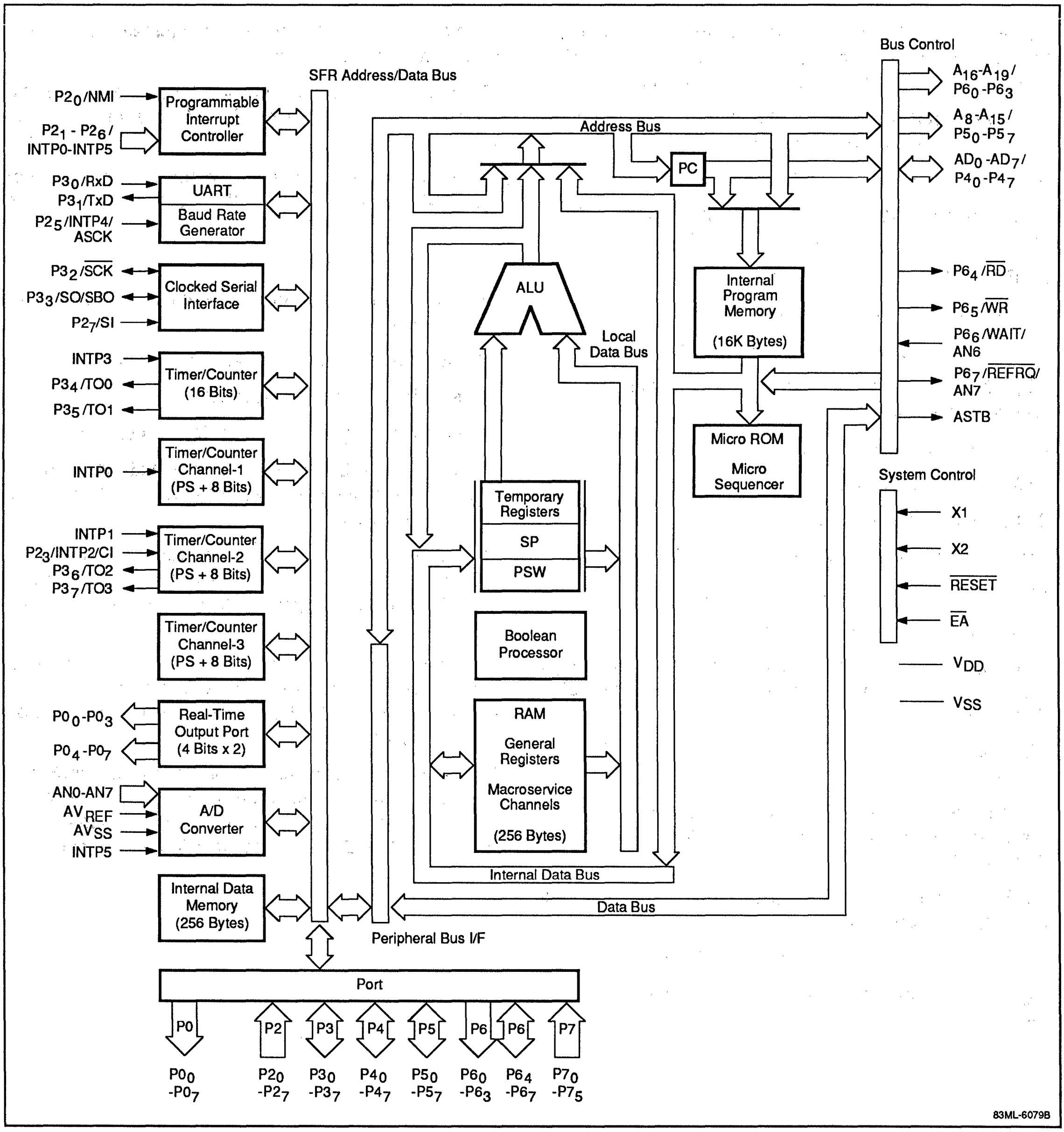
ASTB. Address strobe output used by external circuitry to latch the low-order 8 address bits during the first part of a read or write cycle.

$\overline{\text{RESET}}$. A low level on this external reset input sets all registers to their specified reset values. This pin, together with P2$_0$/NMI, sets the μPD78P214 in the PROM programming mode.

$\overline{\text{EA}}$. Control signal input that selects external memory ($\overline{\text{EA}}$ low) or internal ROM ($\overline{\text{EA}}$ high) as the program memory. When $\overline{\text{EA}}$ is low, μPD78214 is set in ROMless mode and external memory is accessed.

X1, X2. For frequency control of the internal clock oscillator, a crystal is connected to X1 and X2. If the clock is supplied by an external source, the clock signal is connected to X1 and the inverted clock signal is connected to X2.

μPD7821x Block Diagram



FUNCTIONAL DESCRIPTION

Timing

The maximum clock frequency is 12 MHz. The clock is derived from an external crystal or an external oscillator. The internal processor clock is two-phase and the machine states are executed at a rate of 6 MHz. The shortest instructions require two states (333 ns). The CPU contains a one-byte instruction prefetch. This allows a subsequent instruction to be fetched during the execution of an instruction that does not reference memory.

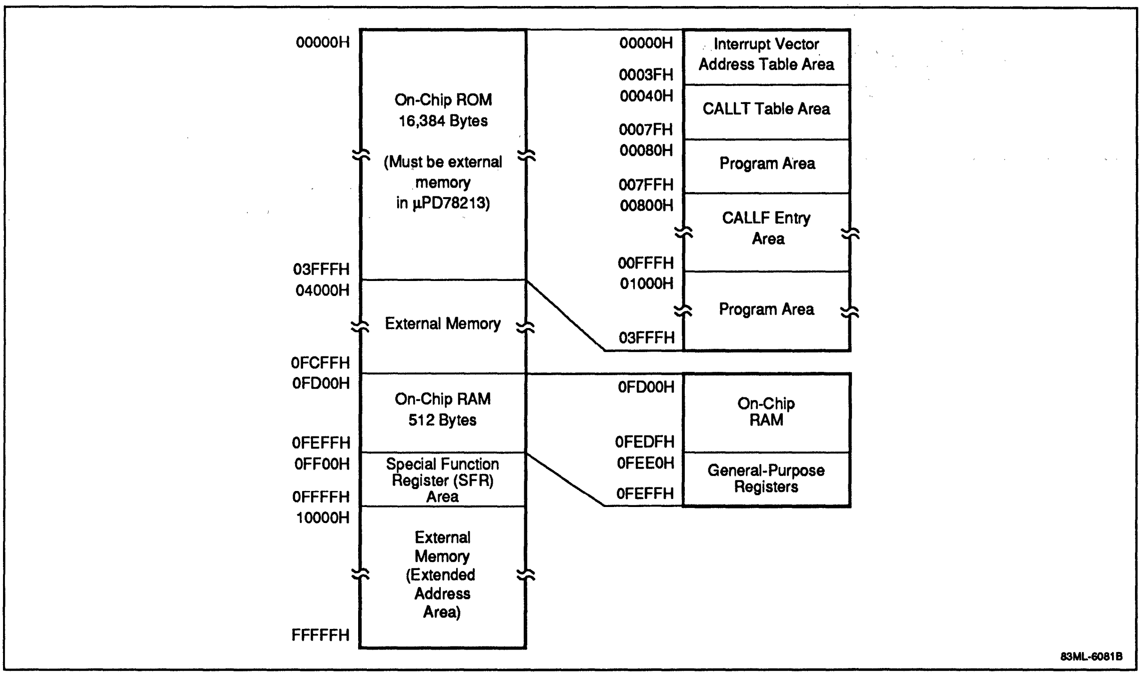
Memory Map

The μPD7821x has 1M byte of address space. This address space is partitioned into 64K bytes of program memory starting at address 00000H. (See figure 1.) The remainder of the 1M bytes can be accessed as data memory space.

External memory is supported by I/O port 4, an 8-bit multiplexed address/data bus. The memory mapping register controls the size of external memory as well as the number of added wait states. The upper address byte is derived from port 5, and the extended address nibble is derived from port 6.

The μPD78214 has on-chip mask ROM occupying the space from 00000H to 03FFFH. When the ROM is used and no other program or data space is required, ports 4, 5, and 6 are available as additional I/O ports.

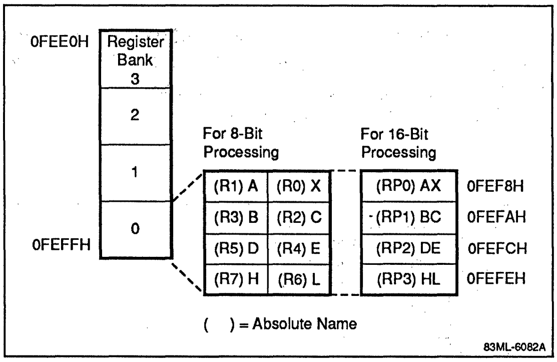
Figure 1. Memory Map



General-Purpose Registers

The general-purpose registers are mapped into specific addresses in data memory. They are made up of four banks, each bank consisting of eight 8-bit or four 16-bit registers. The register bank used is specified by a CPU instruction. This can be checked by reading RBS0 and RBS1 in the program status word (PSW).The general-purpose register configuration is shown in figure 2.

Figure 2. Register Mapping



Special Registers

There are three different special registers. The first is a 16-bit binary counter that holds the next program address to be executed and is named the program counter. The stack pointer is the second special 16-bit register. The stack pointer holds the address of the stack area (a last in, first out system). The third special register is an 8-bit program status word. This register contains various flags that are set or reset depending on the results of instruction execution. The program status word format is as follows:

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

Special Function Registers

These registers are assigned to special functions such as the mode and control registers for on-chip peripheral hardware. They are mapped into the 256-byte memory space from 0FF00H to 0FFFFH. Table 1 is a list of special function registers.

Table 1. Special Function Registers

				Handleable Bit Unit			
				1	8	16	
Address	Special Function Register (SFR) Name	Symbol	R/W	Bit	Bit	Bit	On Reset
0FF00H	Port 0	P0	R/W	o	o	–	Indeterminate
0FF02H	Port 2	P2	R	o	o	–	Indeterminate
0FF03H	Port 3	P3	R/W	o	o	–	Indeterminate
0FF04H	Port 4	P4	R/W	o	o	–	Indeterminate
0FF05H	Port 5	P5	R/W	o	o	–	Indeterminate
0FF06H	Port 6	P6	R/W	o	o	–	x0H
0FF07H	Port 7	P7	R/W	o	o	–	Indeterminate
0FF0AH	Port 0 buffer register (low)	P0L	R/W	o	o	–	Indeterminate
0FF0BH	Port 0 buffer register (high)	P0H	R/W	o	o	–	Indeterminate
0FF0CH	Real-time output port control register	RTPC	R/W	o	o	–	00H
0FF10H, 0FF11H	16-bit compare register 0 (16-bit timer/counter)	CR00	R/W	–	–	o	Indeterminate
0FF12H, 0FF13H	16-bit compare register 1 (16-bit timer/counter)	CR01	R/W	–	–	o	Indeterminate
0FF14H	8-bit compare register (8-bit timer/counter 1)	CR10	R/W	–	o	–	Indeterminate
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	–	o	–	Indeterminate
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	–	o	–	Indeterminate
0FF17H	8-bit compare register (8-bit timer/counter 3)	CR30	R/W	–	o	–	Indeterminate
0FF18H, 0FF19H	16-bit capture register (16-bit timer/counter)	CR02	R	–	–	o	Indeterminate
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	–	o	–	Indeterminate
0FF1CH	8-bit capture/compare register (8-bit timer/counter 1)	CR11	R/W	–	o	–	Indeterminate
0FF20H	Port 0 mode register	PM0	W	–	o	–	FFH
0FF23H	Port 3 mode register	PM3	W	–	o	–	FFH
0FF25H	Port 5 mode register	PM5	W	–	o	–	FFH
0FF26H	Port 6 mode register	PM6	R/W	–	o	–	FxH
0FF30H	Capture/compare control register 0	CRC0	W	–	o	–	10H
0FF31H	Timer output control register	TOC	W	–	o	–	00H
0FF32H	Capture/compare control register 1	CRC1	W	–	o	–	00H
0FF34H	Capture/compare control register 2	CRC2	W	–	o	–	00H
0FF40H	Pull-up option register	PUO	R/W	o	o	–	00H
0FF43H	Port 3 mode control register	PMC3	R/W	o	o	–	00H
0FF50H, 0FF51H	16-bit timer register 0	TM0	R	–	–	o	0000H
0FF52H	8-bit timer register 1	TM1	R	–	o	–	00H

Table 1. Special Function Registers (cont)

					Handleable Bit Unit			
Address	Special Function Register (SFR) Name	Symbol		R/W	1 Bit	8 Bit	16 Bit	On Reset
0FF54H	8-bit timer register 2	TM2		R	–	o	–	00H
0FF56H	8-bit timer register 3	TM3		R	–	o	–	00H
0FF5CH	Prescaler mode register 0	PRM0		W	–	o	–	00H
0FF5DH	Timer control register 0	TMC0		R/W	–	o	–	00H
0FF5EH	Prescaler mode register 1	PRM1		W	–	o	–	00H
0FF5FH	Timer control register 1	TMC1		R/W	–	o	–	00H
0FF68H	A/D converter mode register	ADM		R/W	o	o	–	00H
0FF6AH	A/D conversion result register	ADCR		R	o	o	–	Indeterminate
0FF80H	Clocked serial interface mode register	CSIM		R/W	o	o	–	00H
0FF82H	Serial bus interface control register	SBIC		R/W	o	o	–	00H
0FF86H	Serial shift register	SIO		R/W	–	o	–	Indeterminate
0FF88H	Asynchronous serial interface mode register	ASIM		R/W	o	o	–	80H
0FF8AH	Asynchronous serial interface status register	ASIS		R	o	o	–	00H
0FF8CH	Serial receive buffer: UART	RxB		R	–	o	–	Indeterminate
0FF8EH	Serial send shift register: UART	TxS		W	–	o	–	Indeterminate
0FF90H	Baud rate generator control register	BRGC		W	–	o	–	00H
0FFC0H	Standby control register	STBC		R/W	–	o	–	0000 x 000B
0FFC4H	Memory expansion mode register	MM		R/W	o	o	–	20H
0FFC5H	Programmable wait control register	PW		R/W	o	o	–	80H
0FFC6H	Refresh mode register	RFM		R/W	o	o	–	00H
0FFE0H	Interrupt request flag register L	IF0L	IF0	R/W	o	o	o	0000H
0FFE1H	Interrupt request flag register H	IF0H		R/W	o	o		0000H
0FFE4H	Interrupt mask flag register L	MK0L	MK0	R/W	o	o	o	FFFFH
0FFE5H	Interrupt mask flag register H	MK0H		R/W	o	o		FFFFH
0FFE8H	Priority specification flag register L	PR0L	PR0	R/W	o	o	o	FFFFH
0FFE9H	Priority specification flag register H	PR0H		R/W	o	o		FFFFH
0FFECH	Interrupt service mode specification flag register L	ISM0L	ISM0	R/W	o	o	o	0000H
0FFEDH	Interrupt service mode specification flag register H	ISM0H		R/W	o	o		0000H
0FFF4H	External interrupt mode register 0	INTM0		R/W	o	o	–	00H
0FFF5H	External interrupt mode register 1	INTM1		R/W	o	o	–	00H
0FFF8H	Interrupt status register	IST		R/W	o	o	–	00H

Input/Output Ports

Port 0 is a byte programmable tristate output port. Port 2 is bit selectable as input or control pins. Port 3 is bit programmable as input, output, or control pins. Port 4 is byte programmable as an I/O port or as the external address/ data bus. Port 5 is bit programmable as I/O or the upper address byte. Port 6 is bit programmable as I/O, control pins, or the extended address nibble. Port 7 is an input only port.

Real-Time Output Port

The real-time output port (figure 3) shares pins with port 0. The high and low nibbles may be treated separately or together. In the real-time output function, data stored beforehand in the buffer register is transferred to the output latch simultaneously with the generation of either a timer interrupt or external interrupt. Using the real-time output function in conjunction with the macroservice function enables port 0 to output preprogrammed patterns at preprogrammed variable time intervals.

A/D Converter

The μPD7821x A/D converter (figure 4) uses the successive-approximation method of converting any or all of the eight multiplexed analog inputs into 8-bit digital data. This data is stored in a result register that can be accessed at any time. The conversion time is 30 μs at 12-MHz operation. Quantization error is ±1/2 LSB; maximum full-scale error is 0.4%.

There are two methods for starting the A/D conversion operation. Conversion may be started by hardware by using an external interrupt as a trigger. The second method of starting conversion is with a software command.

There are also two methods by which the μPD7821x will operate after conversion has begun. The first, the scan method, selects several analog input signals sequentially and obtains data from each pin producing an interrupt with each conversion. The converted data can be successively transferred to memory by using the macroservice function. The second, the select mode, chooses any one input and the result is updated continuously, with or without interrupt generation depending on the chosen start method.

Figure 3. Real-Time Output Port

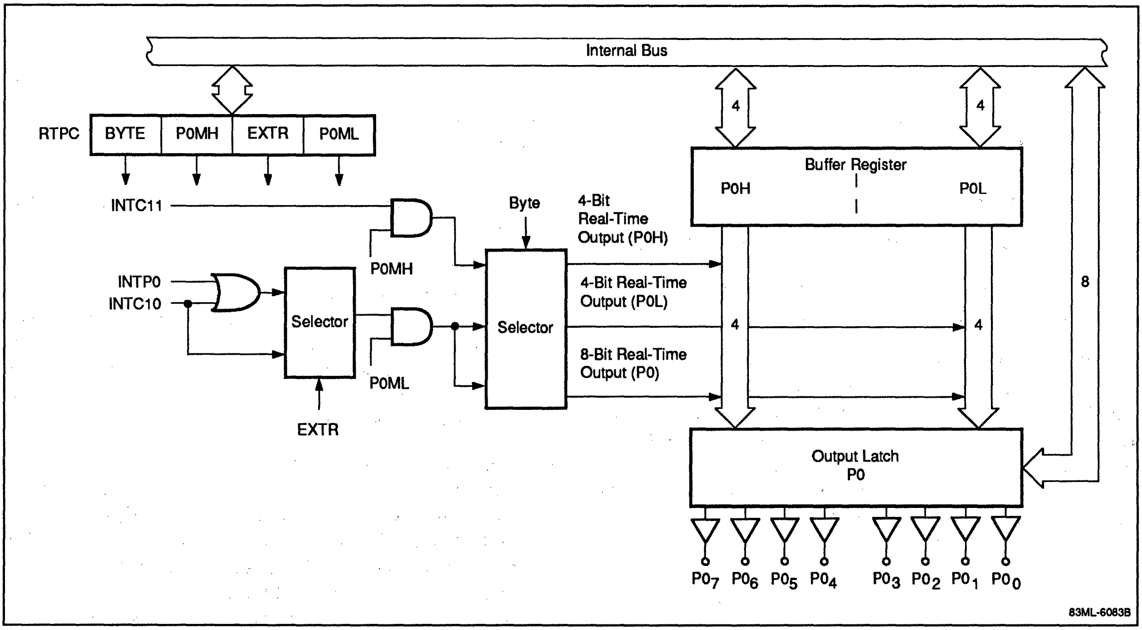
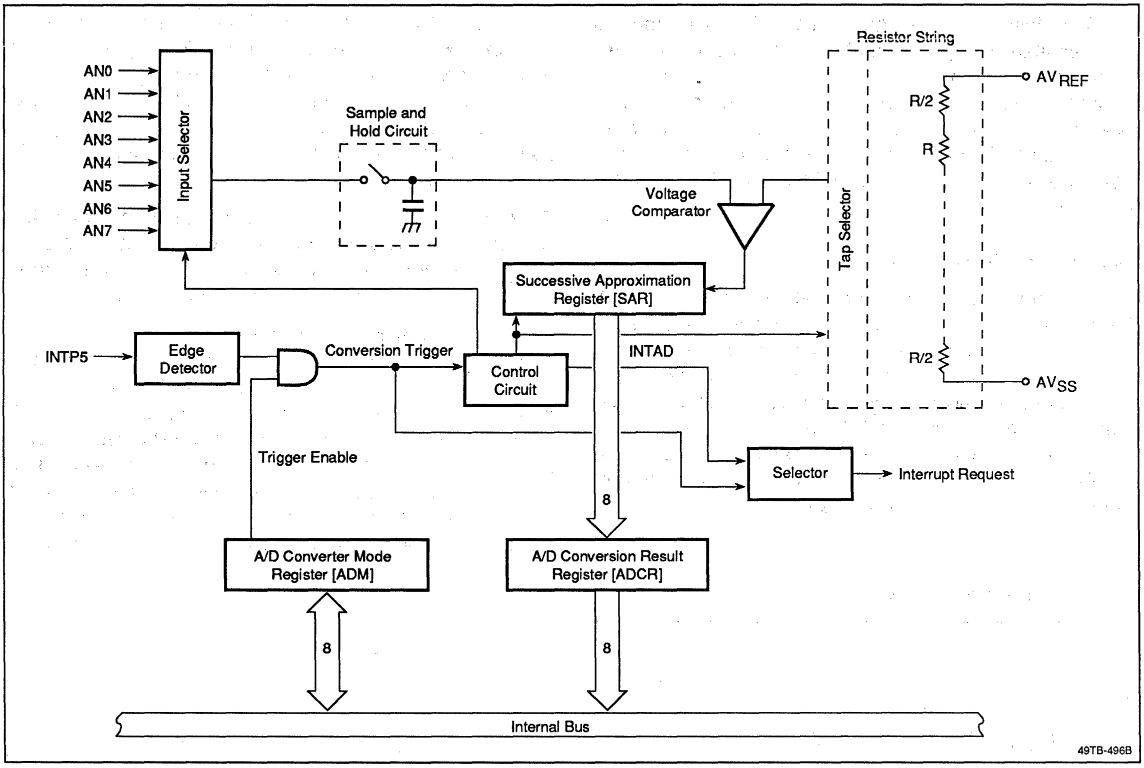


Figure 4. Analog-to-Digital Converter



Serial Interface

The μPD7821x has two independent serial interfaces.

- Asynchronous serial interface (UART) (figure 5)
- Clock-synchronized serial interface (figure 6)

A universal asynchronous receiver transmitter (UART) is used as an asynchronous serial interface. This interface transfers one byte of data following a start bit. The μPD7821x contains a baud rate generator. This allows data to be transferred over a wide range of transfer rates. Transfer rates may also be defined by dividing the clock input to the ASCK pin. Transfer rates may also be generated by 8-bit timer counter 3.

The clock-synchronized serial interface has two different modes of operation:

- Three-line serial I/O mode.
  In this mode, data 8 bits long is transferred along three lines: a serial clock (SCK) line and two serial bus lines (SO and SI). This mode is convenient when the μPD7821x is connected to peripheral I/Os and display controllers that have the conventional clock-synchronized serial interface.
- Serial bus interface mode (SBI).
  In this mode the μPD7821x can communicate data with several devices using the serial clock (SCK) and the serial data bus (SBO) lines. This mode conforms to NEC's serial bus format. In SBI mode, addresses that select a device to communicate with, commands that direct the device, and actual data are output to the serial data bus. A handshake line, which was required for connecting several devices in the conventional clock-synchronized serial interface, is not needed.

Figure 5. Asynchronous Serial Interface

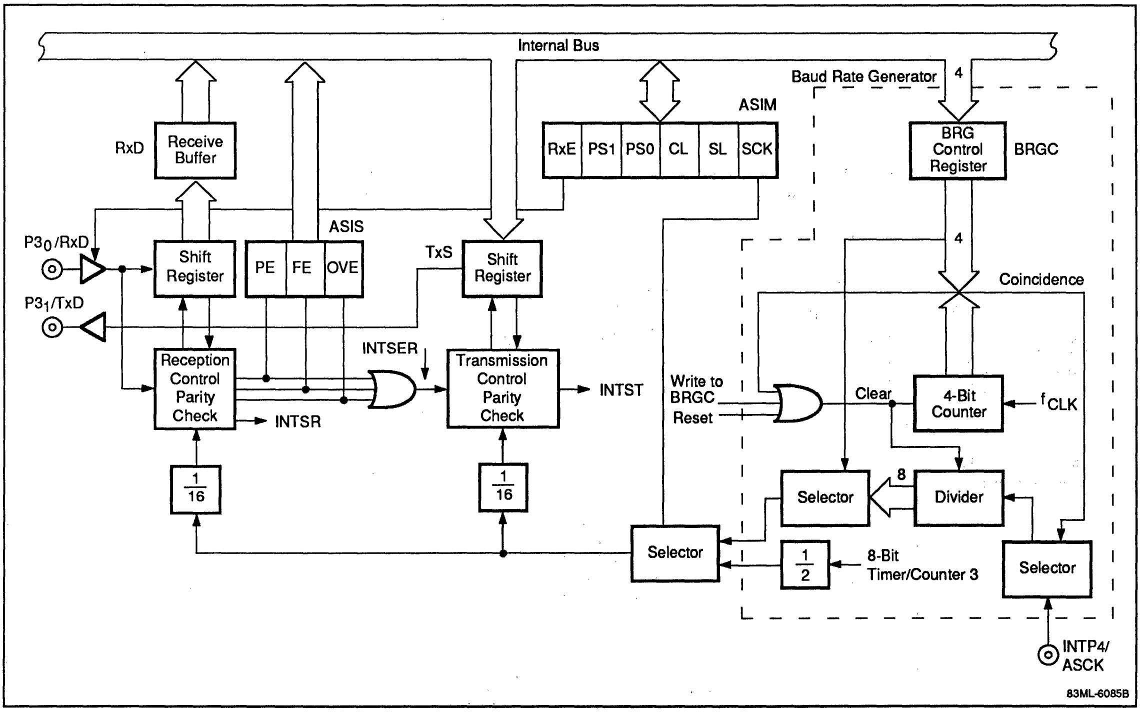
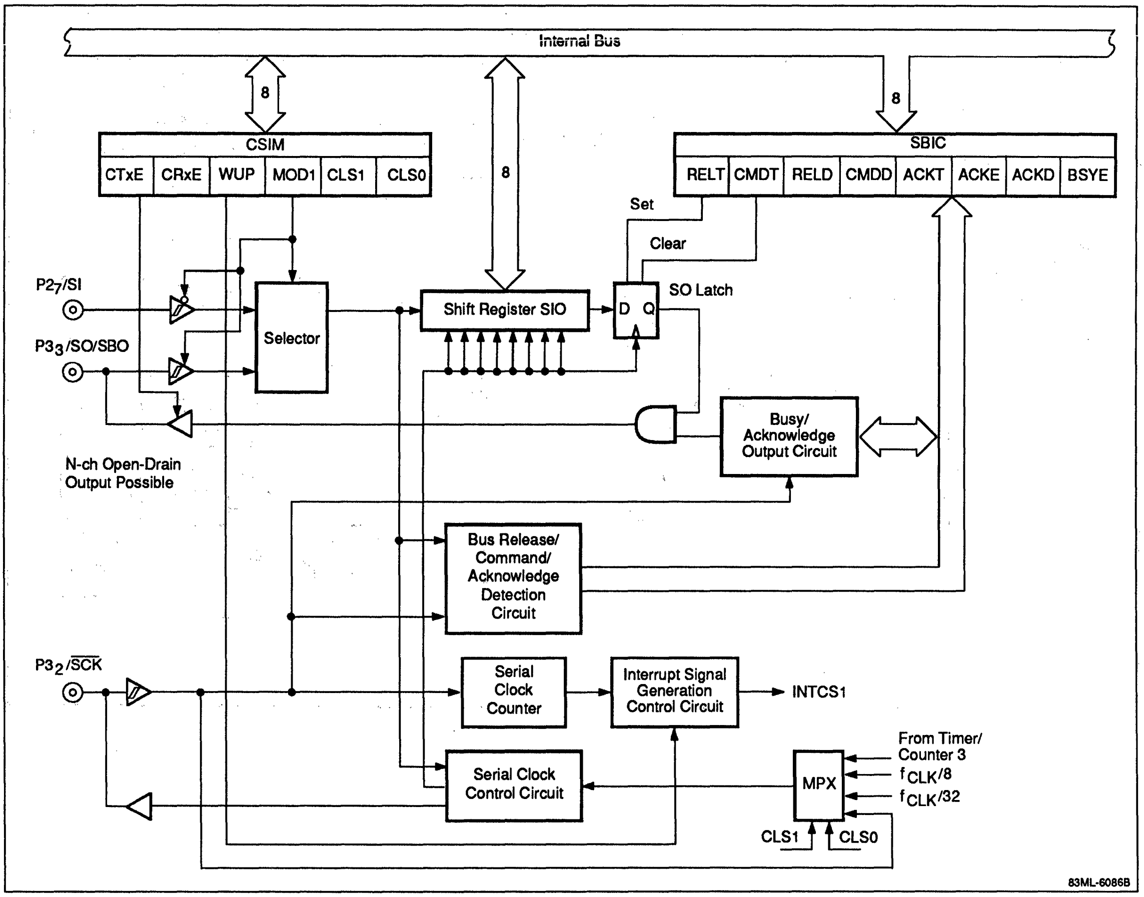


Figure 6. Clock-Synchronized Serial Interface



Timer/Counters

The μPD7821X has four timer/counters: one 16-bit and three 8-bit. The 16-bit timer/counter (figure 7) has the basic functionality of an interval timer, a programmable square-wave output, and a pulse width measurer. These functions can provide a digital delayed one-shot output, a pulse width modulated output, and a cycle measurer.

The first two 8-bit timer/counters can provide the basic functions of an interval timer and a pulse width measurer. Timer/counter 1 can also be used as a timer for output trigger generation for the real-time output port. Timer/counter 2 can also provide an external event counter, a one-shot timer, a programmable square-wave output, a pulse-width modulated output, and a cycle measurer. Timer/counter 3 can operate as an internal timer or as a counter to generate clocks for a baud rate generator. See figures 8, 9, and 10.

Figure 7. 16-Bit Timer/Counter

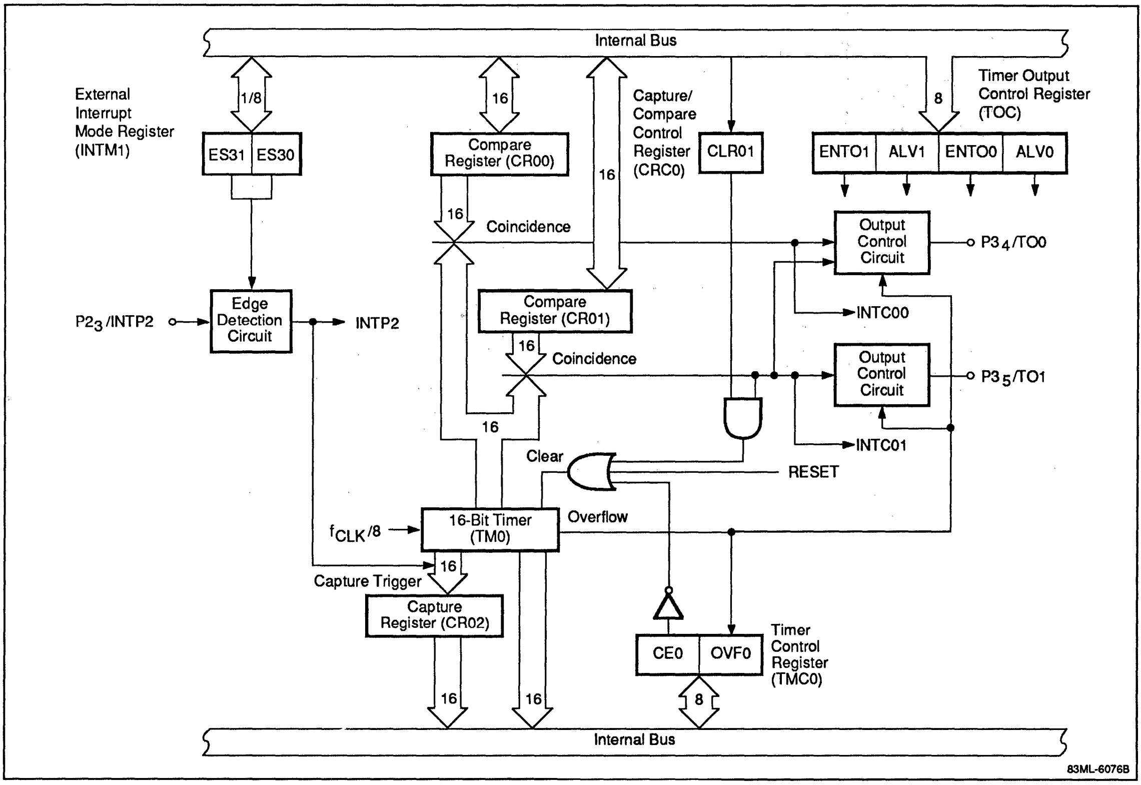


Figure 8. 8-Bit Timer/Counter 1

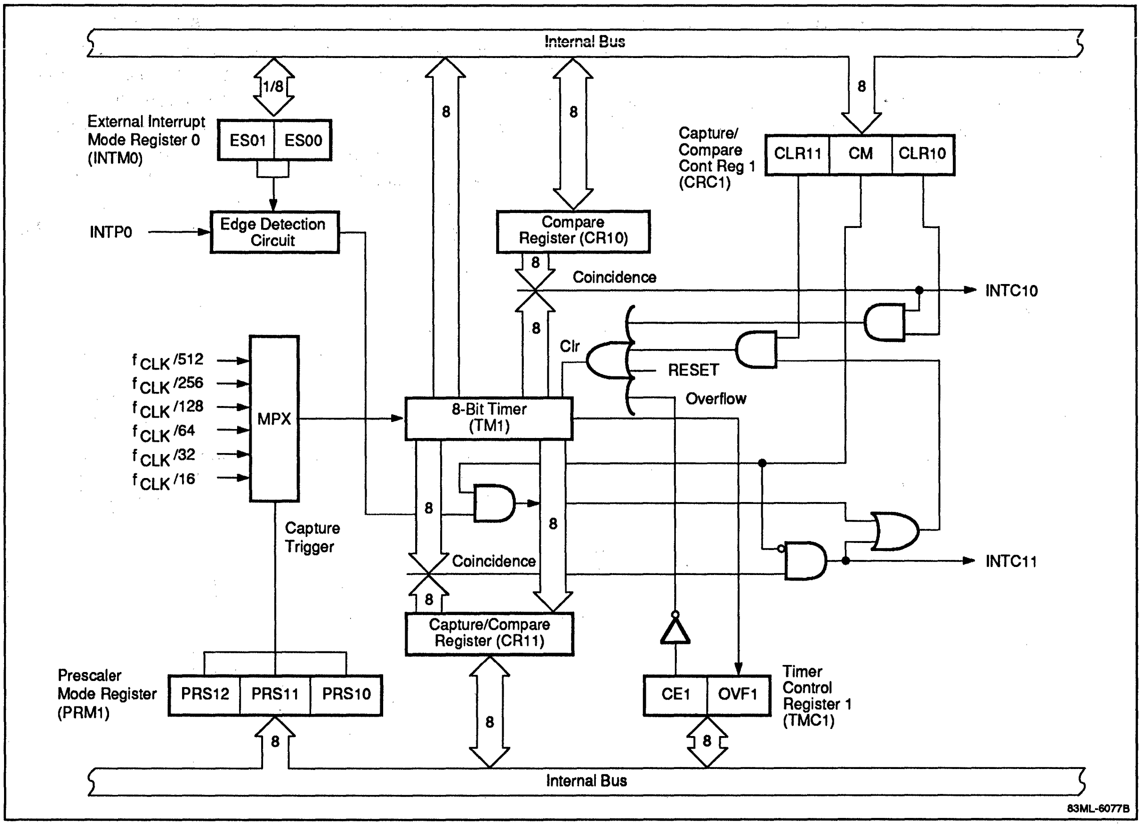
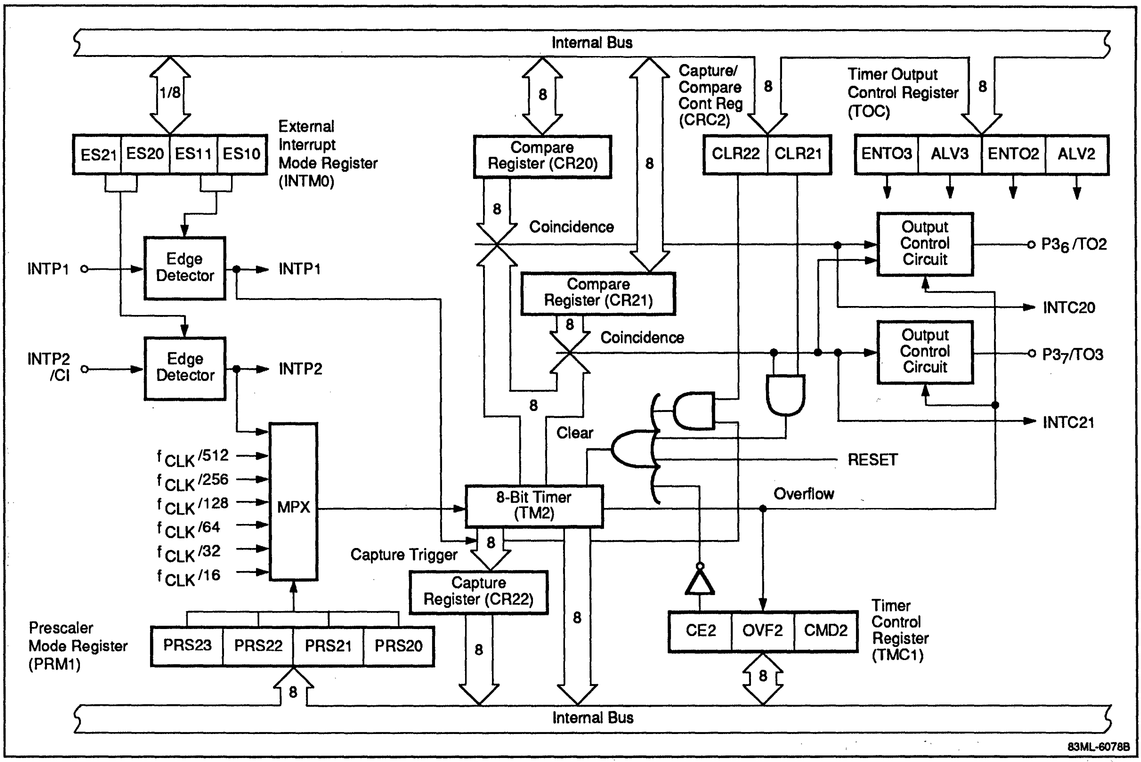


Figure 9. 8-Bit Timer/Counter 2



Interrupts

There are 20 interrupt request sources; each source is allocated a location in the vector table. (See table 2.) There is one software interrupt request and one of the remaining 19 interrupts is non-maskable. The software interrupt and the non-maskable interrupt are unconditionally received even in the DI state. These two interrupts possess the maximum priority. The maskable interrupt requests are subject to mask control by the setting of the interrupt mask flag.

There are default priorities associated with each maskable interrupt and these can be assigned to either of two programmable priority levels. Interrupts may be serviced by the vectored interrupt method where a branch to a desired service program is executed. Interrupts may also be handled by the macroservice function where a preassigned process is performed without program intervention.

Figure 10. 8-Bit Timer/Counter 3

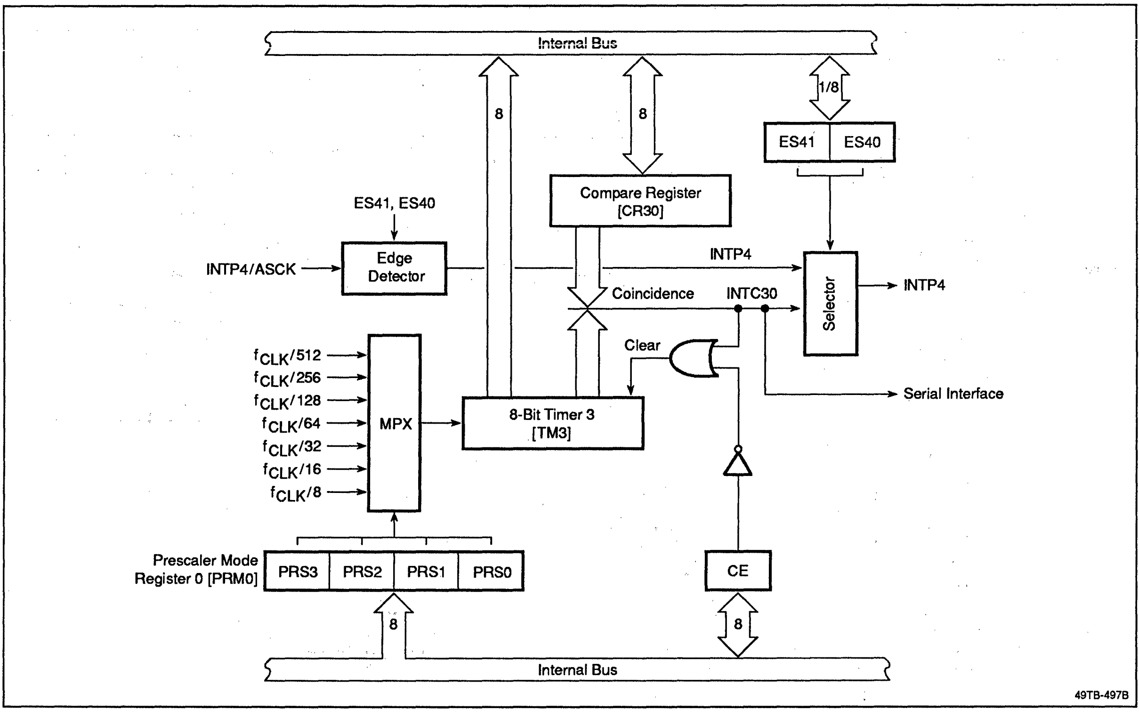


Table 2. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Generation Source	Macroservice Mode	Vector Table Address
Software	None	BRK instruction execution	–	003EH
Non-maskable	None	NMI (pin input edge detection)	–	0002H
Maskable	0	INTP0 (pin input edge detection)	Yes	0006H
	1	INTP1 (pin input edge detection)	Yes	0008H
	2	INTP2 (pin input edge detection)	Yes	000AH
	3	INTP3 (pin input edge detection)	Yes	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	Yes	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	Yes	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	Yes	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	Yes	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	Yes	001CH
	9	INTP4 (pin input edge detection)/INTC30 (TM3-CR30 coincidence signal generation)	Yes	000EH
	10	INTP5 (pin input edge detection)/INTAD (end of A/D conversion)	Yes	0010H
	11	INTC20 (TM2-CR20 coincidence signal generation)	Yes	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	–	0020H
	13	INTSR (end of asynchronous serial interface reception)	Yes	0022H
	14	INTST (end of asynchronous serial interface transmission)	Yes	0024H
	15	INTCSI (end of clocked serial interface transmission)	Yes	0026H

Macroservice

The macroservice function can be programmed to transfer data from a special function register to memory or from memory to a special function register. Transfer events are triggered by interrupt requests and take place without software intervention. There are 17 interrupt requests where macroservicing can be executed. The macroservice function is controlled by the macroservice mode register and the macroservice channel pointer. The macroservice mode register assigns the macroservicing mode and the macroservice channel pointer indicates the address of the memory location pointers. The location of each register and its corresponding interrupt is shown in figure 11.

Refresh

The refresh signal is used with a pseudostatic RAM. The refresh cycle can be set to one of four intervals ranging from 2.6 to 21.3 μs. The refresh is timed to follow a read or write operation so there is no interference.

Standby Modes

Halt and stop functions reduce system power consumption. In the halt mode, the CPU stops and the system clock continues to run. A release of the halt mode is initiated by an unmasked interrupt request, an NMI, or a $\overline{\text{RESET}}$ input. In the stop mode, the CPU and system clock are both stopped, reducing the power consumption even further. The stop mode is released by an NMI input or a $\overline{\text{RESET}}$ input.

Figure 11. Macroservice Control Word Map

0FEDFH	Channel Pointer	INTSR
0FEDEH	Mode Register	
0FEDDH	Channel Pointer	INTST
0FEDCH	Mode Register	
0FEDBH	Channel Pointer	INTCSI
0FEDAH	Mode Register	
0FED9H	Channel Pointer	INTC10
0FED8H	Mode Register	
0FED7H	Channel Pointer	INTC11
0FED6H	Mode Register	
0FED5H	Channel Pointer	INTP4/INTC30
0FED4H	Mode Register	
0FED3H	Channel Pointer	INTP5/INTAD
0FED2H	Mode Register	
0FED1H	Channel Pointer	INTC00
0FED0H	Mode Register	
0FECFH	Channel Pointer	INTC01
0FECEH	Mode Register	
0FECDH	Channel Pointer	INTC20
0FECCH	Mode Register	
0FECBH	Channel Pointer	INTC21
0FECAH	Mode Register	
0FEC9H	Channel Pointer	INTP0
0FEC8H	Mode Register	
0FEC7H	Channel Pointer	INTP1
0FEC6H	Mode Register	
0FEC5H	Channel Pointer	INTP2
0FEC4H	Mode Register	
0FEC3H	Channel Pointer	INTP3
0FEC2H	Mode Register	

83ML-6087A

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

$T_A = +25°C$.

Item	Symbol	Conditions	Rating	Unit
Operating voltage	V_{DD}		–0.5 to +7.0	V
	AV_{REF}		–0.5 to V_{DD}	V
	AV_{SS}		–0.5 to +0.5	V
Input voltage	V_{I1}	Note 1	–0.5 to V_{DD} + 0.5	V
	V_{I2}	Note 2	–0.5 to AV_{REF} + 0.5	V
	V_{I3}	Note 3; for μPD78P214	–0.5 to +13.5	V
Output voltage	V_O		–0.5 to V_{DD} + 0.5	V
Low-level output current	I_{OL}	One output pin	30 (peak)	mA
			15 (mean value)	mA
		All output pins total	150 (peak)	mA
			100 (mean value)	mA
High-level output current	I_{OH}	One output pin	–2	mA
		All output pins total	–50	mA
Operating temperature	T_{OPT}		–40 to +85	°C
Storage temperature	T_{STG}		–65 to +150	°C

Notes:

(1) Pins $P7_0$-$P7_5$/AN0-AN5, $P6_6$/$\overline{WAIT}$/AN6, and $P6_7$/$\overline{REFRQ}$/AN7 except when Note 2 is applicable.

(2) Pin used as the A/D converter input or pin selected by bits ANI0-ANI2 of the ADM register when the A/D converter is not in operation.

(3) $P2_0$/NMI, $\overline{EA}$/V_{PP}, and $P2_1$/INTP0/A_9 pins in the PROM programming mode.

Operating Frequency

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	–40 to +85°C	+5V ± 10%

Capacitance

$T_A = +25°C$; $V_{DD} = V_{SS} = 0$ V.

Item	Symbol	Typ	Max	Unit	Conditions
Input capacitance	C_I		20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O		20	pF	
Input/output capacitance	C_{IO}		20	pF	

DC Characteristics

T_A = −40 to +85°C; V_{DD} = +5 V ±10%; V_{SS} = 0 V.

Item	Symbol	Conditions		Min	Typ	Max	Unit
Low-level input voltage	V_{IL}			0		0.8	V
High-level input voltage	V_{IH1}	Except the specified pins (Notes 1, 2)		2.2		V_{DD}	V
	V_{IH2}	Specified pins (Note 1)		2.2		AV_{REF}	V
	V_{IH3}	Specified pins (Note 2)		0.8 V_{DD}		V_{DD}	V
Low-level output voltage	V_{OL1}	I_{OL} = 2.0 mA				0.45	V
	V_{OL2}	I_{OL} = 8.0 mA (Note 3)				1.0	V
High-level output voltage	V_{OH1}	I_{OH} = −1.0 mA		V_{DD} − 1.0			V
	V_{OH2}	I_{OH} = −100 μA		V_{DD} − 0.5			V
	V_{OH3}	I_{OH} = −5.0 mA (Note 4)		2.0			V
Input leakage current	I_{LI}	0V ≤ V_I ≤ V_{DD}				±10	μA
Output leakage current	I_{LO}	0V ≤ V_O ≤ V_{DD}				±10	μA
AV_{REF} current	AI_{REF}	Operating mode, f_{XX} = 12 MHz			1.5	5.0	mA
V_{DD} power supply current	I_{DD1}	Operating mode, f_{XX} = 12 MHz			20	40	mA
	I_{DD2}	HALT mode, f_{XX} = 12 MHz			7	20	mA
Data retention voltage	V_{DDDR}	STOP mode		2.5		5.5	V
Data retention current	I_{DDDR}	STOP mode	V_{DDDR} = 2.5V		2	20	μA
			V_{DDDR} = 5V ±10%		5	50	μA
Pullup resistor	R_L	V_I = 0 V		15	40	80	kΩ

Notes:

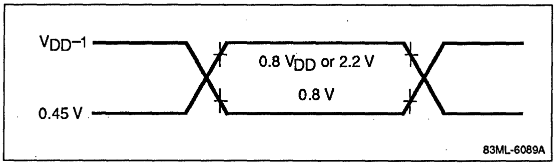
(1) Pins $P7_0$-$P7_5$/AN0-AN5, $P6_6$/WAIT/AN6, and $P6_7$/$\overline{\text{REFRQ}}$/AN7 when the pin is used as the A/D converter input or is selected by bits ANI0-ANI2 of the ADM register when the A/D converter is not in operation.

(2) X1, X2, $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/$\overline{\text{CI}}$, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/$\overline{\text{SCK}}$, $P3_3$/SO/SBO, and EA pins.

(3) Pins $P4_0$-$P4_7$/AD_0-AD_7 and $P5_0$-$P5_7$/A_8-A_{15}.

(4) Pins $P0_0$-$P0_7$.

Figure 12. Voltage Thresholds for Timing Measurements



Read/Write Operation

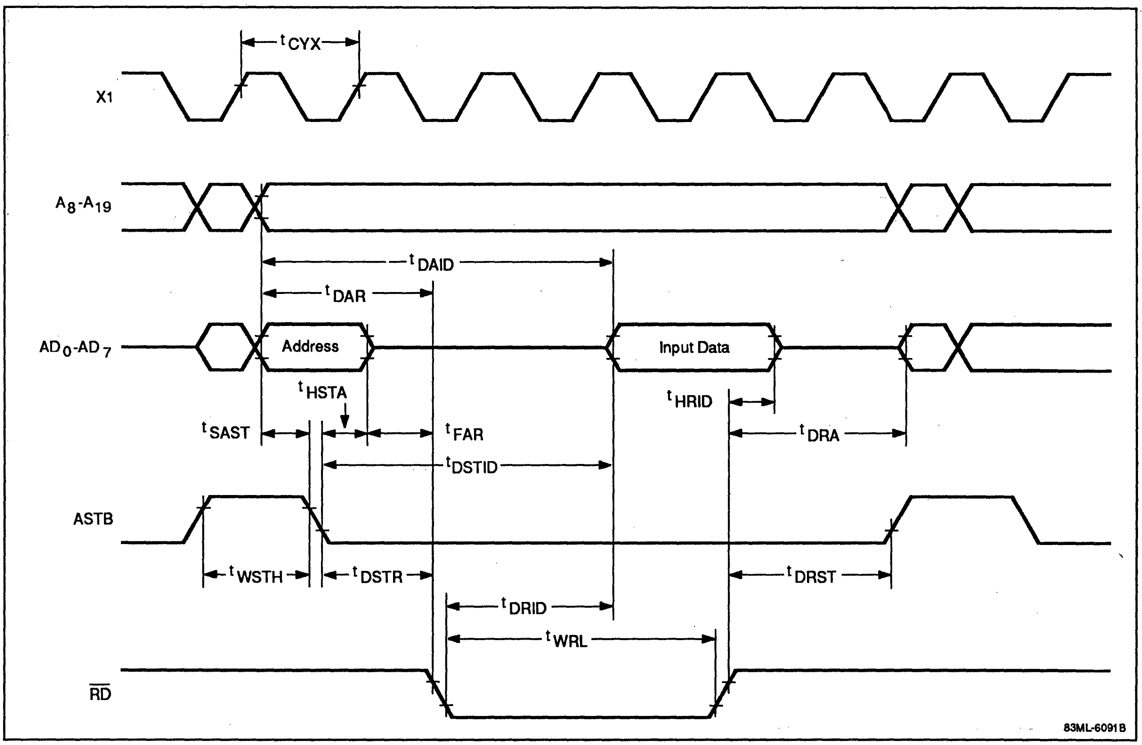
$T_A = -40$ to $+85°C$; $V_{DD} = +5\ V \pm 10\%$; $V_{SS} = 0\ V$; $f_{XX} = 12\ MHz$; $C_L = 100\ pF$.

Item	Symbol	Conditions	Min	Max	Unit
X1 input clock cycle time	t_{CYX}		82	250	ns
Address setup time to ASTB ↓	t_{SAST}		52		ns
Address hold time from ASTB ↓ (Note 1)	t_{HSTA}	$R_L = 5\ k\Omega$, $C_L = 50\ pF$	25		ns
Address to $\overline{RD}$ ↓ delay time	t_{DAR}		129		ns
Address float time from $\overline{RD}$ ↓	t_{FAR}		11		ns
Address to data input time	t_{DAID}			228	ns
ASTB ↓ to data input time	t_{DSTID}			181	ns
$\overline{RD}$ ↓ to data input time	t_{DRID}			99	ns
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}		52		ns
Data hold time from $\overline{RD}$ ↑	t_{HRID}		0		ns
$\overline{RD}$ ↑ to address active time	t_{DRA}		124		ns
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}		124		ns
$\overline{RD}$ low-level width	t_{WRL}		124		ns
ASTB high-level width	t_{WSTH}		52		ns
Address to $\overline{WR}$ ↓ delay time	t_{DAW}		129		ns
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}			142	ns
$\overline{WR}$ ↓ to data output time	t_{DWOD}			60	ns
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}		52		ns
	t_{DSTW2}	Refresh mode	129		ns
Data setup time to WR ↑	t_{SODWR}		146		ns
Data setup time to $\overline{WR}$ ↓ (Note 1)	t_{SODWF}	Refresh mode	22		ns
Data hold time from $\overline{WR}$ ↑	t_{HWOD}		20		ns
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}		42		ns
$\overline{WR}$ low-level width	t_{WWL1}		196		ns
	t_{WWL2}	Refresh mode	114		ns
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}			146	ns
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}			84	ns
$\overline{WAIT}$ hold time from X1 ↓	t_{HWTX}		0		ns
$\overline{WAIT}$ setup time to X1 ↑	t_{SWTX}		0		ns

Notes:

(1) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: $C_L = 100\ pF$ and $R_L = 2\ k\Omega$.

Figure 13. Read Operation Timing



6

Figure 14. Write Operation Timing

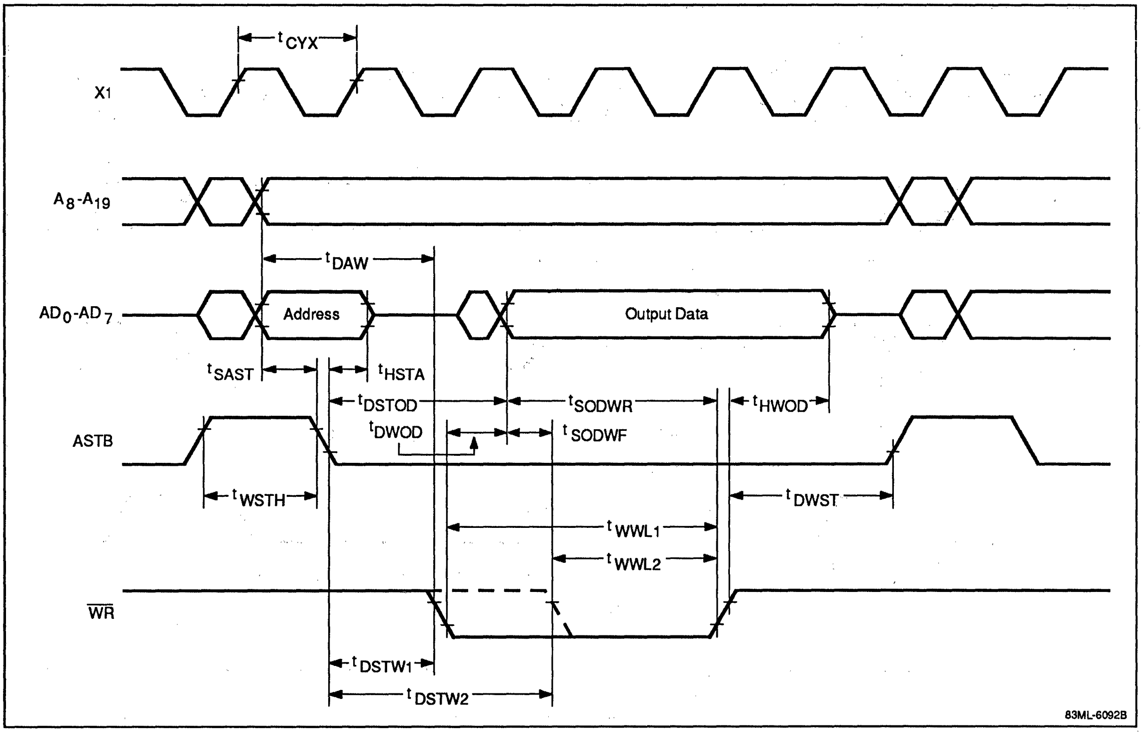
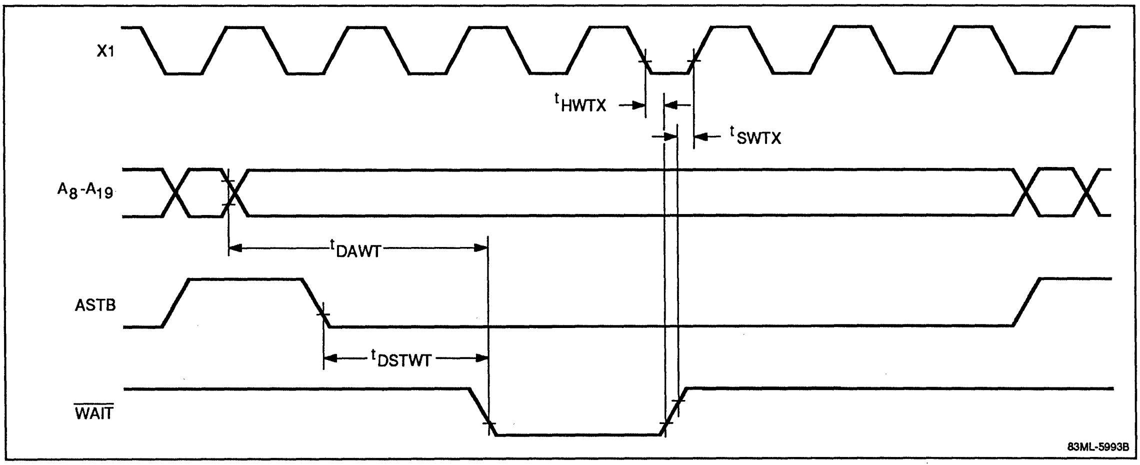


Figure 15. External $\overline{WAIT}$ Input Timing



Serial Port Operation

T_A = –40 to +85°C; V_{DD} = +5 V ± 10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF.

Item	Symbol	Conditions		Min	Max	Unit
Serial clock cycle time	t_{CYSK}	Input	External clock	1.0		µs
		Output	Internal clock/16	1.3		µs
			Internal clock/64	5.3		µs
Serial clock low-level width	t_{WSKL}	Input	External clock	420		ns
		Output	Internal clock/16	556		ns
			Internal clock/64	2.5		µs
Serial clock high-level width	t_{WSKH}	Input	External clock	420		ns
		Output	Internal clock/16	556		ns
			Internal clock/64	2.5		µs
SI, SBO setup time to $\overline{\text{SCK}}$ ↑	t_{SSSK}			150		ns
SI, SBO hold time from $\overline{\text{SCK}}$ ↓	t_{HSSK}			400		ns
SO/SBO output delay time from $\overline{\text{SCK}}$ ↑	t_{DSBSK1}	CMOS push-pull output (3-line serial I/O mode)		0	300	ns
	t_{DSBSK2}	Open-drain output (SBI mode), R_L = 1 kΩ		0	800	ns
SBO high, hold time from $\overline{\text{SCK}}$ ↑	t_{HSBSK}	SBI mode		4		t_{CYX}
SBO low, setup time to $\overline{\text{SCK}}$ ↓	t_{SSBSK}	SBI mode		4		t_{CYX}
SBO low-level width	t_{WSBL}			4		t_{CYX}
SBO high-level width	t_{WSBH}			4		t_{CYX}
RxD setup time to $\overline{\text{SCK}}$ ↑	t_{SRXSK}			80		ns
RxD hold time after $\overline{\text{SCK}}$ ↑	t_{HSKRX}			80		ns
$\overline{\text{SCK}}$ ↓ to TxD delay time	t_{DSKTX}				210	ns

Figure 16. Three-Line Serial I/O Timing

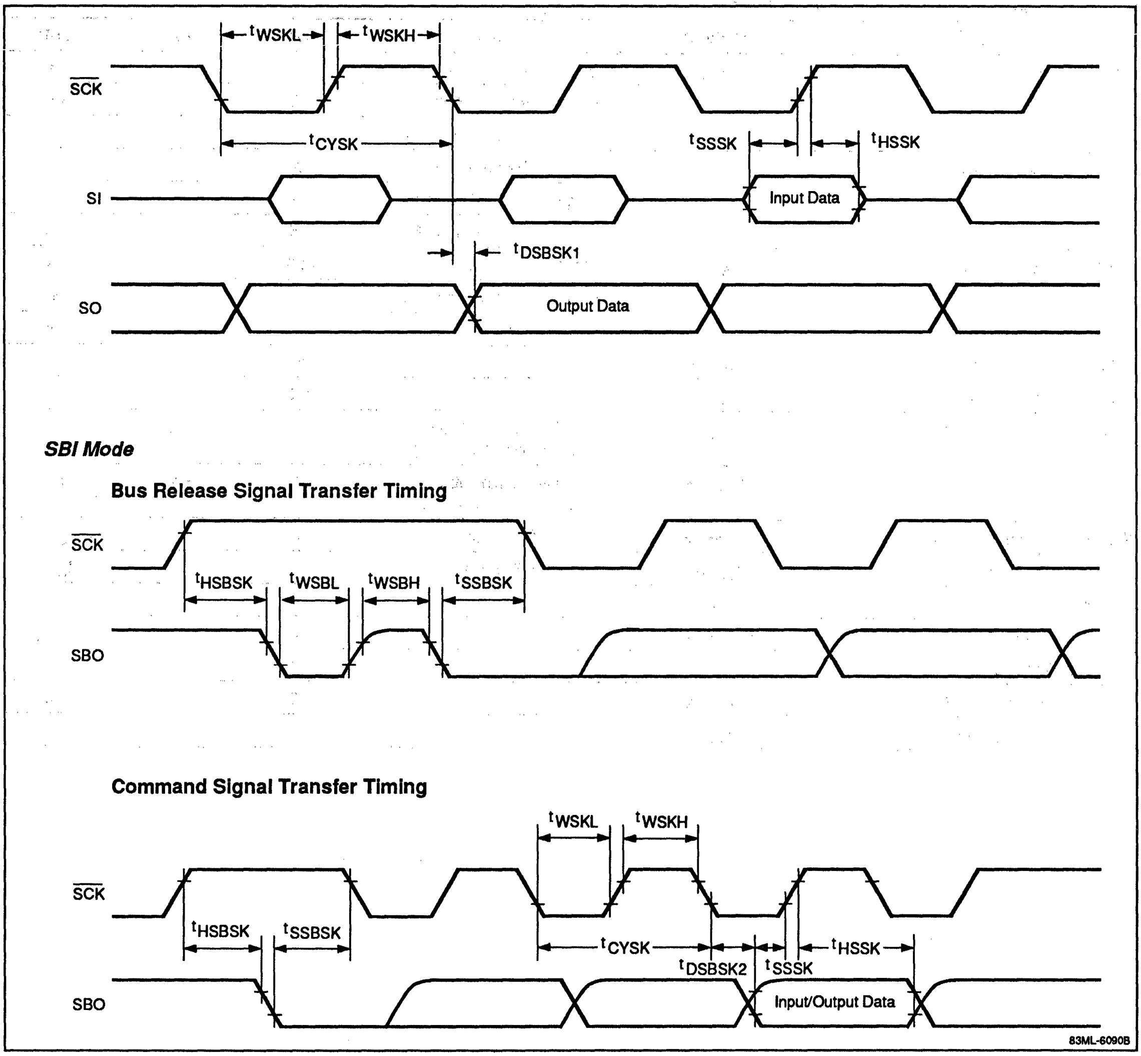
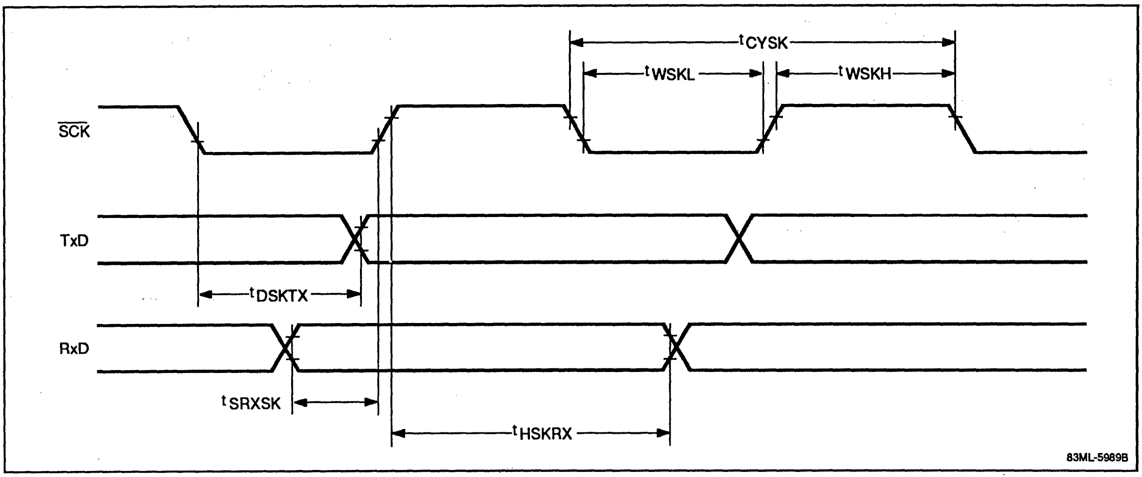


Figure 17. Asynchronous Mode Timing



A/D Converter Operation

$T_A = -40$ to $+85°C$; $V_{DD} = +5\ V \pm 10\%$; $V_{SS} = AV_{SS} = 0\ V$.

Item	Symbol	Conditions	Min	Typ	Max	Unit
Resolution			8			Bit
Full-scale error		$AV_{REF} = 4.0\ V$ to V_{DD}; $T_A = -10$ to $+70°C$			0.4	%
		$AV_{REF} = 3.4\ V$ to V_{DD}; $T_A = -10$ to $+70°C$			0.8	%
		$AV_{REF} = 4.0\ V$ to V_{DD}			0.8	%
Quantization error					±1/2	LSB
Conversion time	t_{CONV}	$82\ ns \leqq t_{CYX} \leqq 125\ ns$	360			t_{CYX}
		$125\ ns \leqq t_{CYX} \leqq 250\ ns$	240			t_{CYX}
Sampling time	t_{SAMP}	$82\ ns \leqq t_{CYX} \leqq 125\ ns$	72			t_{CYX}
		$125\ ns \leqq t_{CYX} \leqq 250\ ns$	48			t_{CYX}
Analog input voltage	V_{IAN}		0		AV_{REF}	V
Input impedance	R_{AN}			1000		MΩ
Analog reference voltage	AV_{REF}		3.4		V_{DD}	V
AV_{REF} current	AI_{REF}	Operating mode, $f_{XX} = 12\ MHz$		1.5	5.0	mA
		STOP mode		0.2	1.5	mA

6

Interrupt Timing Operation

Item	Symbol	Conditions	Min	Max	Unit
NMI low-level width	t_{WNIL}		10		µs
NMI high-level width	t_{WNIH}		10		µs
INTP0-INTP5 low-level width	t_{WITL}		24		t_{CYX}
INTP0-INTP5 high-level width	t_{WITH}		24		t_{CYX}
$\overline{\text{RESET}}$ low-level width	t_{WRSL}		10		µs
$\overline{\text{RESET}}$ high-level width	t_{WRSH}		10		µs

Figure 18. Interrupt Input Timing

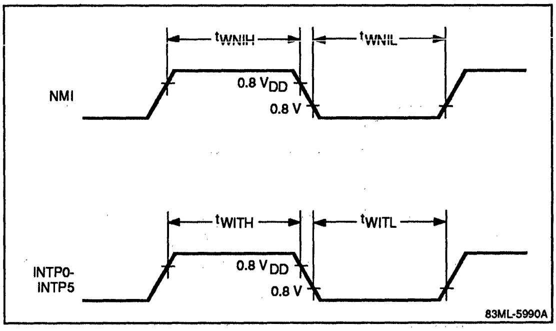
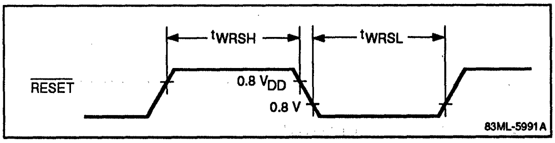


Figure 19. Reset Input Timing



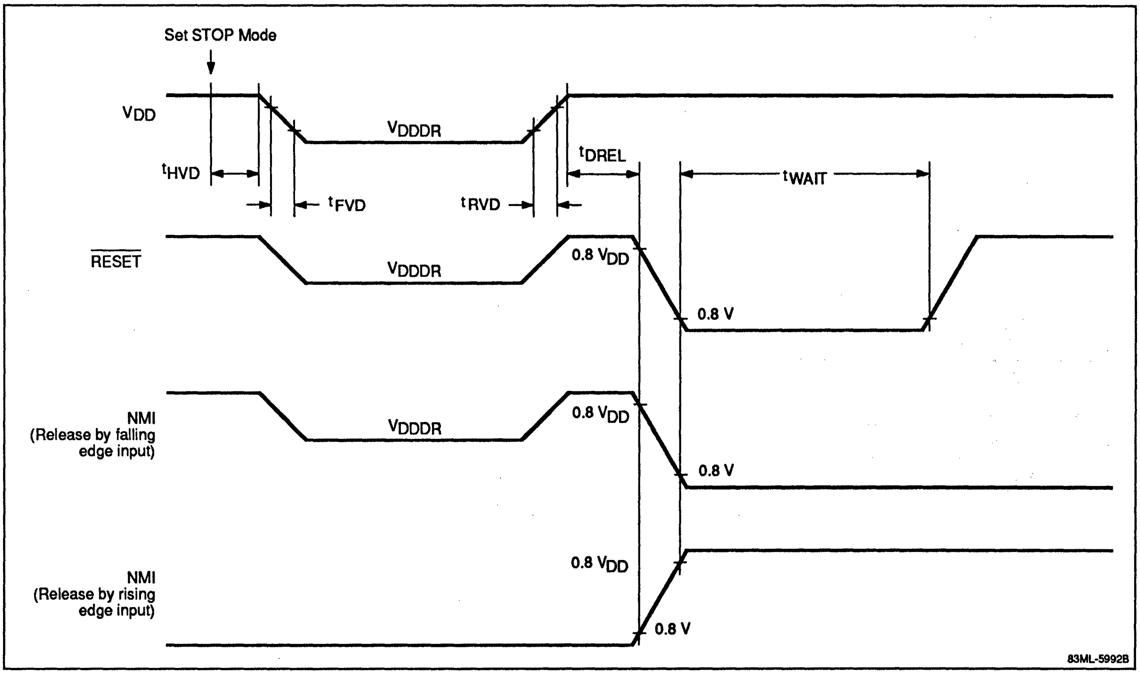
Data Retention Characteristics

Item	Symbol	Conditions	Min	Typ	Max	Unit
Data retention voltage	V_{DDDR}	STOP mode	2.5		5.5	V
Data retention current	I_{DDDR}	V_{DDDR} = 2.5 V		2	15	µA
		V_{DDDR} = 5 V ±10%		5	20	µA
V_{DD} rise time	t_{RVD}		200			µs
V_{DD} fall time	t_{FVD}		200			µs
V_{DD} retention time (for STOP mode setting)	t_{HVD}		0			ms
STOP release signal input time	t_{DREL}		0			ms
Oscillation stabilization wait time	t_{WAIT}	Crystal oscillator	30			ms
		Ceramic resonator	5			ms
Low-level input voltage	V_{IL}	Specified pins (Note 1)	0		0.1 V_{DDDR}	V
High-level input voltage	V_{IH}		0.9 V_{DDDR}		V_{DDDR}	V

Notes:

(1) $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/$\overline{\text{CI}}$, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/$\overline{\text{SCK}}$, $P3_3$/SO/SB0, and $\overline{\text{EA}}$ pins.

Figure 20. Data Retention Characteristics



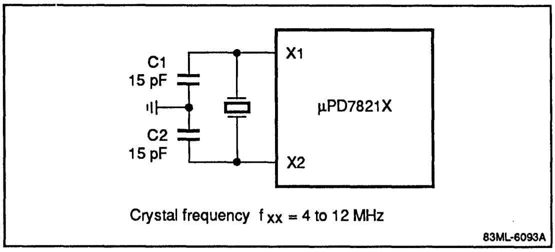
Timing Dependent on t_{CYX}

Item	Symbol	Calculation Formula	Min/Max	12 MHz	Unit
X1 input clock cycle time	t_{CYX}		Min	82	ns
Address setup time to ASTB ↓	t_{SAST}	$t_{CYX}-30$	Min	52	ns
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX}-35$	Min	129	ns
Address float time from $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2-30$	Min	11	ns
Address to data input time	t_{DAID}	$(4+2n)\,t_{CYX}-100$	Max	228	ns
ASTB ↓ to data input time	t_{DSTID}	$(3+2n)\,t_{CYX}-65$	Max	181	ns
$\overline{RD}$ ↓ to data input time	t_{DRID}	$(2+2n)\,t_{CYX}-65$	Max	99	ns
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	$t_{CYX}-30$	Min	52	ns
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX}-40$	Min	124	ns
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}-40$	Min	124	ns
$\overline{RD}$ low-level width	t_{WRL}	$(2+2n)\,t_{CYX}-40$	Min	124	ns
ASTB high-level width	t_{WSTH}	$t_{CYX}-30$	Min	52	ns
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX}-35$	Min	129	ns
ASTB ↓ to data output time	t_{DSTOD}	$t_{CYX}+60$	Max	142	ns
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	$t_{CYX}-30$	Min	52	ns
	t_{DSTW2}	$2t_{CYX}-35$ (refresh mode)	Min	129	ns
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	$(3+2n)\,t_{CYX}-100$	Min	146	ns
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	$t_{CYX}-60$ (refresh mode)	Min	22	ns
WR ↑ to ASTB ↑ delay time	t_{DWST}	$t_{CYX}-40$	Min	42	ns
WR low-level width	t_{WWL1}	$(3+2n)\,t_{CYX}-50$	Min	196	ns
	t_{WWL2}	$(2+2n)\,t_{CYX}-50$ (refresh mode)	Min	114	ns
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX}-100$	Max	146	ns
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX}-80$	Max	84	ns

Notes:

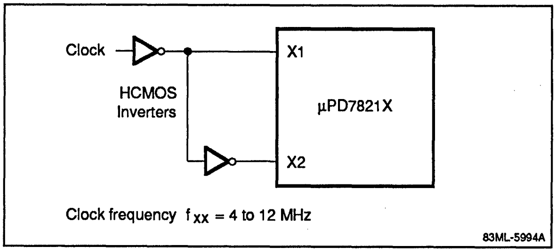
(1) n indicates the number of wait states.

Figure 21. Recommended Oscillator Circuit



Crystal frequency f_{XX} = 4 to 12 MHz

Figure 22. Recommended External Clock Circuit

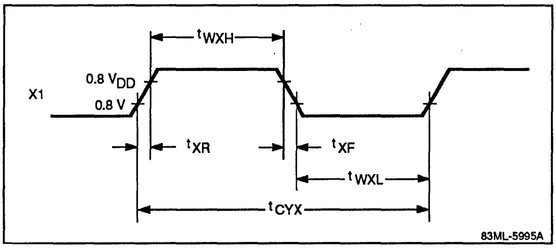


Clock frequency f_{XX} = 4 to 12 MHz

External Clock Operation

Item	Symbol	Conditions	Min	Max	Unit
X1 input low-level width	t_{WXL}		30	130	ns
X1 input high-level width	t_{WXH}		30	130	ns
X1 input rise time	t_{XR}		0	30	ns
X1 input fall time	t_{XF}		0	30	ns
X1 input clock cycle time	t_{CYX}		82	250	ns

Figure 23. External Clock Timing



μPD78P214 PROGRAMMING

In the 78P214, the mask ROM of 78214 is replaced by a one-time programmable ROM (OTP ROM) or a reprogrammable, ultraviolet erasable ROM (UV EPROM). The ROM is 16,384 × 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P214CW/GJ/GQ/L are the socket adaptors used for configuring the μPD78P214 to fit a standard PROM socket.

Refer to tables 3 through 6 and figures 24 and 25 for special information applicable to PROM programming.

Table 3. Pin Functions During EPROM Programming

Pin		Function
$P0_0$-$P0_7$	A_0-A_7	Input pins for PROM write/verify operations
$P5_0/A_8$	A_8	Input pin for PROM write/verify operation
$P2_1$/INTP0	A_9	Input pin for PROM write/verify operation
$P5_2$-$P5_6/A_{10}$-A_{14}	A_{10}-A_{14}	Input pins for PROM write/verify operations
$P4_0$-$P4_7/AD_0$-AD_7	D_0-D_7	Data pins for PROM write/verify operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobe data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enable a data read from the PROM
$P2_0$/NMI	NMI	PROM programming mode is entered by applying a high voltage to this pin
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
$\overline{EA}$	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{SS}	V_{SS}	Ground

Table 4. Summary of Operation Modes for PROM Programming

Mode	NMI	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	V_{pp}	V_{DD}	D_0-D_7
Program write	+12.5 V	L	L	H	+12.5 V	+6 V	Data input
Program verify	+12.5 V	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	+12.5 V	L	H	H	+12.5 V	+6 V	High Z
Read out	+12.5 V	L	L	L	+5 V	+5 V	Data output
Output disable	+12.5 V	L	L	H	+5 V	+5 V	High Z
Standby	+12.5 V	L	H	L/H	+5 V	+5 V	High Z

Notes:
When +12.5 V is applied to V_{pp} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

Table 5. DC Programming Characteristics

T_A = 25 ±5°C, V_{IP} = 12.5 ± 0.5 V applied to NMI pin, V_{ss} = 0 V.

Parameter	Symbol	Symbol*	Condition	Min	Typ	Max	Unit
High-level input voltage	V_{IH}	V_{IH}		2.4		V_{DDP} +0.3	V
Low-level input voltage	V_{IL}	V_{IL}		–0.3		0.8	V
Input leakage current	V_{LIP}	V_{LI}	$0 \le V_I \le V_{DDP}$			10	μA
High-level output voltage	V_{OH1}	V_{OH}	I_{OH} = –400 μA	2.4			V
	V_{OH2}	V_{OH2}	I_{OH} = –100 μA	V_{DD}–0.7			V
Low-level output voltage	V_{OL}	V_{OL}	I_{OH} = 2.1 mA			0.45	V
Output leakage current	I_{LO}		$0 \le V_O \le V_{DPP}$, $\overline{OE} = V_{IH}$			10	μA
NMI pin high-voltage input current	I_{IP}					±10	μA
V_{DDP} power voltage	V_{DDP}	V_{CC}	Program memory write mode	5.75	6.0	6.25	V
			Program memory read mode	4.5	5.0	5.5	V
V_{PP} power voltage	V_{PP}	V_{PP}	Program memory write mode	12.2	12.5	12.8	V
			Program memory read mode		$V_{PP} = V_{DDP}$		V
V_{DDP} power current	I_{DD}	I_{CC}	Program memory write mode		5	30	mA
			Program memory read mode $\overline{CE} = V_{IL}$, $V_I = V_{IH}$		5	30	mA
V_{PP} power current	I_{PP}	I_{PP}	Program memory write mode $\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$		5	30	mA
			Program memory read mode		1	100	μA

* Corresponding symbols of the μPD27C256A.

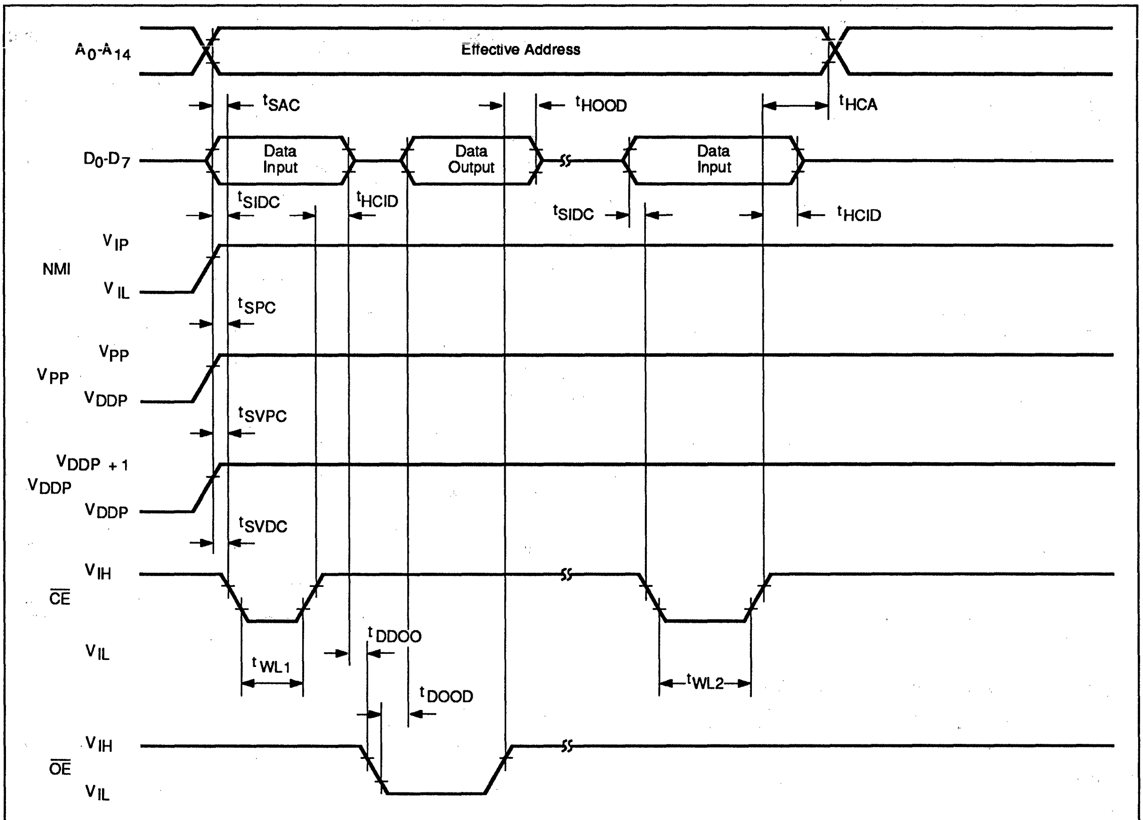
Table 6. AC Programming Characteristics

T_A = 25 ±5°C, V_{IP} = 12.5 ± 0.5 V applied to NMI pin, V_{ss} = 0 V, V_{DD} = 6 ±0.25 V, V_{PP} = 12.5 ±0.3 V.

Parameter	Symbol	Symbol*	Condition	Min	Typ	Max	Unit
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}		2			μs
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}		2			μs
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}		2			μs
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}		2			μs
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}		2			μs
Output data hold time to $\overline{OE}$ ↑	t_{HOOD}	t_{DF}		0		130	ns
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}		1			ms
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}		1			ms
Initial program pulse width	t_{WL1}	t_{PW}		0.95	1.0	1.05	ms
Additional program pulse width	t_{WL2}	t_{OPW}		2.85		78.75	ms
NMI high-voltage input setup time (vs. $\overline{CE}$ ↓)	t_{SPC}			2			μs
Address to data output time	t_{DAOD}	t_{ACC}	$\overline{CE} = \overline{OE} = V_{IL}$			200	ns
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}	$\overline{OE} = V_{IL}$			200	ns
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}	$\overline{CE} = V_{IL}$			75	ns
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	$\overline{CE} = V_{IL}$	0		60	ns
Data hold time from address	t_{HAOD}	t_{OH}	$\overline{CE} = \overline{OE} = V_{IL}$	0			ns

* Corresponding symbols of the μPD27C256A.

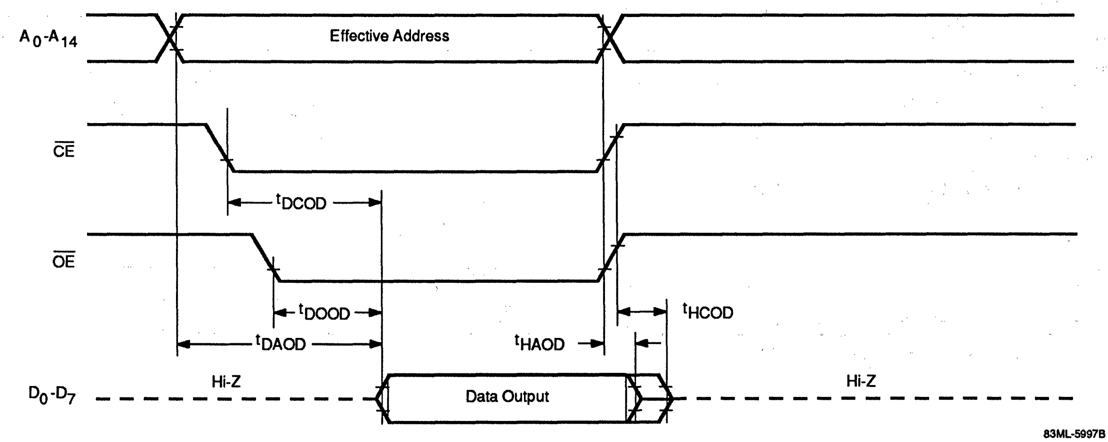
Figure 24. PROM Write Mode Timing



Notes:
(1) V_{DDP} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.
(2) V_{PP} must not exceed +13 V, including overshoot.

83ML-5996B

Figure 25. PROM Read Mode Timing



83ML-5997B

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin to a low level and apply +12.5 V to the NMI pin.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{pp} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This bit is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps 4 to 6.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Fix the $\overline{\text{RESET}}$ pin to a low level and apply +12.5 V to the NMI pin.

(2) Apply +5 V to the V_{DD} and V_{pp} pins.

(3) Input the address of the data to be read to pins A_0-A_{14}.

(4) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(5) Data is output to the D_0 to D_7 pins.

EPROM Erasure

Data in an EPROM is erased by exposing the quartz window in the ceramic package to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 $W \cdot s/cm^2$ (ultraviolet ray intensity × exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12 mW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

INSTRUCTION SET

All microcomputers in the µPD7821x family have a 1-byte instruction lookahead buffer. This allows the first byte of the next opcode in program memory to be fetched while the current opcode is being executed. This pipeline architecture allows instruction fetch and excute cycles to overlap. An instruction can be fetched from program memory while data is being read from or written to RAM or an I/O port.

The advantage of the pipeline is that one instruction can be executed while another is being fetched, virtually halving the time required for these two operations and thereby reducing overall program execution time.

Operands and Operations

Refer to tables 7 and 8 for the meanings of symbols in the operand and operations columns of the Instruction Set table.

Specify operands in accordance with the rules of operand representation; for details, refer to the assembler specifications. If two or more description methods are available, select one. The symbols +, –, #, !, $, /, [ ], and & are keywords and must be used in conjunction with each instruction.

When describing immediate data as a label, use one of the following modifiers: +, –, #, !, $, /, [ ], and &. Symbols r and rp can be described in both the function name and absolute name.

Table 7. Operands

Symbol	Meaning
+	Autoincrement
–	Autodecrement
#	Immediate data
!	Absolute address
$	Relative address
/	Bit inversion
[ ]	Indirect addressing
&	Subbank
r	Register Function name: X, A, C, B, E, D, L, H Absolute name: R0 to R7
r1	Register group 1: C, B
rp	Register pair Function name: AX, BC, DE, HL Absolute name: RP0 to RP3
sfr	Special function register: P0, P2-P7, P0H, P0L, RTPC, CR10, CR11, CR20, CR21, CR22, CR30, PM0, PM3, PM5, PM6, PMC3, PU0, CRC0-CRC2, TOC, TM1-TM3, TMC0, TMC1, PRM0, PRM1, ADM, ADCR, CSIM, SBIC, SIO, ASIM, ASIS, RxB, TxS, BRGC, STBC (dedicated instruction only), MM, PW, RFM, IF0L, IF0H, MK0L, MK0H, PR0L, PR0H, ISM0L, ISM0H, INTM0, INTM1, IST

Table 7. Operands (cont)

Symbol	Meaning
sfrp	Special function register pair: CR00-CR02, TM0, IF0, MK0, PR0, ISM0
mem	Memory address indirectly addressed Register indirect mode: [DE], [HL], [DE+], [HL+], [DE–], [HL–] Base mode: [DE+byte], [HL+byte], [SP+byte] Indexed mode: word[A], word[B], word[DE], word [HL]
mem1	Memory address addressed by means of indirect addressing group 1: [DE], [HL]
saddr	Memory address indirectly addressed: FE20H-FF1FH immediate data or label
saddrp	Memory address addressed by means of direct addressing pair: FE20H-FF1EH immediate data (LSB=0; odd address) or label
addr16	16-bit address: 0000H-FEFFH immediate data or label
addr11	11-bit address: 800H-FFFH immediate data or label
addr5	5-bit address: 40H-7EH immediate data or label
word	16-bit data: 16-bit immediate data or label
byte	8-bit data: 8-bit immediate data or label
bit	3-bit data: 3-bit immediate data or label
n	Number of shift bits: 3-bit immediate data (0-7)
RBn	Register bank: RB0-RB3

Table 8. Registers and Flags

Symbol	Meaning
A	A register; 8-bit accumulator
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R7	Registers 0 to 7 (absolute names)
AX	Register pair (AX); 16-bit accumulator
BC	Register pair (BC)
DE	Register pair (DE)
HL	Register pair (HL)
RP0-RP3	Register pairs 0 to 3 (absolute names)
PC	Program counter
SP	Stack pointer
PSW	Program status word
CY	Carry flag
AC	Auxiliary flag
Z	Zero flag
RBS1-RBS0	Register bank select flags
IE	Interrupt enable flag
STBC	Standby control register

Table 8. Registers and Flags (cont)

Symbol	Meaning
( )	Memory contents indicated by address or register contents in ( )
xxH	Hexadecimal number
x_H, x_L	Higher 8 bits and lower 8 bits of 16-bit register pair

Clocks

The clock field specifies the number of clocks required under the conditions defined by the four column headings as follows:

IROM Program in internal ROM is executed.

IRAM Program in external ROM is executed and internal RAM is accessed.

SFR Program in external ROM is executed and special function register is accessed.

EMEM Program in external ROM is executed and external memory is accessed.

In a shift/rotate instruction, n in the clock field indicates the number of bits by which data is shifted.

The hyphen (-) indicates a range of values; for example 10-13 means 10, 11, 12, or 13.

The virgule symbol (/) means either/or; for example, a/b means either a or b.

The number of clocks when execution is branched by a conditional branch instruction is shown after the symbol (/).

The number of clocks for instruction having the saddr or saddrp operand and when an SFR is accessed with FF00H to FFFFH described as saddr or saddrp is shown after the symbol (/).

Bytes and Clocks

The number of bytes and clocks for instructions with a mem or &mem operand depends on the particular instruction and the memory addressing mode (register indirect, base, or indexed). Table 9 is applicable when the program in internal ROM is executed (ROM clock column of the Instruction Set table). Table 10 is applicable when the program in external ROM is executed (IRAM, SFR, and EMEM clock columns).

Flags

The symbols in the flag field have the following meanings.

Blank No change
0 Cleared to 0
1 Set to 1
x Set or cleared depending on the result
R Value previously saved is restored

Operation Codes

Table 11 defines the symbols used in the operation code field.

Registers and Register Pairs. The r, rl, and rp operands are specified in the opcode by one or more bits as shown in figure 26. For example, 001 as bits $R_2R_1R_0$ (or $R_6R_5R_4$) specifies register A.

In the first and second operands are registers or register pairs, the higher 4 bits of the register specification byte define the first operand and the lower 4 bits define the second operand. For example, in the MOV A,L instruction (transfer L register contents to register A), the second byte of the opcode is obtained from figure 26 as shown below.

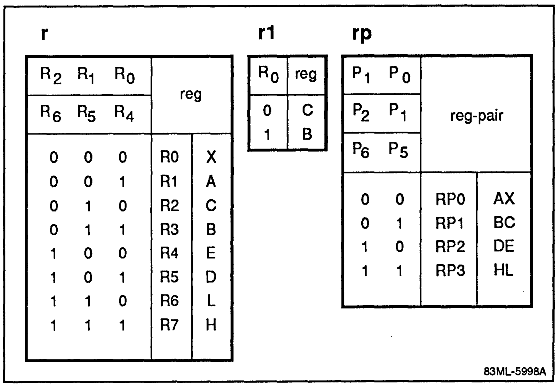
Instruction	Opcode, Bytes 1 and 2	
MOV r,r	0 0 1 0	0 1 0 0
	0 $R_6R_5R_4$	0 $R_2R_1R_0$
MOV A,L	0 0 1 0	0 1 0 0
	0 0 0 1	0 1 1 0

Memory Addressing Modes. The 3-bit mem code and the 5-bit mod code are selected from figure 27 according to the description of mem in the operand field (table 7).

A MOV instruction with register indirect mode specified for mem is a special 1-byte instruction. When base mode or indexed mode is specified for mem, the 8-bit or 16-bit offset data corresponding to byte and word, respectively, is added from the third byte onward.

The opcode for an &mem or &mem1 operand is modified by inserting a 01H code as the first byte preceding the first-byte code listed in the Instruction Set table. Subsequent bytes are as shown in the table.

Figure 26. Opcodes for Registers (r, r1, rp)



r

R_2 R_1 R_0	reg	
R_6 R_5 R_4		
0 0 0	R0	X
0 0 1	R1	A
0 1 0	R2	C
0 1 1	R3	B
1 0 0	R4	E
1 0 1	R5	D
1 1 0	R6	L
1 1 1	R7	H

r1

R_0	reg
0	C
1	B

rp

P_1 P_0	reg-pair	
P_2 P_1		
P_6 P_5		
0 0	RP0	AX
0 1	RP1	BC
1 0	RP2	DE
1 1	RP3	HL

83ML-5998A

6

Figure 27. Opcodes for Memory Addressing Modes (mem, mod)

Mod / Mem	1 0110	0 0110	0 1010
	Register Indirect Mode	Base Mode	Index Mode
0 0 0	[DE+]	[DE+byte]	word [DE]
0 0 1	[HL+]	[SP+byte]	word [A]
0 1 0	[DE-]	[HL+byte]	word [HL]
0 1 1	[HL-]	-	word [B]
1 0 0	[DE]	-	-
1 0 1	[HL]	-	-

83ML-5999A

Table 9. Bytes and Clocks for Instructions With "mem" and "&mem" Operands; Internal ROM (IROM)

			Register Indirect Mode		Base Mode		Indexed Mode
Instruction			[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE+byte] [HL+byte]	[SP+byte]	word[A] word[B] word[DE] word[HL]
Bytes		mem	1/2*	1/2*	3	3	4
		&mem	2/3*	2/3*	4	4	5
Clock Cycles	MOV	A, mem	6/8	6/8	8-11	9-12	8-11
		mem, A					
		A, &mem	8/10	8/10	10-13	11-14	10-13
		&mem, A					
	XCH	A, mem	11-15	9-13	10-15	11-16	10-15
		A, &mem	13-17	11-15	12-17	13-18	12-17
	ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	10/12	8/12	9/12	10-13	9-12
		A, &mem	12/14	10/14	11/14	12-15	11-14

* When internal RAM is accessed with an instruction having a mem operand, the number of bytes is the number before the symbol (/). When the external memory (including the SFR area) is accessed, the number of bytes is the number after the symbol (/).

Table 10. Bytes and Clocks for Instructions With "mem" and "&mem" Operands; External ROM (IRAM, SFR, EMEM)

			Register Indirect Mode		Base Mode		Indexed Mode
Instruction			[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE + byte] [HL + byte]	[SP + byte]	word[A] word[B] word[DE] word[HL]
Bytes		mem	2*	2*	3	3	4
		&mem	3*	3*	4	4	5
Clock Cycles	MOV	A, mem	9/11	6/8	11/13	12/14	14/16
		mem, A					
		A, &mem	12/14	9/11	14/16	15/17	17/19
		&mem, A					
	XCH	A, mem	14/18	12/16	13/17	14/18	16/20
		A, &mem	17/21	15/19	16/20	17/21	19/23
	ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	13/15	11/13	12/14	13/15	15/17
		A, &mem	16/18	14/16	15/17	16/18	18/20

* When [DE], [HL], [DE+], [HL+], [DE–], or [HL–] is specified as the mem operand of a MOV instruction, the instruction is used as a dedicated 1-byte type. When the operand is &mem, the instruction is 2-byte.

Table 11. Opcode Symbols

Symbol	Meaning
Bn	Immediate data corresponding to bit
Nn	Immediate data corresponding to n
Data	8-bit immediate data corresponding to byte
Low/High Byte	16-bit immediate data corresponding to word
Saddr-offset	Lower 8-bit offset data of 16-bit address corresponding to saddr
Sfr-offset	Lower 8-bit offset data of 16-bit address of special function register (sfr)
Low/High Offset	16-bit offset data corresponding to word in indexed addressing
Low/High Addr	16-bit immediate data corresponding to addr16
jdisp	Signed 2's complement data (8 bits) indicating relative address distance between first address of next instruction and branch destination address
fa	Lower 11 bits of immediate data corresponding to addr11
ta	Lower 5 bits of immediate data corresponding to (addr5xdis)

Instruction Set

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z AC CY**	**Bytes B1 thru B5**
8-Bit Data Transfer									
MOV	r,#byte	r ← byte	2	2	6				1 0 1 1 1 R_2 R_1 R_0 Data
	saddr,#byte	(saddr) ← byte	3	3/5	9	9	12		0 0 1 1 1 0 1 0 Saddr-offset Data
	sfr,#byte	sfr ← byte	3	5		9	12		0 0 1 0 1 0 1 1 Sfr-offset Data
	r,r	r ← r	2	2	6				0 0 1 0 0 1 0 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,r	A ← r	1	2	3				1 1 0 1 0 R_2 R_1 R_0
	A,saddr	A ← (saddr)	2	2/4	6	6	9		0 0 1 0 0 0 0 0 Saddr-offset
	saddr,A	(saddr) ← A	2	3/5	6	8			0 0 1 0 0 0 1 0 Saddr-offset
	saddr, saddr	(saddr) ← (saddr)	3	3-7	9				0 0 1 1 1 0 0 0 Saddr-offset Saddr-offset
	A,sfr	A ← sfr	2	4		6			0 0 0 1 0 0 0 0 Sfr-offset
	sfr,A	sfr ← A	2	5		6			0 0 0 1 0 0 1 0 Sfr-offset
	A,mem	A ← (mem)	1-4	6-12	6-14	8-16	8-16		* 0 1 0 1 1 mem 0 0 0 mod 0 mem 0 0 0 0 Low Offset High Offset
	A,&mem	A ← (&mem)	2-5	8-14	9-17	11-19	11-19		* 0 0 0 0 0 0 0 1 0 1 0 1 1 mem 0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 0 0 0 0 Low Offset High Offset

Note:

* If [DE], [HL], [DE+], [DE–], [HL+] or [HL–] is described as mem, these instructions are used as dedicated 1-byte codes. If the register name is described as &mem, the instructions are used as dedicated 2-byte codes.

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Data Transfer (cont)											
MOV	mem,A	(mem) ← A	1-4	6-12	6-14	8-16	8-16				* 0 1 0 1 0 mem
											0 0 0 mod
											1 mem 0 0 0 0
											Low Offset
											High Offset
	&mem,A	(&mem) ← A	2-5	8-14	9-17	11-19	11-19				* 0 0 0 0 0 0 0 1
											0 1 0 1 0 mem
											0 0 0 0 0 0 0 1
											0 0 0 mod
											1 mem 0 0 0 0
											Low Offset
											High Offset
	A,!addr16	A ← (!addr16)	4	6/8	14		16				0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 0
											Low Addr
											High Addr
	A,&!addr16	A ← (&!addr16)	5	8/10			19				0 0 0 0 0 0 0 1
											0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 0
											Low Addr
											High Addr
	!addr16, A	(!addr16) ← A	4	6/8	14		17				0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 1
											Low Addr
											High Addr
	&!addr16,A	(&!addr16) ← A	5	8/10			20				0 0 0 0 0 0 0 1
											0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 1
											Low Addr
											High Addr
	PSW,#byte	PSW ← byte	3	3	9	9	9	x	x	x	0 0 1 0 1 0 1 1
											1 1 1 1 1 1 1 0
											Data
	PSW,A	PSW ← A	2	2	6	6	6	x	x	x	0 0 0 1 0 0 1 0
											1 1 1 1 1 1 1 0
	A,PSW	A ← PSW	2	2	6	6	6				0 0 0 1 0 0 0 0
											1 1 1 1 1 1 1 0

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z AC CY	Bytes B1 thru B5
8-Bit Data Transfer (cont)									
XCH	A,r	A ⟷ r	1	4	4				1 1 0 1 1 R_2 R_1 R_0
	r,r	r ⟷ r	2	3	6				0 0 1 0 0 1 0 1
									0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,mem	A ⟷ (mem)	2-4	9-16	12-16		16-20		0 0 0 mod
									0 mem 0 1 0 0
									Low Offset
									High Offset
	A,&mem	A ⟷ (&mem)	3-5	11-18	15-19		19-23		0 0 0 0 0 0 0 1
									0 0 0 mod
									0 mem 0 1 0 0
									Low Offset
									High Offset
	A,saddr	A ⟷ (saddr)	2	4/8	6				0 0 1 0 0 0 0 1
									Saddr-offset
	A,sfr	A ⟷ sfr	3	6/10		13			0 0 0 0 0 0 0 1
									0 0 1 0 0 0 0 1
									Sfr-offset
	saddr,saddr	(saddr) ⟷ (saddr)	3	6-14		10			0 0 1 1 1 0 0 1
									Saddr-offset
									Saddr-offset
16-Bit Data Transfer									
MOVW	rp,#word	rp ← word	3	3	9				0 1 1 0 0 P_2 P_1 0
									Low Byte
									High Byte
	saddrp,#word	(saddrp) ← word	4	4/8	12	12	18		0 0 0 0 1 1 0 0
									Saddr-offset
									Low Byte
									High Byte
	sfrp,#word	sfrp ← word	4	8		12			0 0 0 0 1 0 1 1
									Saddr-offset
									Low Byte
									High Byte
	rp,rp	rp ← rp	2	4	6				0 0 1 0 0 1 0 0
									0 P_6 P_5 0 1 P_2 P_1 0
	AX,saddrp	AX ← (saddrp)	2	6/10	8	12			0 0 0 1 1 1 0 0
									Saddr-offset
	saddrp,AX	(saddrp) ← AX	2	5/9	8	12			0 0 0 1 1 0 1 0
									Saddr-offset

Instruction Set (cont)

				Clocks				Flags			
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
16-Bit Data Transfer (cont)											
MOVW	AX,sfrp	AX ← sfrp	2	10		12					0 0 0 1 0 0 0 1
											Sfr-offset
	sfrp,AX	sfrp ← AX	2	9		12					0 0 0 1 0 0 1 1
											Sfr-offset
	AX,mem1	AX ← (mem1)	2	9-15	12	16	16				0 0 0 0 0 1 0 1
											1 1 1 0 0 0 1 R_0
	AX,&mem1	AX ← (&mem1)	3	11-17	15	19	19				0 0 0 0 0 0 0 1
											0 0 0 0 0 1 0 1
											1 1 1 0 0 0 1 R_0
	mem1, AX	(mem1) ← AX	2	8-14	11	15	15				0 0 0 0 0 1 0 1
											1 1 1 0 0 1 1 R_0
	&mem1,AX	(&mem1) ← AX	3	10-16	14	18	18				0 0 0 0 0 0 0 1
											0 0 0 0 0 1 0 1
											1 1 1 0 0 1 1 R_0
8-Bit Operation											
ADD	A,#byte	A,CY ← A + byte	2	2	6			x	x	x	1 0 1 0 1 0 0 0
											Data
	saddr,#byte	(saddr),CY ← (saddr) + byte	3	3/7	9	11		x	x	x	0 1 1 0 1 0 0 0
											Saddr-offset
											Data
	sfr,#byte	sfr,CY ← sfr + byte	4	9		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 0 0 0
											Sfr-offset
											Data
	r,r	r,CY ← r + r	2	3	7			x	x	x	1 0 0 0 1 0 0 0
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A + (saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 0 0
											Saddr-offset
	A,sfr	A,CY ← A + sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 0 0 0
											Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) + (saddr)	3	3-9	9	11		x	x	x	0 1 1 1 1 0 0 0
											Saddr-offset
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
ADD	A,mem	A,CY ← A + (mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod
											0 mem 1 0 0 0
											Low Offset
											High Offset
	A,&mem	A,CY ← A + (&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 0 0 0
											Low Offset
											High Offset
ADDC	A,#byte	A,CY ← A + byte + CY	2	2	6			x	x	x	1 0 1 0 1 0 0 1
											Data
	saddr,#byte	(saddr),CY ← (saddr) + byte + CY	3	3/7	9	11		x	x	x	0 1 1 0 1 0 0 1
											Saddr-offset
											Data
	sfr,#byte	sfr,CY ← sfr + byte + CY	4	9		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 0 0 1
											Sfr-offset
											Data
	r,r	r,CY ← r + r + CY	2	3	7			x	x	x	1 0 0 1 1 0 0 1
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A + (saddr) + CY	2	2/5	6	7	8	x	x	x	1 0 0 1 1 0 0 1
											Saddr-offset
	A,sfr	A,CY ← A + sfr + CY	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 0 0 1
											Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) + (saddr) + CY	3	3-9	9	11		x	x	x	0 1 1 1 1 0 0 1
											Saddr-offset
											Saddr-offset
	A,mem	A,CY ← A + (mem) + CY	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod
											0 mem 1 0 0 1
											Low Offset
											High Offset
	A,&mem	A,CY ← A + (&mem) + CY	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 0 0 0
											Low Offset
											High Offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
SUB	A,#byte	A,CY ← A−byte	2	2	6			x	x	x	1 0 1 0 1 0 1 0 Data
	saddr,#byte	(saddr),CY ← (saddr)−(byte)	3	3/7	9	11		x	x	x	0 1 1 0 1 0 1 0 Saddr-offset Data
	sfr,#byte	sfr,CY ← sfr−byte	4	9		14		x	x	x	0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 0 Sfr-offset Data
	r,r	r,CY ← r−r	2	3	7			x	x	x	1 0 0 0 1 0 1 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A−(saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 1 0 Saddr-offset
	A,sfr	A,CY ← A−sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 0 1 0 Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) − (saddr)	3	3-9	9	11		x	x	x	0 1 1 1 1 0 1 0 Saddr-offset Saddr-offset
	A,mem	A,CY ← A−(&mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 1 0 Low Offset High Offset
	A,&mem	A,CY ← A−(&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 1 0 Low Offset High Offset
SUBC	A,#byte	A,CY ← A−byte−CY	2	2	6			x	x	x	1 0 1 0 1 0 1 1 Data
	saddr,#byte	(saddr),CY ← (saddr)−byte−CY	3	3/7	9	11		x	x	x	0 1 1 0 1 0 1 1 Saddr-offset Data
	sfr,#byte	sfr,CY ← sfr−byte−CY	4	9		14		x	x	x	0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 1 Sfr-offset Data

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
SUBC	r,r	r,CY ← r−r−CY	2	3	7			x	x	x	1 0 0 0 1 0 1 1
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A−(saddr)−CY	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 1 1
											Saddr-offset
	A,sfr	A,CY ← A−sfr−CY	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 0 1 1
											Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr)−(saddr) −CY	3	3-9	9	11		x	x	x	0 1 1 1 1 0 1 1
											Saddr-offset
											Saddr-offset
	A,mem	A,CY ← A−(mem)−CY	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod
											0 mem 1 0 1 1
											Low Offset
											High Offset
	A,&mem	A,CY ← A−(&mem)−CY	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 0 1 1
											Low Offset
											High Offset
AND	A,#byte	A ← A∧byte	2	2	6			x			1 0 1 0 1 1 0 0
											Data
	saddr,#byte	(saddr) ← (saddr)∧byte	3	3/7	9	11		x			0 1 1 0 1 1 0 0
											Saddr-offset
											Data
	sfr,#byte	sfr ← sfr∧byte	4	9		14		x			0 0 0 0 0 0 0 1
											0 1 1 0 1 1 0 0
											Sfr-offset
											Data
	r,r	r ← r∧r	2	3	7			x			1 0 0 0 1 1 0 0
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∧(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 0 0
											Saddr-offset
	A,sfr	A ← A∧(sfr)	3	7		10		x			0 0 0 0 0 0 0 1
											1 0 0 1 1 1 0 0
											Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∧(saddr)	3	3-9	9	11		x			0 1 1 1 1 1 0 0
											Saddr-offset
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
AND	A,mem	A ← A∧(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod 0 mem 1 1 0 0 Low Offset High Offset
	A,&mem	A ← A∧(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 0 0 Low Offset High Offset
OR	A,#byte	A ← A∨byte	2	2	6			x			1 0 1 0 1 1 1 0 Data
	saddr,#byte	(saddr) ← (saddr) ∨ byte	3	3/7	9	11		x			0 1 1 0 1 1 1 0 Saddr-offset Data
	sfr,#byte	sfr ← sfr ∨ byte	4	9		14		x			0 0 0 0 0 0 0 1 0 1 1 0 1 1 1 0 Sfr-offset Data
	r,r	r ← r∨r	2	3	7			x			1 0 0 0 1 1 1 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∨(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 1 0 Saddr-offset
	A,sfr	A ← A∨sfr	3	7		10		x			0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 0 Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∨(saddr)	3	3-9	9	11		x			0 1 1 1 1 1 1 0 Saddr-offset Saddr-offset
	A,mem	A ← A∨(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod 0 mem 1 1 1 0 Low Offset High Offset
	A,&mem	A ← A∨(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 1 0 Low Offset High Offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
XOR	A,#byte	A ← A∀byte	2	2	6			x			1 0 1 0 1 1 0 1
											Data
	saddr,#byte	(saddr) ← (saddr)∀byte	3	3/5	9	11		x			0 1 1 0 1 1 0 1
											Saddr-offset
											Data
	sfr,#byte	sfr ← sfr∀byte	4	7		14		x			0 0 0 0 0 0 0 1
											0 1 1 0 1 1 0 1
											Sfr-offset
											Data
	r,r	r ← r∀r	2	3	7			x			1 0 0 0 1 1 0 1
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∀(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 0 1
											Saddr-offset
	A,sfr	A ← A∀(sfr)	3	7		10		x			0 0 0 0 0 0 0 1
											1 0 0 1 1 1 0 1
											Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∀(saddr)	3	3-9	9	11		x			0 1 1 1 1 1 0 1
											Saddr-offset
											Saddr-offset
	A,mem	A ← A∀(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod
											0 mem 1 1 0 1
											Low Offset
											High Offset
	A,&mem	A ← A∀(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 1 0 1
											Low Offset
											High Offset
CMP	A,#byte	A – byte	2	2	6			x	x	x	1 0 1 0 1 1 1 1
											Data
	saddr,#byte	(saddr) – byte	3	3/5	9	11		x	x	x	0 1 1 0 1 1 1 1
											Saddr-offset
											Data
	sfr,#byte	sfr – byte	4	7		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 1 1 1
											Sfr-offset
											Data

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
CMP	r,r	r−r	2	3	7			x	x	x	1 0 0 0 1 1 1 1 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A−(saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 1 1 1 Saddr-offset
	A,sfr	A−sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 1 Sfr-offset
	saddr,saddr	(saddr)−(saddr)	3	3-7	9	11		x	x	x	0 1 1 1 1 1 1 1 Saddr-offset Saddr-offset
	A,mem	A−(mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 1 1 1 Low Offset High Offset
	A,&mem	A−(&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 1 1 Low Offset High Offset
16-Bit Operation											
ADDW	AX,#word	AX,CY ← AX + word	3	4	9			x	x	x	0 0 1 0 1 1 0 1 Low Byte High Byte
	AX,rp	AX,CY ← AX + rp	2	6	8			x	x	x	1 0 0 0 1 0 0 0 0 0 0 0 1 P_2 P_1 0
	AX,saddrp	AX,CY ← AX + (saddrp)	2	7/11	9	13		x	x	x	0 0 0 1 1 1 0 1 Saddr-offset
	AX,sfrp	AX,CY ← AX + sfrp	3	13		16		x	x	x	0 0 0 0 0 0 0 1 0 0 0 1 1 1 0 1 Sfr-offset
SUBW	AX,#word	AX,CY ← AX − word	3	4	9			x	x	x	0 0 1 0 1 1 1 0 Low Byte High Byte
	AX,rp	AX,CY ← AX − rp	2	6	8			x	x	x	1 0 0 0 1 0 1 0 0 0 0 0 1 P_2 P_1 P_0
	AX,saddrp	AX,CY ← AX − (saddrp)	2	7/11	9	13		x	x	x	0 0 0 1 1 1 1 0 Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
16-Bit Operation (cont)											
SUBW	AX,sfrp	AX,CY ← AX−sfrp	3	13		16		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 1 0
											Sfr-offset
CMPW	AX,#word	AX−word	3	3	9			x	x	x	0 0 1 0 1 1 1 1
											Low Byte
											High Byte
	AX,rp	AX−rp	2	5	7			x	x	x	1 0 0 0 1 1 1 1
											0 0 0 0 1 P_2 P_1 0
	AX,saddrp	AX−(saddrp)	2	6/10	8	12		x	x	x	0 0 0 1 1 1 1 1
											Saddr-offset
	AX,sfrp	AX−sfrp	3	12		15		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 1 1
											Sfr-offset
Multiplication/Division											
MULU	r	AX ← Axr	2	22	24						0 0 0 0 0 1 0 1
											0 0 0 0 1 R_2 R_1 R_0
DIVUW	r	AX(quotient), r (remainder) ← AX ÷ r	2	71	76						0 0 0 0 0 1 0 1
											0 0 0 1 1 R_2 R_1 R_0
Increment/Decrement											
INC	r	r ← r + 1	1	2	3			x	x		1 1 0 0 0 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) + 1	2	2/6	6	7		x	x		0 0 1 0 0 1 1 0
											Saddr-offset
DEC	r	r ← r−1	1	2	3			x	x		1 1 0 0 1 R_2 R_1 R_0
	saddr	(saddr) ← (saddr)−1	2	2/6	6	7		x	x		0 0 1 0 0 1 1 1
											Saddr-offset
INCW	rp	rp ← rp + 1	1	3	3						0 1 0 0 0 1 P_1 P_0
DECW	rp	rp ← rp−1	1	3	3						0 1 0 0 1 1 P_1 P_0
Shift/Rotate											
ROR	r,n	(CY,r_7 ← r_0, r_{m-1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 0
											0 1 N_2 N_1 N_0 R_2 R_1 R_0
ROL	r,n	(CY,r_0 ← r_7, r_{m+1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 1
											0 1 N_2 N_1 N_0 R_2 R_1 R_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Clocks				Flags			Operation Code (Bits 7-0)
				IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Shift/Rotate (cont)											
RORC	r,n	$(CY \leftarrow r_0, r_7 \leftarrow CY, r_{m-1} \leftarrow r_m)$ xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 0 0 0 N_2 N_1 N_0 R_2 R_1 R_0
ROLC	r,n	$(CY \leftarrow r_7, r_0 \leftarrow CY, r_{m+1} \leftarrow r_m)$ xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 1 0 0 N_2 N_1 N_0 R_2 R_1 R_0
SHR	r,n	$(CY \leftarrow r_0, r_7 \leftarrow 0, r_{m-1} \leftarrow r_m)$ xn times, n=0-7	2	3+2n	5+2n			x	0	x	0 0 1 1 0 0 0 0 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHL	r,n	$(CY \leftarrow r_7, r_0 \leftarrow 0, r_{m+1} \leftarrow r_m)$ xn times, n=0-7	2	3+2n	5+2n			x	0	x	0 0 1 1 0 0 0 1 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHRW	rp,n	$(CY \leftarrow rp_0, rp_{15} \leftarrow 0, rp_{m-1} \leftarrow rp_m)$ xn times, n=0-7	2	3+3n	5+3n			x	0	x	0 0 1 1 0 0 0 0 1 1 N_2 N_1 N_0 R_2 R_1 R_0
SHLW	rp,n	$(CY \leftarrow rp_{15}, rp_0 \leftarrow 0, rp_{m+1} \leftarrow rp_m)$ xn times, n=0-7	2	3+3n	5+3n			x	0	x	0 0 1 1 0 0 0 1 1 1 N_2 N_1 N_0 R_2 R_1 R_0
ROR4	mem1	$A_{3-0} \leftarrow (mem1)_{3-0}, (mem1)_{7-4} \leftarrow A_{3-0}, (mem1)_{3-0} \leftarrow (mem1)_{7-4}$	2	24	26	34	34				0 0 0 0 0 1 0 1 1 0 0 0 1 1 R_1 0
	&mem1	$A_{3-0} \leftarrow (\&mem1)_{3-0}, (\&mem1)_{7-4} \leftarrow A_{3-0}, (\&mem1)_{3-0} \leftarrow (\&mem1)_{7-4}$	3	26	29	37	37				0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 0 1 1 R_1 0
ROL4	mem1	$A_{3-0} \leftarrow (mem1)_{7-4}, (mem1)_{3-0} \leftarrow A_{3-0}, (mem1)_{7-4} \leftarrow (mem1)_{3-0}$	2	25	27	35	35				0 0 0 0 0 1 0 1 1 0 0 1 1 1 R_1 0
	&mem1	$A_{3-0} \leftarrow (\&mem1)_{7-4}, (\&mem1)_{3-0} \leftarrow A_{3-0}, (\&mem1)_{7-4} \leftarrow (\&mem1)_{3-0}$	3	27	30	38	38				0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 1 1 1 R_1 0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
BCD Adjustment											
ADJBA		Decimal adjust accumulator after addition	1	3		3		x	x	x	0 0 0 0 1 1 1 0
ADJBS		Decimal adjust accumulator after addition	–1	3		3		x	x	x	0 0 0 0 1 1 1 1
Bit Manipulation											
MOV1	CY,saddr.bit	CY ← (saddr bit)	3	5/7	9	9	11			x	0 0 0 0 1 0 0 0
											0 0 0 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,sfr.bit	CY ← sfr.bit	3	7		9				x	0 0 0 0 1 0 0 0
											0 0 0 0 1 B_2 B_1 B_0
											Sfr-offset
	CY,A.bit	CY ← A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 0 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 0 0 0 B_2 B_1 B_0
											Sfr-offset
	CY,PSW.bit	CY ← PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 0 0 0 0 B_2 B_1 B_0
	saddr.bit,CY	(saddr bit) ← CY	3	8/12	12	14	14				0 0 0 0 1 0 0 0
											0 0 0 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit,CY	sfr.bit ← CY	3	12		14					0 0 0 0 1 0 0 0
											0 0 0 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit,CY	A.bit ← CY	2	8	10						0 0 0 0 0 0 1 1
											0 0 0 1 1 B_2 B_1 B_0
	X.bit,CY	X.bit ← CY	2	8	10						0 0 0 0 0 0 1 1
											0 0 0 1 0 B_2 B_1 B_0
											Sfr-offset
	PSW.bit,CY	PSW.bit ← CY	2	7		9		x	x		0 0 0 0 0 0 1 0
											0 0 0 1 0 B_2 B_1 B_0
AND1	CY,saddr.bit	CY ← CY ∧ (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 0 1 0 0 B_2 B_1 B_0
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z AC CY**	**Bytes B1 thru B5**
Bit Manipulation (cont)									
AND1	CY,/saddr.bit	CY ← CY ∧ $\overline{\text{(saddr.bit)}}$	3	5/7	9	11		x	0 0 0 0 1 0 0 0
									0 0 1 1 0 B_2 B_1 B_0
									Sfr-offset
	CY,sfr.bit	CY ← CY ∧ sfr.bit	3	7		11		x	0 0 0 0 1 0 0 0
									0 0 1 0 1 B_2 B_1 B_0
									Sfr-offset
	CY,/sfr.bit	CY ← CY ∧ $\overline{\text{sfr.bit}}$	3	7		11		x	0 0 0 0 1 0 0 0
									0 0 1 1 1 B_2 B_1 B_0
	CY,A.bit	CY ← CY ∧ A.bit	2	5	7			x	0 0 0 0 0 0 1 1
									0 0 1 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY ∧ $\overline{\text{A.bit}}$	2	5	7			x	0 0 0 0 0 0 1 1
									0 0 1 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY ∧ X.bit	2	5	7			x	0 0 0 0 0 0 1 1
									0 0 1 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY ∧ $\overline{\text{X.bit}}$	2	5	7			x	0 0 0 0 0 0 1 1
									0 0 1 1 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY ∧ PSW.bit	2	5		7		x	0 0 0 0 0 0 1 0
									0 0 1 0 0 B_2 B_1 B_0
	CY,/PSW.bit	CY ← CY ∧ $\overline{\text{PSW.bit}}$	2	5		7		x	0 0 0 0 0 0 1 0
									0 0 1 1 0 B_2 B_1 B_0
OR1	CY,saddr.bit	CY ← CY V (saddr.bit)	3	5/7	9	11		x	0 0 0 0 1 0 0 0
									0 1 0 0 0 B_2 B_1 B_0
									Saddr-offset
	CY,/saddr.bit	CY ← CY V $\overline{\text{(saddr.bit)}}$	3	5/7	9	11		x	0 0 0 0 1 0 0 0
									0 1 0 1 0 B_2 B_1 B_0
									Sfr-offset
	CY,sfr.bit	CY ← CY V sfr.bit)	3	7		11		x	0 0 0 0 1 0 0 0
									0 1 0 1 1 B_2 B_1 B_0
									Sfr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Bit Manipulation (cont)											
OR1	CY/sfr.bit	CY ← CY V $\overline{\text{sfr.bit}}$	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 0 1 1 B_2 B_1 B_0
	CY,A.bit	CY ← CY V A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY V $\overline{\text{A.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY V X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY V $\overline{\text{X.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 1 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY V PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 0 0 0 B_2 B_1 B_0
	CY,/PSW.bit	CY ← CY V $\overline{\text{PSW.bit}}$	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 0 1 0 B_2 B_1 B_0
XOR1	CY,saddr.bit	CY ← CY $\veebar$ (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 1 1 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,sfr.bit	CY ← CY $\veebar$ sfr.bit	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 1 0 1 B_2 B_1 B_0
											Sfr-ofset
	CY,A.bit	CY ← CY $\veebar$ A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 1 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY $\veebar$ X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 1 0 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY $\veebar$ PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 1 0 0 B_2 B_1 B_0
SET1	saddr.bit	(saddr.bit) ← 1	2	3/7	6						1 0 1 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← 1	3	10		14					0 0 0 0 1 0 0 0
											1 0 0 0 1 B_2 B_1 B_0
											Sfr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Bit Manipulation (cont)											
SET1	A.bit	A.bit ← 1	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 0 1 B_2 B_1 B_0
	X.bit	X.bit ← 1	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 0 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← 1	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											1 0 0 0 0 B_2 B_1 B_0
CLR1	saddr.bit	(saddr.bit) ← 0	2	6/10	6						1 0 1 0 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← 0	3	10		14					0 0 0 0 1 0 0 0
											1 0 0 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit	A.bit ← 0	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 1 1 B_2 B_1 B_0
	X.bit	X.bit ← 0	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 1 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← 0	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											1 0 0 1 0 B_2 B_1 B_0
NOT1	saddr.bit	(saddr.bit) ← 0 $\overline{\text{(saddr.bit)}}$	3	6/10	10	14					0 0 0 0 1 0 0 0
											0 1 1 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3	10		14					0 0 0 0 1 0 0 0
											0 1 1 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2	6	8						0 0 0 0 0 0 1 1
											0 1 1 1 1 B_2 B_1 B_0
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2	6	8						0 0 0 0 0 0 1 1
											0 1 1 1 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← $\overline{\text{PSW.bit}}$	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											0 1 1 1 0 B_2 B_1 B_0
SET1	CY	CY ← 1	1	2		3				1	0 1 0 0 0 0 0 1
CLR1	CY	CY ← 0	1	2		3				0	0 1 0 0 0 0 0 0
NOT1	CY	CY ← $\overline{\text{CY}}$	1	2		3				x	0 1 0 0 0 0 1 0

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z AC CY**	**Bytes B1 thru B5**
Call/Return									
CALL	!addr16	$(SP-1) \leftarrow (PC+3)_H$, $(SP-2) \leftarrow (PC+3)_L$, $PC \leftarrow$!addr16, $SP \leftarrow SP-2$	3	10-15	17		21		0 0 1 0 1 0 0 0 Low Addr High Addr
	rp	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_H \leftarrow rp_H$, $PC_L \leftarrow rp_L$, $SP \leftarrow SP-2$	2	12-17	15		19		0 0 0 0 0 1 0 1 0 1 0 1 1 P_2 P_1 0
CALLF	!addr11	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_{15+11} \leftarrow 00001$, $PC_{10-0} \leftarrow$!addr11, $SP \leftarrow SP-2$	2	10-15	14		18		1 0 0 1 0 ← fa →
CALLT	[addr5]	$(SP-1) \leftarrow (PC+1)_H$, $(SP-2) \leftarrow (PC+1)_L$, $PC_H \leftarrow$ (00000000, addr5 + 1), $PC_L \leftarrow$ (00000000, addr5), $SP \leftarrow SP-2$	1	14-20	20		24		1 1 1 ← ta →
BRK		$(SP-1) \leftarrow PSW$, $(SP-2) \leftarrow (PC+1)_H$, $(SP-3) \leftarrow (PC+1)_L$, $PC_H \leftarrow$ (003FH), $PC_H \leftarrow$ (003FH), $SP \leftarrow SP-3$, $IE \leftarrow 0$	1	16-26	22		28		0 1 0 1 1 1 1 0
RET		$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$	1	10-15	11		15		0 1 0 1 0 1 1 0
RETI		$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $PSW \leftarrow (SP+2)$, $SP \leftarrow SP+3$, $NMIS \leftarrow 0$	1	12-20	15		21	R R R	0 1 0 1 0 1 1 1
RETB		$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$, $PSW \leftarrow (SP+2)$, $SP \leftarrow SP+3$	1	12-20	13		19	R R R	0 1 0 1 1 1 1 1
Stack Manipulation									
PUSH	PSW	$(SP-1) \leftarrow PSW$, $SP \leftarrow SP-1$	1	4-8	5		7		0 1 0 0 1 0 0 1
	sfr	$(SP-1) \leftarrow sfr$, $SP \leftarrow SP-1$	2	7-9	9		12		0 0 1 0 1 0 0 1 Sfr-offset
	rp	$(SP-1) \leftarrow rp_H$ $(SP-2) \leftarrow rp_L$, $SP \leftarrow SP-2$	1	8-13	8		12		0 0 1 1 1 1 P_1 P_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Clocks				Flags			Operation Code (Bits 7-0)
				IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Stack Manipulation (cont)											
POP	PSW	PSW ← (SP), SP ← SP + 1	1	4-8	6		8	R	R	R	0 1 0 0 1 0 0 0
	sfr	sfr ← (SP), SP ← SP + 1	2	9-11	9		12				0 1 0 0 0 0 1 1
											Sfr-offset
	rp	rp_L ← (SP), rp_H ← (SP + 1), SP ← SP +2	1	10-15	11		15				0 0 1 1 0 1 P_1 P_0
MOVW	SP,#word	SP ← word	4	8		12					0 0 0 0 1 0 1 1
											1 1 1 1 1 1 0 0
											Low Byte
											High Byte
	SP,AX	SP ← AX	2	9		11					0 0 0 1 0 0 1 1
											1 1 1 1 1 1 0 0
	AX,SP	AX ← SP	2	10		12					0 0 0 1 0 0 0 1
											1 1 1 1 1 1 0 0
INCW	SP	SP ← SP + 1	2	5		7					0 0 0 0 0 1 0 1
											1 1 0 0 1 0 0 0
DECW	SP	SP ← SP − 1	2	5		7					0 0 0 0 0 1 0 1
											1 1 0 0 1 0 0 1

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z AC CY	Bytes B1 thru B5
Unconditional Branch								
BR	!addr16	PC ← !addr16	3	5	11			0 0 1 0 1 1 0 0
								Low Addr
	rp	$PC_H \leftarrow rp_H$, $PC_L \leftarrow rp_L$	2	6	10			0 0 0 0 0 1 0 1
								0 1 0 0 1 P_2 P_1 0
	$addr16	PC ← $addr16	2	4	9			0 0 0 1 0 1 0 0
								jdisp
Conditional Branch								
BC	$addr16	PC ← $addr16 if CY = 1	2	2/4	9	6		1 0 0 0 0 0 1 1
BL								jdisp
BNC	$addr16	PC ← $addr16 if CY = 0	2	2/4	9	6		1 0 0 0 0 0 1 0
BNL								jdisp
BZ	$addr16	PC ← $addr16 if Z = 1	2	2/4	9	6		1 0 0 0 0 0 0 1
BE								jdisp
BNZ	$addr16	PC ← $addr16 if Z = 0	2	2/4	9	6		1 0 0 0 0 0 0 0
BNE								jdisp
BT	saddr.bit, $addr16	PC ← $addr16 if (saddr.bit) = 1	3	5-9	12	9		0 1 1 1 0 B_2 B_1 B_0
								Saddr-offset
								jdisp
	sfr.bit, $addr16	PC ← $addr16 if sfr.bit = 1	4	7/9	16	13		0 0 0 0 1 0 0 0
								1 0 1 1 1 B_2 B_1 B_0
								Sfr-offset
								jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 1
								1 0 1 1 1 B_2 B_1 B_0
								jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 1
								1 0 1 1 0 B_2 B_1 B_0
								jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 0
								1 0 1 1 0 B_2 B_1 B_0
								jdisp

Instruction Set (cont)

				Clocks			Flags			
Mnemonic	**Operand**	**Operation**	**Bytes**	**Int ROM**	**Branch**	**No Branch**	**Z**	**AC**	**CY**	**Operation Code (Bits 7-0) Bytes B1 thru B5**
Conditional Branch (cont)										
BF	saddr.bit,$addr16	PC ← $addr16 if (saddr.bit) = 0	4	5-9	15	12				0 0 0 0 1 0 0 0 1 0 1 0 1 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← $addr16 if sfr.bit = 0	4	7/9	16	13				0 0 0 0 1 0 0 0 1 0 1 0 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 0	3	5/7	12	9				0 0 0 0 0 0 1 1 1 0 1 0 1 B_2 B_1 B_0 jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 0	3	5/7	12	9				0 0 0 0 0 0 1 1 1 0 1 0 0 B_2 B_1 B_0 jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 0	3	5/7	12	9				0 0 0 0 0 0 1 0 1 0 1 0 0 B_2 B_1 B_0 jdisp
BTCLR	saddr.bit,$addr16	PC ← $addr16 if (saddr.bit) = 1 then reset (saddr.bit)	4	5-13	15	12				0 0 0 0 1 0 0 0 1 1 0 1 1 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← $addr16 if sfr.bit = 1 then reset sfr.bit	4	7/13	18	13				0 0 0 0 1 0 0 0 1 1 0 1 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 1 then reset A.bit	3	5/9	12	9				0 0 0 0 0 0 1 1 1 1 0 1 1 B_2 B_1 B_0 jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 1 then reset X.bit	3	5/9	12	9				0 0 0 0 0 0 1 1 1 1 0 1 0 B_2 B_1 B_0 jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 1 then reset PSW.bit	3	5/8	12	9	x	x	x	0 0 0 0 0 0 1 0 1 1 0 1 0 B_2 B_1 B_0 jdisp

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z AC CY	Bytes B1 thru B5
Conditional Branch (cont)								
DBNZ	rl,$addr16	rl ← rl − 1, then PC ← $addr16 if rl ≠ 0	2	3/5	9	6		0 0 1 1 0 0 1 R_0
								jdisp
	saddr,$addr16	(saddr) ← (saddr) − 1, then PC ← $addr16 if (saddr) ≠ 0	3	4-10	12	9		0 0 1 1 1 0 1 1
								Saddr-offset
								jdisp
CPU Control								
MOV	STBC,#byte	STBC ← byte	4	10	15			0 0 0 0 1 0 0 1
								1 1 0 0 0 0 0 0
								Data
								Data
SEL	RBn	RBS1–0 ← n, n = 0–3	2	2	6			0 0 0 0 0 1 0 1
								1 0 1 0 1 0 N_1 N_0
NOP		No Operation	1	2	3			0 0 0 0 0 0 0 0
EI		IE ← 1 (Enable Interrupt)	1	2	3			0 1 0 0 1 0 1 1
DI		IE ← 0 (Disable Interrupt)	1	2	3			0 1 0 0 1 0 1 0

NEC
NEC Electronics Inc.

μPD7822x Advanced, 8-Bit Real-Time Control Microcomputers With Analog Comparators

Description

The μPD78220, μPD78224, and μPD78P224 are high-performance, 8-bit, single-chip microcomputers. They contain extended addressing capabilities for up to 1M byte of external memory. The devices also integrate sophisticated analog and digital peripherals as well as two low-power standby modes that make them ideal for low-power/battery backup applications.

The μPD7822x family focuses on embedded control with features such as hardware multiply and divide, two levels of interrupt response, four banks of main registers for multitasking, and macroservice for processor-independent peripheral and memory DMA. Augmenting this high-performance core are advanced components; for example, eight analog voltage comparators, two independent serial interfaces, several counter/timers for PWM outputs, and a real-time output port. On board memory includes 640 bytes of RAM and 16K bytes of mask ROM or OTP ROM.

The macroservice routine allows data to be transferred between any combination of memory and peripherals independent of the current program execution. The four banks of processor registers allow simplified context switching to be performed. Both features combined with powerful on-chip peripherals make this part ideal for a wide variety of embedded control applications.

Features

□ Complete single-chip microcomputer
— 8-bit ALU
— 16K ROM
— 640 bytes RAM
— Both 1-bit and 8-bit logic

□ Instruction prefetch queue

□ Hardware multiply and divide

□ Memory expansion
— 8085 bus-compatible
— 64K program address space
— 1M data address space

□ Large I/O capacity: up to 71 I/O port lines

□ Extensive timer/counter functions
— One 16-bit timer/counter/event counter
— Two 8-bit timer/counter/event counter

□ Four timer-controlled PWM channels

□ Two 4-bit real-time output ports

□ Extensive interrupt handler
— Vectored interrupt handling
— Programmable priority
— Macroservice mode

□ Two independent serial ports

□ Refresh output for pseudostatic RAM

□ On-chip clock generator
— 12-MHz maximum CPU clock frequency
— 0.33-μs instruction cycle

□ CMOS silicon gate technology

□ 5-volt power supply

Ordering Information

Part Number	ROM	Package
μPD78220L μPD78220GJ	ROMless	84-pin PLCC 94-pin plastic QFP
μPD78224L μPD78224GJ	16K Mask ROM	84-pin PLCC 94-plastic QFP
μPD78P224L μPD78P224GJ	16K OTP ROM	84-pin PLCC 94-pin plastic QFP

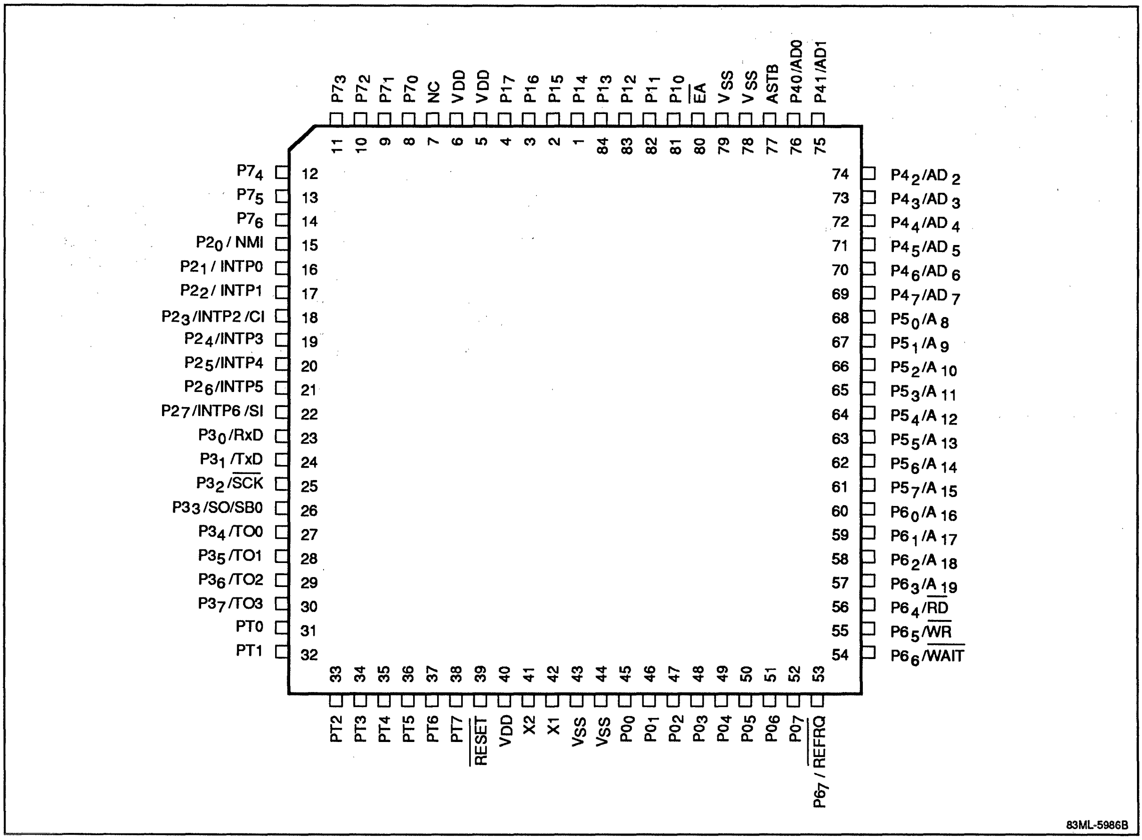
Pin Identification

Symbol	Function
P0$_0$-P0$_7$	Output port 0
P1$_0$-P1$_7$	I/O port 1
P2$_0$/NMI	Input port 2/Non-maskable interrupt input
P2$_1$-P2$_2$/INTP0-INTP1	Input port 2/Ext interrupt input/timer trigger
P2$_3$/INTP2/CI	Input port 2/Ext interrupt input/Clock input
P2$_4$/INTP3	Input port 2/Ext interrupt input/timer trigger
P2$_5$/INTP4	Input port 2/External interrupt input
P2$_6$/INTP5	Input port 2/External interrupt input
P2$_7$/INTP6/SI	Input port 2/Ext interrupt input/Serial input
P3$_0$/RxD	I/O port 3/Serial receive input
P3$_1$/TxD	I/O port 3/Serial transmit output
P3$_2$/$\overline{\text{SCK}}$	I/O port 3/Serial clock input/output
P3$_3$/SO/SB0	I/O port 3/Serial output/Serial bus I/O
P3$_4$-P3$_7$/TO0-TO3	I/O port 3/Timer output
P4$_0$-P4$_7$/AD$_0$-AD$_7$	I/O port 4/Lower address byte/data bus

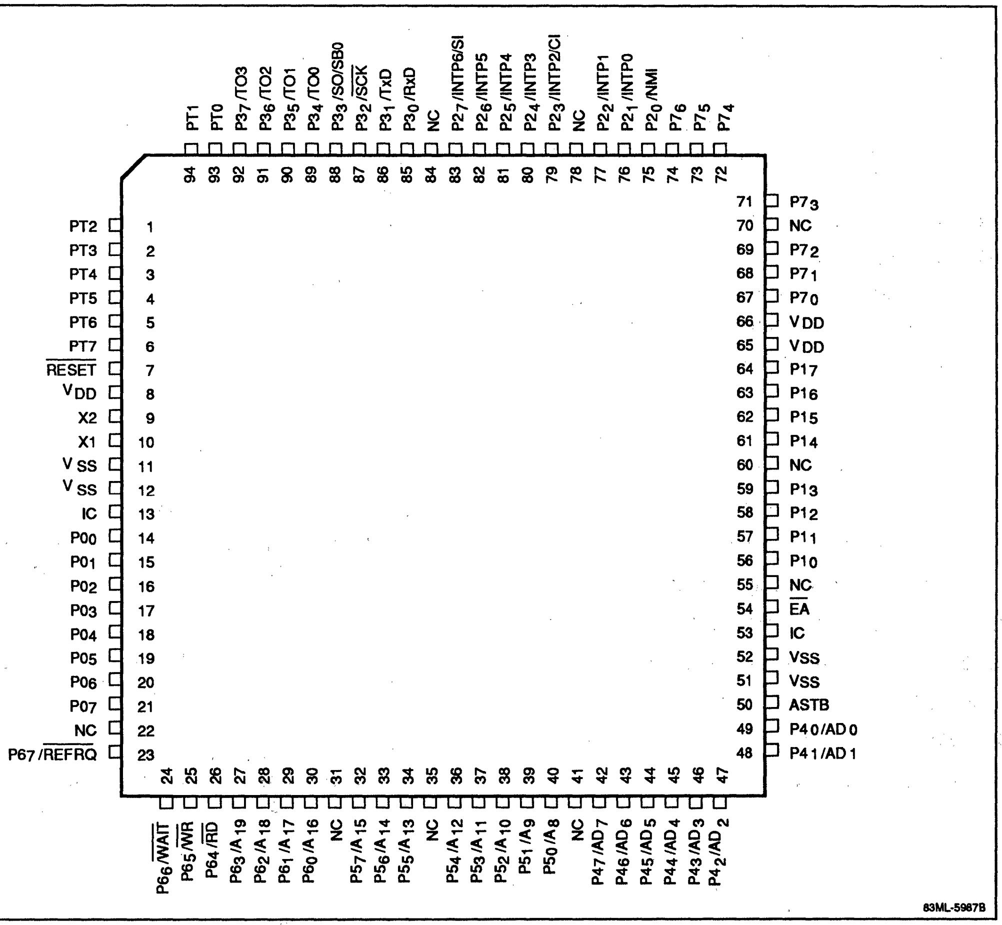
Symbol	Function
P5$_0$-P5$_7$/A$_8$-A$_{15}$	I/O port 5/Upper address byte
P6$_0$-P6$_3$/A$_{16}$-A$_{19}$	Output port 6/Extended address nibble
P6$_4$/$\overline{\text{RD}}$	I/O port 6/Read strobe output
P6$_5$/$\overline{\text{WR}}$	I/O port 6/Write strobe output
P6$_6$/$\overline{\text{WAIT}}$	I/O port 6/Wait input
P6$_7$/$\overline{\text{REFRQ}}$	I/O port 6/Refresh output
P7$_0$-P7$_6$	I/O port 7
PT0-PT7	Port T analog inputs to voltage comparators
ASTB	Address strobe output
$\overline{\text{RESET}}$	External reset input
$\overline{\text{EA}}$	External memory access control input
X1, X2	External crystal or external clock input
V$_{DD}$	Positive power supply input
V$_{SS}$	Power return; normally ground
NC	No connection
IC	Internal connection; connect to V$_{SS}$

Pin Configurations

84-Pin PLCC



94-Pin Plastic QFP



83ML-5987B

Pin Functions

$P0_0$-$P0_7$. Port 0 is an 8-bit, tristate output port. Port 0 can also be configured as two 4-bit, real-time (timer-controlled) output ports.

$P1_0$-$P1_7$. Port 1 is an 8-bit bidirectional tristate port. Bits are individually programmable as input/output. Each pin is capable of driving an LED directly (8 mA).

$P2_0$-$P2_7$. Port 2 is an 8-bit input port.

NMI. Non-maskable interrupt input.

INTP0-INTP6. External interrupt inputs. INTP0, INTP1, and INTP3 are timer capture trigger inputs.

CI. External clock input to the timer.

SI. Serial data input for three-line serial I/O mode.

$P3_0$-$P3_7$. Port 3 is an 8-bit tristate I/O port, each bit programmable as input/output.

RxD. Receive serial data input.

TxD. Transmit serial data output.

$\overline{\text{SCK}}$. Serial shift clock output/input.

SO. Serial data output for three-line serial I/O mode.

SB0. I/O bus for the clocked serial interface.

TO0-TO3. Timer flip-flop outputs.

$P4_0$-$P4_7$. Port 4 is an 8-bit, bidirectional tristate port.

AD_0-AD_7. Multiplexed address/data bus used with external memory or expanded I/O.

$P5_0$-$P5_7$. Port 5 is an 8-bit, tristate output port.

A_8-A_{15}. Upper-order address bus used with external memory or expanded I/O.

$P6_0$-$P6_3$. Pins $P6_0$-$P6_3$ of port 6 are outputs.

A_{16}-A_{19}. Extended-order address bus used with external memory.

$P6_4$-$P6_7$. Pins $P6_4$-$P6_7$ of port 6 are individually programmable tristate input/output pins.

$\overline{\text{RD}}$. Read strobe output used by external memory (or data registers) to place data on the I/O bus during a read operation.

$\overline{\text{WR}}$. Write strobe output used by external memory (or data registers) to latch data from the I/O bus during a write operation.

$\overline{\text{WAIT}}$. Wait signal input.

$\overline{\text{REFRQ}}$. Refresh pulse output used by external pseudo-static memory.

$P7_0$-$P7_6$. Port 7 has seven individually programmable tristate I/O pins.

PT0-PT7. Port T is an eight-line input port. The analog voltage on each line is compared continuously with a programmable threshold voltage.

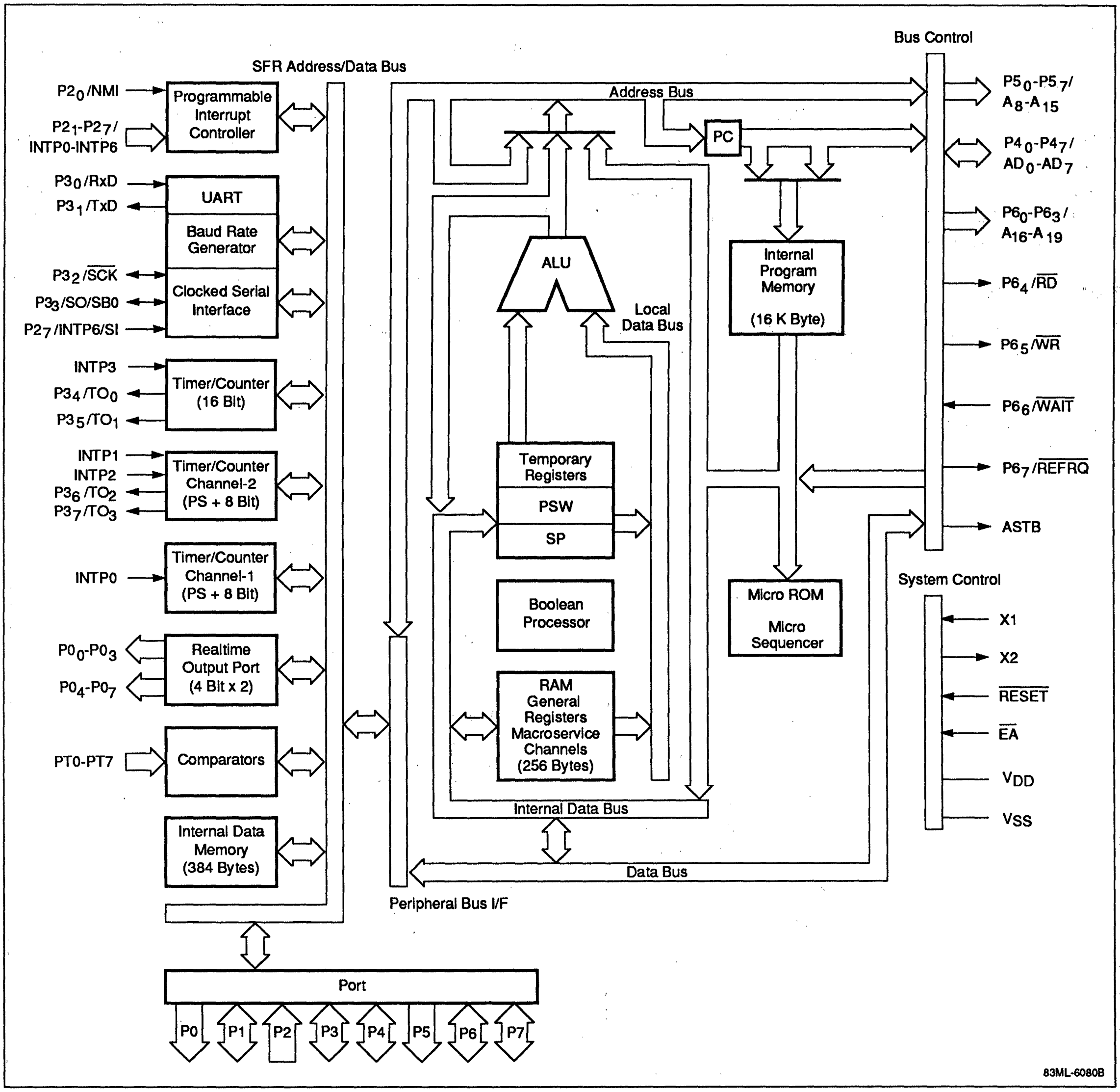
ASTB. Address strobe output used by external circuitry to latch the low-order 8 address bits during the first part of a read or write cycle.

$\overline{\text{RESET}}$. A low level on this external reset input sets all registers to their specified reset values. This pin, together with $P2_0$/NMI, sets the μPD78P224 in the PROM programming mode.

$\overline{\text{EA}}$. Control signal input that selects external memory or internal ROM as the program memory. When $\overline{\text{EA}}$ is low, ROMless mode is initiated and external memory is accessed.

X1, X2. For frequency control of the internal clock oscillator, a crystal is connected to X1 and X2. If the clock is supplied by an external source, the clock signal is connected to X1 and the inverted clock signal is connected to X2.

µPD7822x Block Diagram



FUNCTIONAL DESCRIPTION

Timing

The maximum clock frequency is 12 MHz. The clock is derived from an external crystal or an external oscillator. The internal processor clock is two-phase and the machine states are executed at a rate of 6 MHz. The shortest instructions require two states (333 ns). The CPU contains a one-byte instruction prefetch. This allows a subsequent instruction to be fetched during the execution of an instruction that does not reference memory.

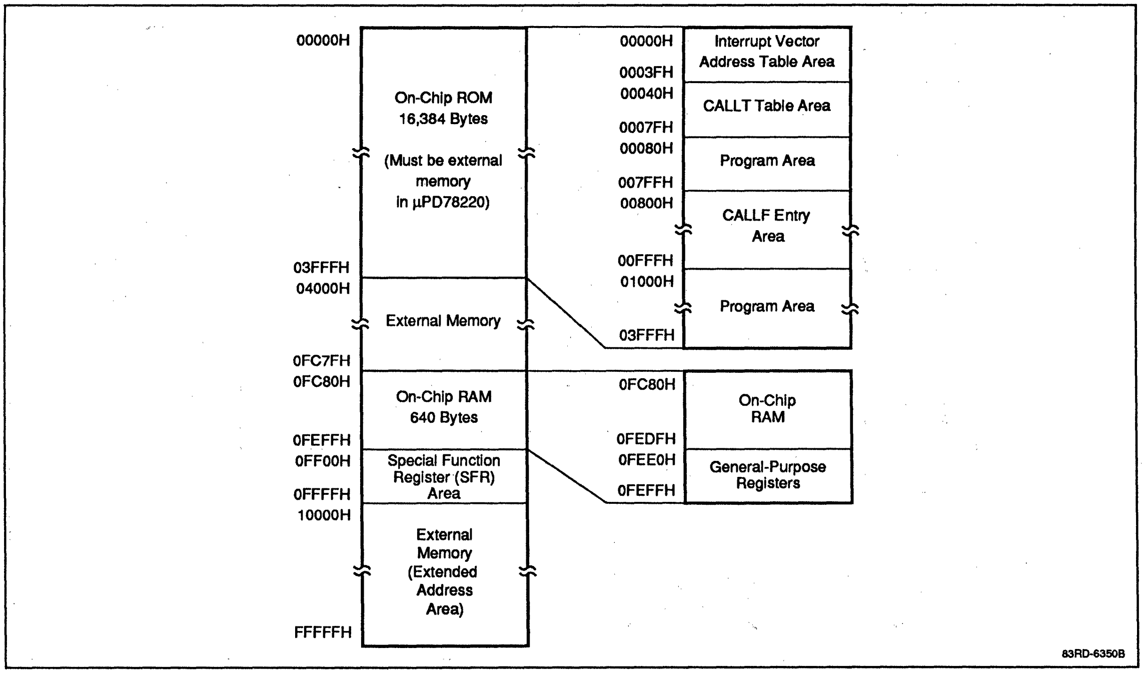
Memory Map

The µPD7822x has 1M bytes of address space. This address space is partitioned into 64K bytes of program memory starting at address 00000H. (See figure 1.) The remainder of the 1M bytes can be accessed as data memory space.

External memory is supported by I/O port 4, an 8-bit multiplexed address/data bus. The memory mapping register controls the size of external memory as well as the number of added wait states. The upper address byte is derived from port 5, and the extended address nibble is derived from port 6.

The µPD78224 has on-chip mask ROM occupying the space from 00000H to 03FFFH. When the ROM is used and no other program or data space is required, ports 4, 5, and 6 are available as additional I/O ports.

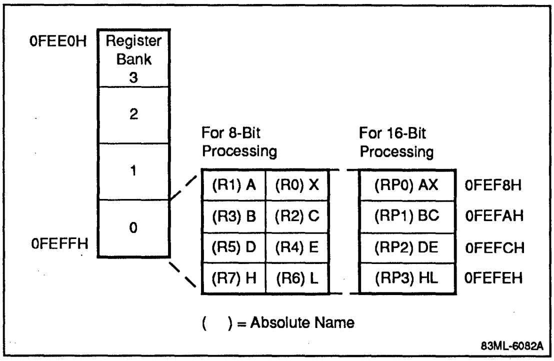
Figure 1. Memory Map



General-Purpose Registers

The general-purpose registers are mapped into specific addresses in data memory. They are made up of four banks, each bank consisting of eight 8-bit or four 16-bit registers. The register bank used is specified by a CPU instruction. This can be checked by reading RBS0 and RBS1 in the program status word (PSW).The general-purpose register configuration is shown in figure 2.

Figure 2. Register Mapping



Special Registers

There are three different special registers. The first is a 16-bit binary counter that holds the next program address to be executed and is named the program counter. The stack pointer is the second special 16-bit register. The stack pointer holds the address of the stack area (a last in, first out system). The third special register is an 8-bit program status word. This register contains various flags that are set or reset depending on the results of instruction execution. The program status word format is as follows:

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

Special Function Registers

These registers are assigned to special functions such as the mode and control registers for on-chip peripheral hardware. They are mapped into the 256-byte memory space from 0FF00H to 0FFFFH. Table 1 is a list of special function registers.

Table 1. Special Function Registers

				Handleable Bit Unit			
Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bit	16 Bit	On Reset
0FF00H	Port 0	P0	R/W	o	o	–	Indeterminate
0FF01H	Port 1	P1	R/W	o	o	–	Indeterminate
0FF02H	Port 2	P2	R	o	o	–	Indeterminate
0FF03H	Port 3	P3	R/W	o	o	–	Indeterminate
0FF04H	Port 4	P4	R/W	o	o	–	Indeterminate
0FF05H	Port 5	P5	R/W	o	o	–	Indeterminate
0FF06H	Port 6	P6	R/W	o	o	–	x0H
0FF07H	Port 7	P7	R/W	o	o	–	Indeterminate
0FF0AH	Port 0 buffer register (low)	P0L	R/W	o	o	–	Indeterminate
0FF0BH	Port 0 buffer register (high)	P0H	R/W	o	o	–	Indeterminate
0FF0CH	Real-time output port control register	RTPC	R/W	o	o	–	00H
0FF10H, 0FF11H	16-bit compare register 0 (16-bit timer/counter)	CR00	R/W	–	–	o	Indeterminate
0FF12H, 0FF13H	16-bit compare register 1 (16-bit timer/counter)	CR01	R/W	–	–	o	Indeterminate
0FF14H	8-bit compare register (8-bit timer/counter 1)	CR10	R/W	–	o	–	Indeterminate
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	–	o	–	Indeterminate
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	–	o	–	Indeterminate
0FF17H	BRG 8-bit compare register	CR30	R/W	–	o	–	Indeterminate
0FF18H, 0FF19H	16-bit capture register (16-bit timer/counter)	CR02	R	–	–	o	Indeterminate
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	–	o	–	Indeterminate
0FF1CH	8-bit capture/compare register (8-bit timer/counter 1)	CR11	R/W	–	o	–	Indeterminate
0FF20H	Port 0 mode register	PM0	W	–	o	–	FFH
0FF21H	Port 1 mode register	PM1	W	–	o	–	FFH
0FF23H	Port 3 mode register	PM3	W	–	o	–	FFH
0FF25H	Port 5 mode register	PM5	W	–	o	–	FFH
0FF26H	Port 6 mode register	PM6	R/W	–	o	–	FFH
0FF27H	Port 7 mode register	PM7	W	–	o	–	7FH
0FF30H	Capture/compare control register 0	CRC0	W	–	o	–	10H
0FF31H	Timer output control register	TOC	W	–	o	–	00H
0FF32H	Capture/compare control register 1	CRC1	W	–	o	–	00H
0FF34H	Capture/compare control register 2	CRC2	W	–	o	–	00H
0FF43H	Port 3 mode control register	PMC3	R/W	o	o	–	00H
0FF50H, 0FF51H	16-bit timer register 0	TM0	R	–	–	o	0000H
0FF52H	8-bit timer register: CH-1	TM1	R	–	o	–	00H

Table 1. Special Function Registers (cont)

					Handleable Bit Unit		
					1	8	16	
Address	**Special Function Register (SFR) Name**	**Symbol**		**R/W**	**Bit**	**Bit**	**Bit**	**On Reset**
0FF54H	8-bit timer register: CH-2	TM2		R	–	o	–	00H
0FF56H	BRG 8-bit timer register	TM3		R	–	o	–	00H
0FF5CH	Prescaler mode register 0	PRM0		W	–	o	–	00H
0FF5DH	Timer control register 0	TMC0		R/W	–	o	–	00H
0FF5EH	Prescaler mode register 1	PRM1		W	–	o	–	00H
0FF5FH	Timer control register 1	TMC1		R/W	–	o	–	00H
0FF6EH	Port T mode register	PMT		R/W	o	o	–	00H
0FF6FH	Port T	PT		R	o	o	–	Indeterminate
0FF80H	Clocked serial interface mode register	CSIM		R/W	o	o	–	00H
0FF82H	Serial bus interface control register	SBIC		R/W	o	o	–	00H
0FF86H	Serial shift register	SIO		R/W	–	o	–	Indeterminate
0FF88H	Asynchronous serial interface mode register	ASIM		R/W	o	o	–	80H
0FF8AH	Asynchronous serial interface status register	ASIS		R	o	o	–	00H
0FF8CH	Serial receive buffer: UART	RxB		R	–	o	–	Indeterminate
0FF8EH	Serial send shift register: UART	TxS		W	–	o	–	Indeterminate
0FFC0H	Standby control register	STBC		R/W	–	o	–	0000 x 000B
0FFC4H	Memory expansion mode register	MM		R/W	o	o	–	20H
0FFC5H	Programmable wait control register	PW		R/W	o	o	–	80H
0FFC6H	Refresh mode register	RFM		R/W	o	o	–	00H
0FFE0H	Interrupt request flag register L	IF0L	IF0	R/W	o	o	o	Indeterminate
0FFE1H	Interrupt request flag register H	IF0H		R/W	o	o		Indeterminate
0FFE4H	Interrupt mask flag register L	MK0L	MK0	R/W	o	o	o	FFFFH
0FFE5H	Interrupt mask flag register H	MK0H		R/W	o	o		FFFFH
0FFE8H	Priority specification flag register L	PR0L	PR0	R/W	o	o	o	FFFFH
0FFE9H	Priority specification flag register H	PR0H		R/W	o	o		FFFFH
0FFECH	Interrupt service mode specification flag register L	ISM0L	ISM0	R/W	o	o	o	0000H
0FFEDH	Interrupt service mode specification flag register H	ISM0H		R/W	o	o		0000H
0FFF4H	External interrupt mode register 0	INTM0		R/W	o	o	–	00H
0FFF5H	External interrupt mode register 1	INTM1		R/W	o	o	–	00H
0FFF8H	Interrupt status register	IST		R/W	o	o	–	00H

Input/Output Ports

Functions of ports P0-P7 and PT are explained below. All ports are 8 bits wide except P7, which is 7 bits wide.

Port	Function
P0	8-bit output port or two 4-bit real time output ports
P1	Bit programmable for input or output; large current capacity
P2	Input
P3	Bit programmable for input or output
P4	Input or output
P5	Output
$P6_0$-$P6_3$	Output
$P6_4$-$P6_7$	Bit programmable for input or output
P7	Bit programmable for input or output
PT	Inputs to eight voltage comparators

Real-Time Output Port

The real-time output port (figure 3) shares pins with port 0. The high and low nibbles may be treated separately or together. In the real-time output function, data stored beforehand in the buffer register is transferred to the output latch simultaneously with the generation of either a timer interrupt or external interrupt. Using the real-time output function in conjunction with the macroservice function enables port 0 to output preprogrammed patterns at preprogrammed variable time intervals.

Port T

As shown in figure 4, the analog input voltage on each line of port T is compared with a programmable threshold voltage. The comparator output is 1 if the input voltage is higher than the threshold or 0 if it is lower.

Four bits from the PTM register are decoded to set the threshold voltage at one of 15 steps: V_{DD} × 1/16 through V_{DD} × 15/16. Each comparator operates continuously as follows.

(1) Threshold voltage is set by writing the PTM register.

(2) As each comparison is completed, the result is latched in port T and the next comparison begins.

(3) Unless the PTM register is rewritten, the threshold voltage is not changed.

Two bits from the PTM register specify the connection of pull-up resistors in 4-bit units. When PTM is set to 00H, the resistor ladder is released and threshold voltage is not supplied to the comparators. This can be done in the standby mode to eliminate unnecessary current drain.

Figure 3. Real-Time Output Port

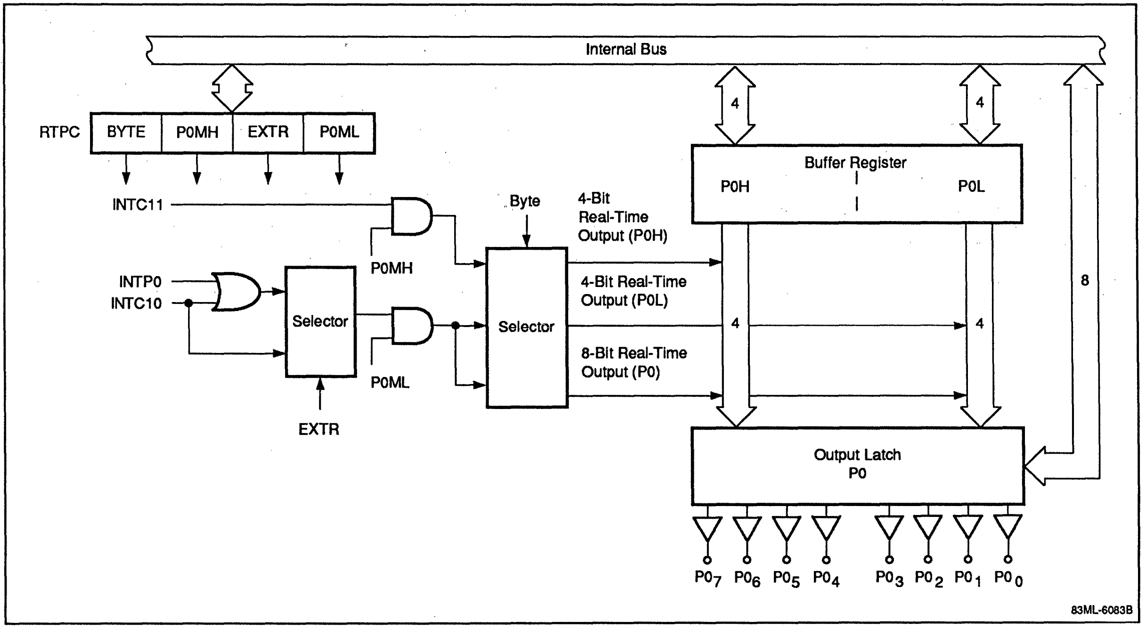
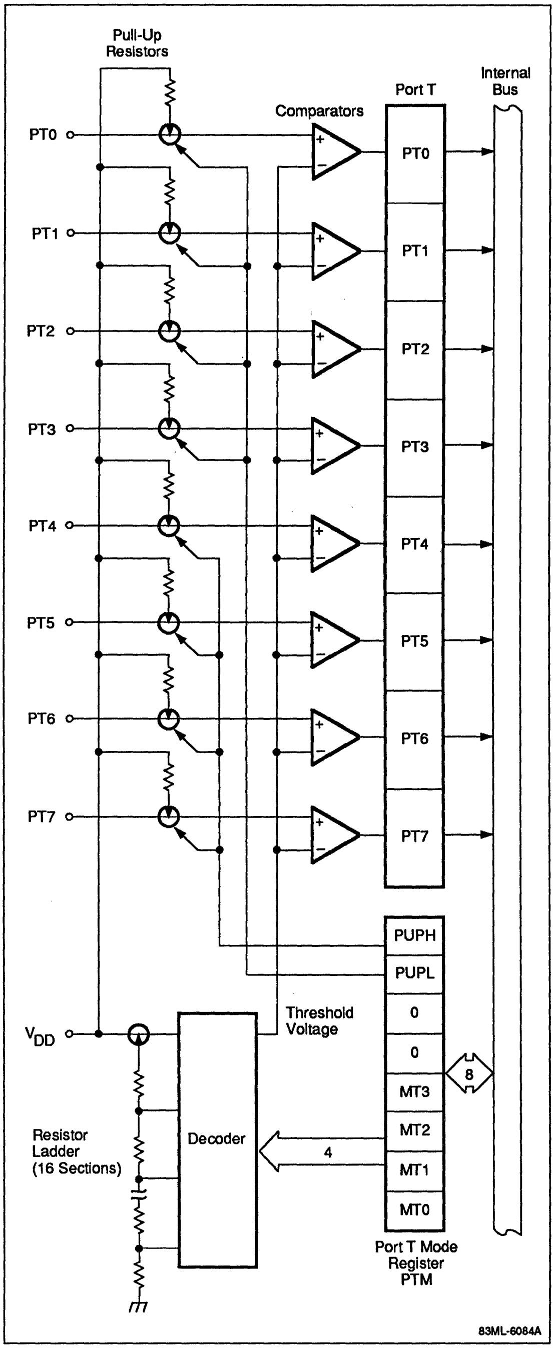

Internal Bus
RTPC
BYTE
P0MH
EXTR
P0ML
INTC11
INTP0
INTC10
Selector
EXTR
P0MH
P0ML
Byte
Selector
4-Bit Real-Time Output (P0H)
4-Bit Real-Time Output (P0L)
8-Bit Real-Time Output (P0)
4
4
4
4
8
Buffer Register
P0H
P0L
Output Latch
P0
$P0_7$ $P0_6$ $P0_5$ $P0_4$ $P0_3$ $P0_2$ $P0_1$ $P0_0$
83ML-6083B

Figure 4. Comparator Port T



Serial Interface

The μPD7822x has two independent serial interfaces.

- Asynchronous serial interface (UART) (figure 5)
- Clock-synchronized serial interface (figure 6)

A universal asynchronous receiver transmitter (UART) is used as an asynchronous serial interface. This interface transfers one byte of data following a start bit. The μPD7822x contains a baud rate generator. This allows data to be transferred over a wide range of transfer rates.

The clock-synchronized serial interface has two different modes of operation:

- Three-line serial I/O mode.
  In this mode, data 8 bits long is transferred along three lines: a serial clock (SCK) line and two serial bus lines (SO and SI). This mode is convenient when the μPD7822x is connected to peripheral I/Os and display controllers that have the conventional clock-synchronized serial interface.
- Serial bus interface mode (SBI).
  In this mode the μPD7822x can communicate data with several devices using the serial clock (SCK) and the serial data bus (SB0) lines. This mode conforms to NEC's serial bus format. In SBI mode, addresses that select a device to communicate with, commands that direct the device, and actual data are output to the serial data bus. A handshake line, which was required for connecting several devices in the conventional clock-synchronized serial interface, is not needed.

Figure 5. Asynchronous Serial Interface

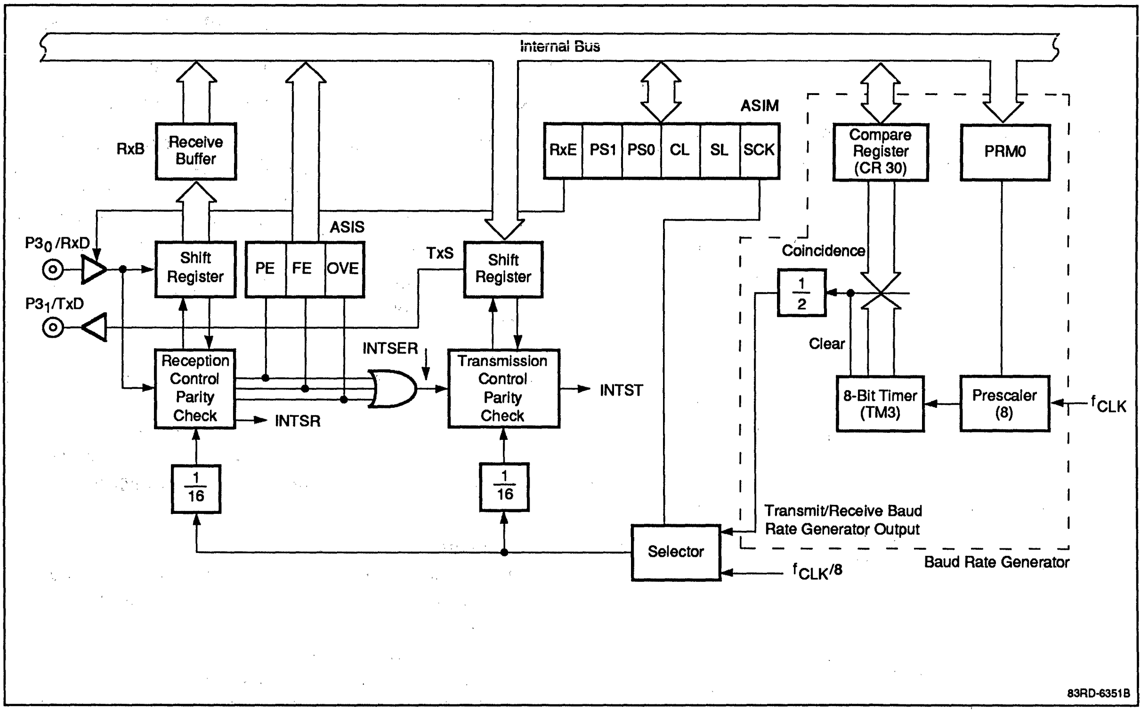
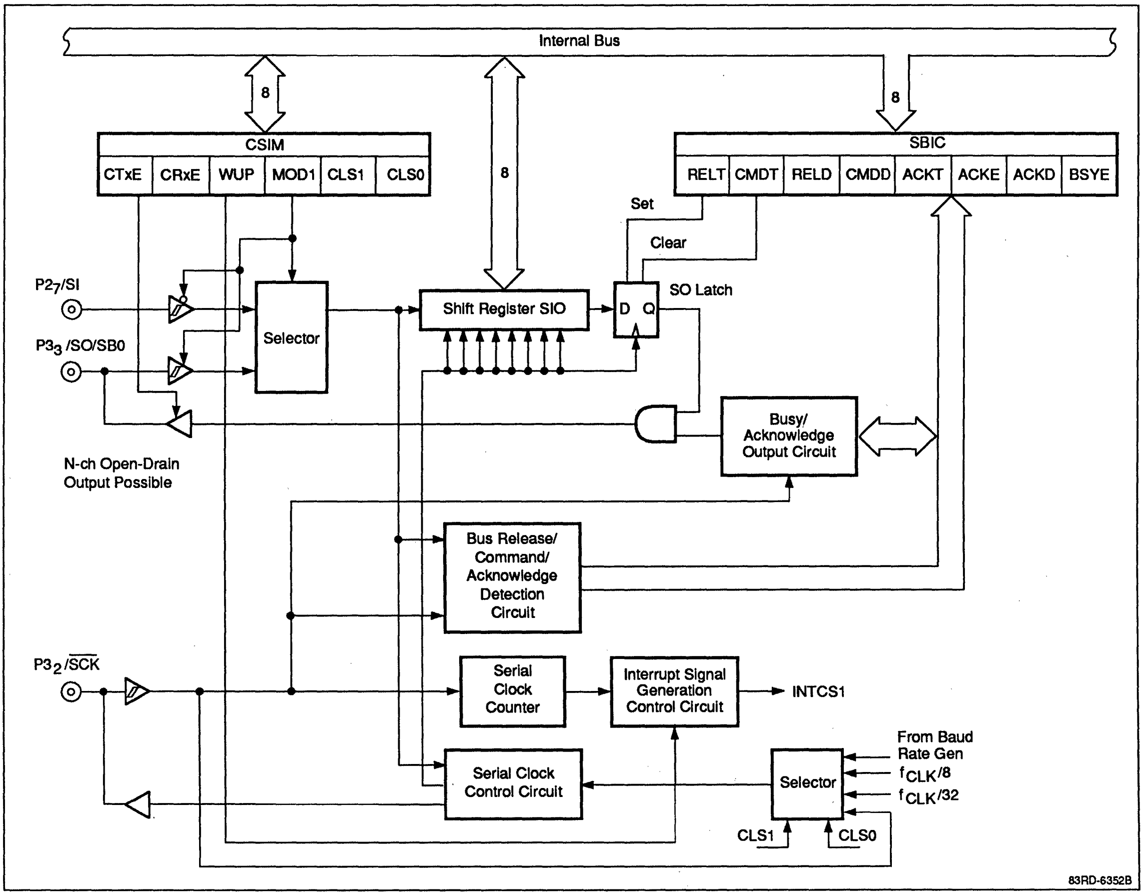


Figure 6. Clock-Synchronized Serial Interface



6

Timer/Counters

The μPD7822x has three timer/counters: one 16-bit and two 8-bit. The 16-bit timer/counter (figure 7) has the basic functionality of an interval timer, a programmable square-wave output, and a pulse width measurer. These functions can provide a digital delayed one-shot output, a pulse width modulated output, and a cycle measurer.

The two 8-bit timer/counters can provide the basic functions of an interval timer and a pulse width measurer. Timer/counter 1 can also be used as a timer for output trigger generation for the real-time output port. Timer/counter 2 can also provide an external event counter, a one-shot timer, a programmable square-wave output, a pulse-width modulated output, and a cycle measurer. See figures 8 and 9.

Figure 7. 16-Bit Timer/Counter

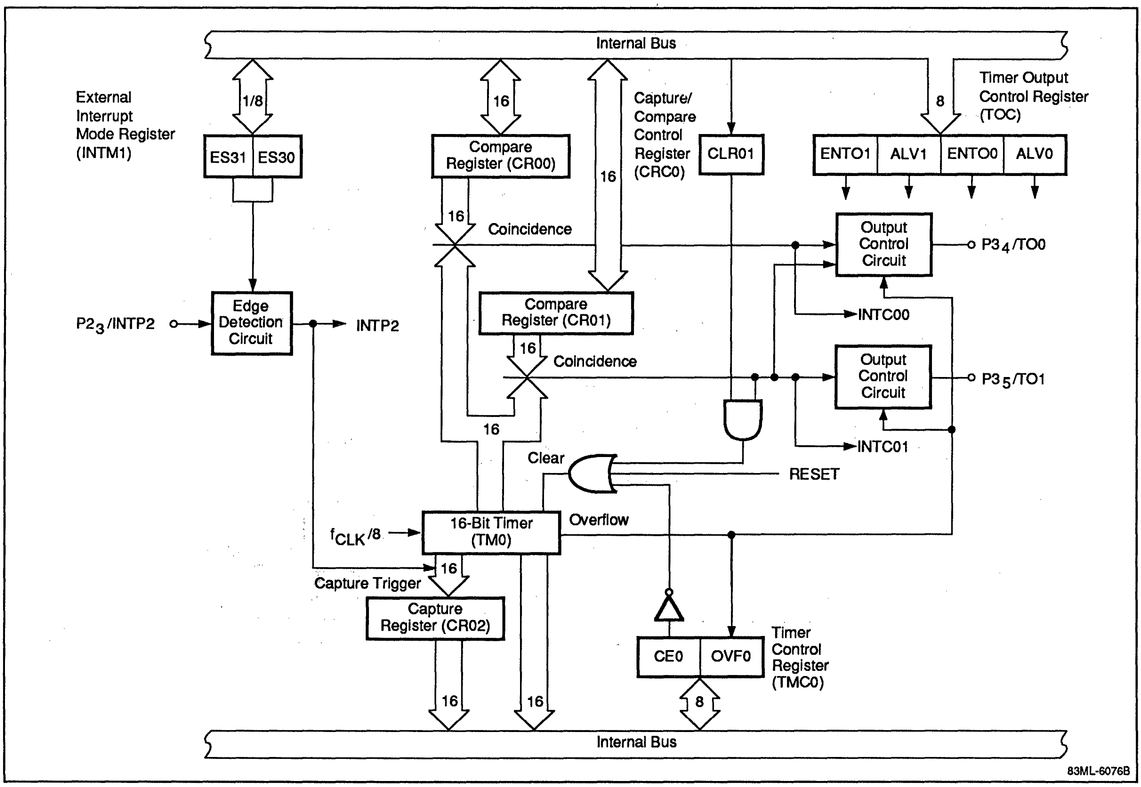


Figure 8. 8-Bit Timer/Counter 1

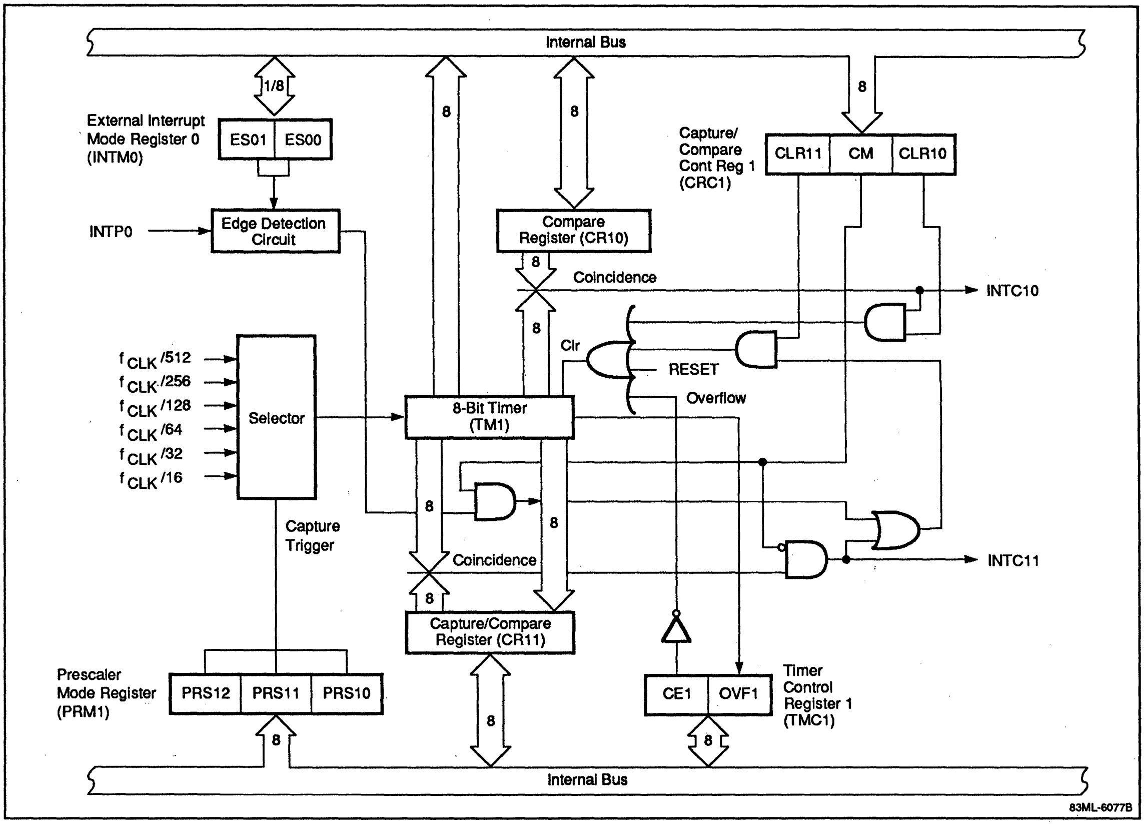
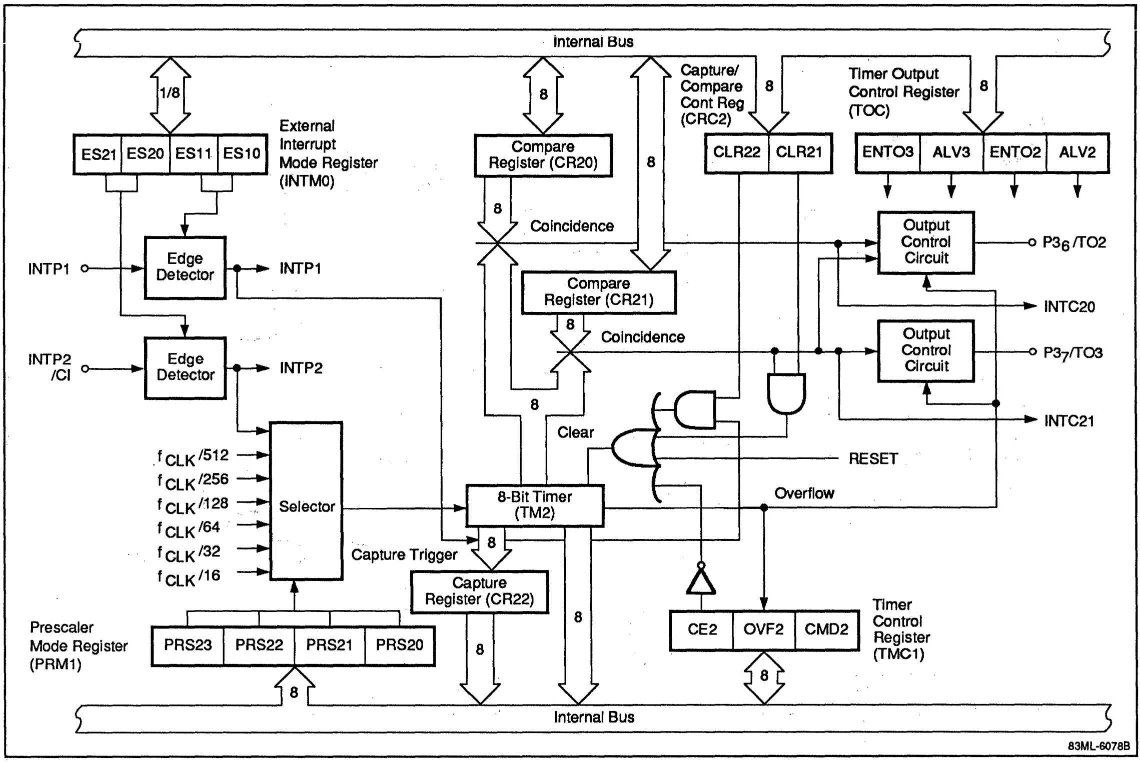


Figure 9. 8-Bit Timer/Counter 2



Interrupts

There are 18 interrupt request sources; each source is allocated a location in the vector table. (See table 2.) There is one software interrupt request and one of the remaining 17 interrupts is non-maskable. The software interrupt and the non-maskable interrupt are unconditionally received even in the DI state. These two interrupts possess the maximum priority. The maskable interrupt requests are subject to mask control by the setting of the interrupt mask flag.

There are default priorities associated with each maskable interrupt and these can be assigned to either of two programmable priority levels. Interrupts may be serviced by the vectored interrupt method where a branch to a desired service program is executed. Interrupts may also be handled by the macroservice function where a preassigned process is performed without program intervention.

Table 2. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Source	Macroservice Handling	Vector Table Address
Software	None	BRK instruction execution	–	003EH
Non-maskable	None	NMI (pin input edge detection)	–	0002H
Maskable	0	INTP0 (pin input edge detection)	–	0006H
	1	INTP1 (pin input edge detection)	–	0008H
	2	INTP2 (pin input edge detection)	–	000AH
	3	INTP3 (pin input edge detection)	–	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	–	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	–	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	Yes	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	Yes	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	–	001CH
	9	INTP4 (pin input edge detection)	Yes	000EH
	10	INTP5 (pin input edge detection)	–	0010H
	11	INTP6 (pin input edge detection)	–	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	–	0020H
	13	INTSR (end of asynchronous serial interface reception)	Yes	0022H
	14	INTST (end of asynchronous serial interface transmission)	Yes	0024H
	15	INTCSI (end of clocked serial interface transfer)	Yes	0026H

Macroservice

The macroservice function can be programmed to transfer data from a special function register to memory or from memory to a special function register. Transfer events are triggered by interrupt requests and take place without software intervention. There are six interrupt requests where macroservicing can be executed. The macroservice function is controlled by the macroservice mode register and the macroservice channel pointer. The macroservice mode register assigns the macroservicing mode and the macroservice channel pointer indicates the address of the memory location pointers. The location of each register and its corresponding interrupt is shown in figure 10.

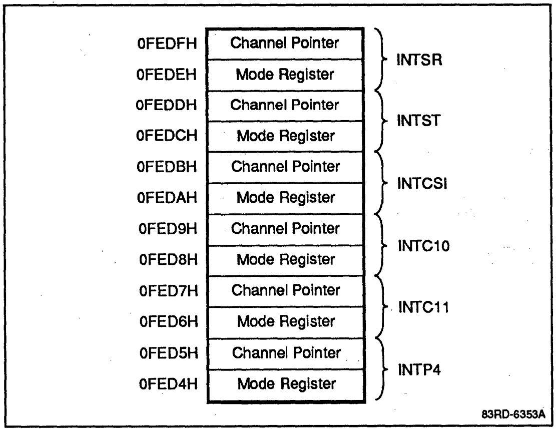
Refresh

The refresh signal is used with a pseudostatic RAM. The refresh cycle can be set to one of four intervals ranging from 2.6 to 21.3 µs. The refresh is timed to follow a read or write operation so there is no interference.

Standby Modes

Halt and stop functions reduce system power consumption. In the halt mode, the CPU stops and the system clock continues to run. A release of the halt mode is initiated by an unmasked interrupt request, an NMI, or a RESET input. In the stop mode, the CPU and system clock are both stopped, reducing the power consumption even further. The stop mode is released by an NMI input or a RESET input.

Figure 10. Macroservice Control Word Map



0FEDFH	Channel Pointer	INTSR
0FEDEH	Mode Register	
0FEDDH	Channel Pointer	INTST
0FEDCH	Mode Register	
0FEDBH	Channel Pointer	INTCSI
0FEDAH	Mode Register	
0FED9H	Channel Pointer	INTC10
0FED8H	Mode Register	
0FED7H	Channel Pointer	INTC11
0FED6H	Mode Register	
0FED5H	Channel Pointer	INTP4
0FED4H	Mode Register	

83RD-6353A

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

$T_A = +25°C$.

Item	Symbol	Conditions	Rating	Unit
Operating voltage	V_{DD}		−0.5 to +7.0	V
Input voltage	V_I		−0.5 to V_{DD} + 0.5	V
Output voltage	V_O		−0.5 to V_{DD} + 0.5	V
Low-level output current	I_{OL}	One output pin	30 (peak)	mA
			15 (mean value)	mA
		All output pins total	150 (peak)	mA
			100 (mean value)	mA
High-level output current	I_{OH}	One output pin	−2	mA
		All output pins total	−50	mA
Operating temperature	T_{OPT}		−40 to +85	°C
Storage temperature	T_{STG}		−65 to +150	°C

Operating Frequency

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	−40 to +85°C	+5 V ±5%
	−10 to +70°C	+5 V ±10%

Capacitance

$T_A = +25°C$; $V_{DD} = V_{SS} = 0$ V.

Item	Symbol	Typ	Max	Unit	Conditions
Input capacitance	C_I		20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O		20	pF	
Input/output capacitance	C_{IO}		20	pF	

DC Characteristics

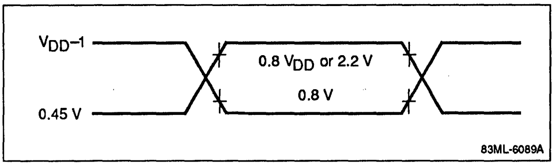
$T_A = -40$ to $+85°C$; $V_{DD} = +5$ V ±10%; $V_{SS} = 0$ V.

Item	Symbol	Conditions		Min	Typ	Max	Unit
Low-level input voltage	V_{IL}	Except PT pins		0		0.8	V
High-level input voltage	V_{IH1}	Except PT pins and pins in Note 1		2.2		V_{DD}	V
	V_{IH2}	Pins in Note 1		0.8 V_{DD}		V_{DD}	V
Low-level output voltage	V_{OL1}	I_{OL} = 2.0 mA				0.45	V
	V_{OL2}	I_{OL} = 8.0 mA (Port PI pins)				1.0	V
High-level output voltage	V_{OH1}	I_{OH} = −1.0 mA		V_{DD} − 1.0			V
	V_{OH2}	I_{OH} = −100 μA		V_{DD} − 0.5			V
Input leakage current	I_{LI}	V_I = 0 to V_{DD}				±10	μA
Output leakage current	I_{LO}	V_O = 0 to V_{DD}				±10	μA
Pull-up current	I_{IPT}	V_I = 0 V; PT pins			−150	−400	μA
V_{DD} power supply current	I_{DD1}	Operating mode, f_{XX} = 12 MHz			16	40	mA
	I_{DD2}	HALT mode, f_{XX} = 12 MHz			7	20	mA
Data retention voltage	V_{DDDR}	STOP mode		2.5		5.5	V
Data retention current	I_{DDDR}	STOP mode	V_{DDDR} = 2.5 V		2	20	μA
			V_{DDDR} = 5 V ±10%		5	50	μA

Notes:

(1) X1, X2, $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4, $P2_6$/INTP5, $P2_7$/INTP6/SI, $P3_2$/$\overline{\text{SCK}}$, $P3_3$/SO/SB0, and $\overline{\text{EA}}$ pins.

Figure 11. Voltage Thresholds for Timing Measurements



Read/Write Operation

T_A = –40 to +85°C; V_{DD} = +5 V ± 10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF.

Item	Symbol	Conditions	Min	Max	Unit
X1 input clock cycle time	t_{CYX}		82	250	ns
Address setup time to ASTB ↓	t_{SAST}		52		ns
Address hold time from ASTB ↓ (Note 2)	t_{HSTA}	$R_L = 5\,k\Omega, C_L = 50\,pF$	25		ns
Address to $\overline{RD}$ ↓ delay time	t_{DAR}		129		ns
Address float time from $\overline{RD}$ ↓	t_{FAR}		11		ns
Address to data input time	t_{DAID}			228	ns
ASTB ↓ to data input time	t_{DSTID}			181	ns
$\overline{RD}$ ↓ to data input time	t_{DRID}			99	ns
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}		52		ns
Data hold time from $\overline{RD}$ ↑	t_{HRID}		0		ns
$\overline{RD}$ ↑ to address active time	t_{DRA}		124		ns
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}		124		ns
$\overline{RD}$ low-level width	t_{WRL}		124		ns
ASTB high-level width	t_{WSTH}		52		ns
Address to $\overline{WR}$ ↓ delay time	t_{DAW}		129		ns
$\overline{ASTB}$ ↓ to data output time	t_{DSTOD}			142	ns
$\overline{WR}$ ↓ to data output time	t_{DWOD}			60	ns
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}		52		ns
	t_{DSTW2}	Refresh mode	129		ns
Data setup time to $\overline{WR}$ ↑	t_{SODWR}		146		ns
Data setup time to $\overline{WR}$ ↓ (Note 1)	t_{SODWF}	Refresh mode	22		ns
Data hold time from $\overline{WR}$ ↑ (Note 2)	t_{HWOD}		20		ns
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}		42		ns
$\overline{WR}$ low-level width	t_{WWL1}		196		ns
	t_{WWL2}	Refresh mode	114		ns
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}			146	ns
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}			84	ns
$\overline{WAIT}$ hold time from X1 ↓	t_{HWTX}		0		ns
$\overline{WAIT}$ setup time to X1 ↑	t_{SWTX}		0		ns

Notes:

(1) When accessing a pseudostatic RAM (μPD4168, etc.) that clocks in data at the falling edge of $\overline{WR}$, use t_{SODWF} instead of t_{SODWR} as the data setup time.

(2) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: C_L = 100 pF and R_L = 2 kΩ.

Figure 12. Read Operation Timing

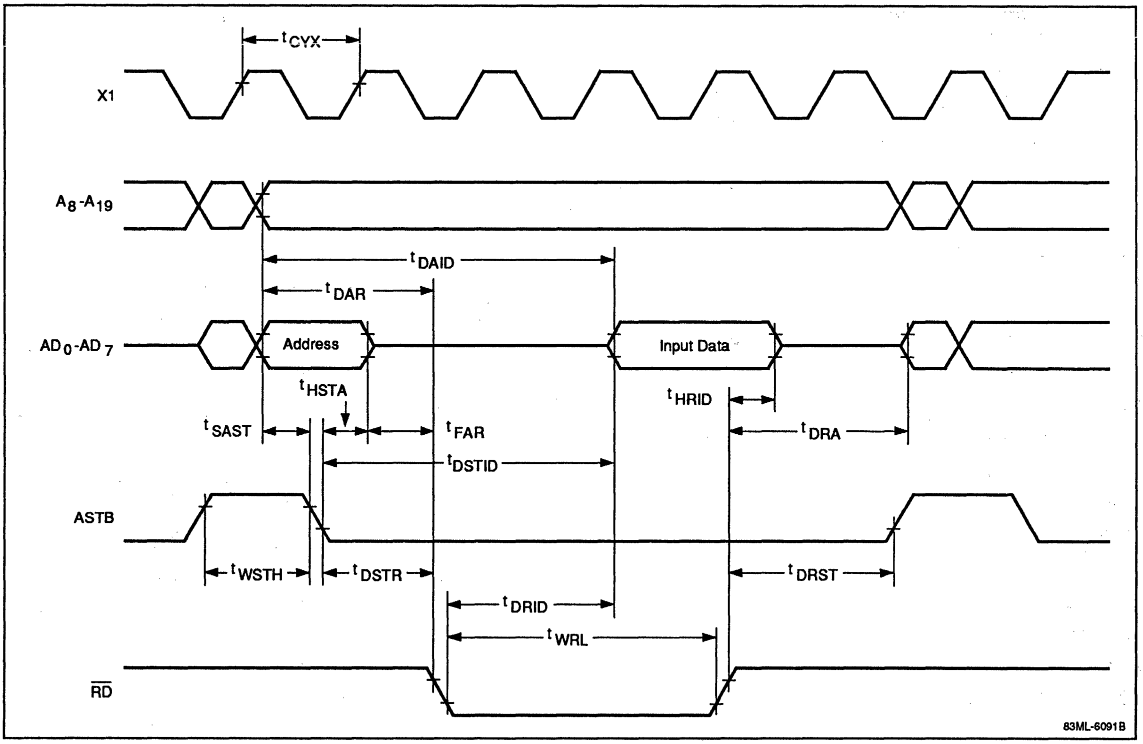


Figure 13. Write Operation Timing

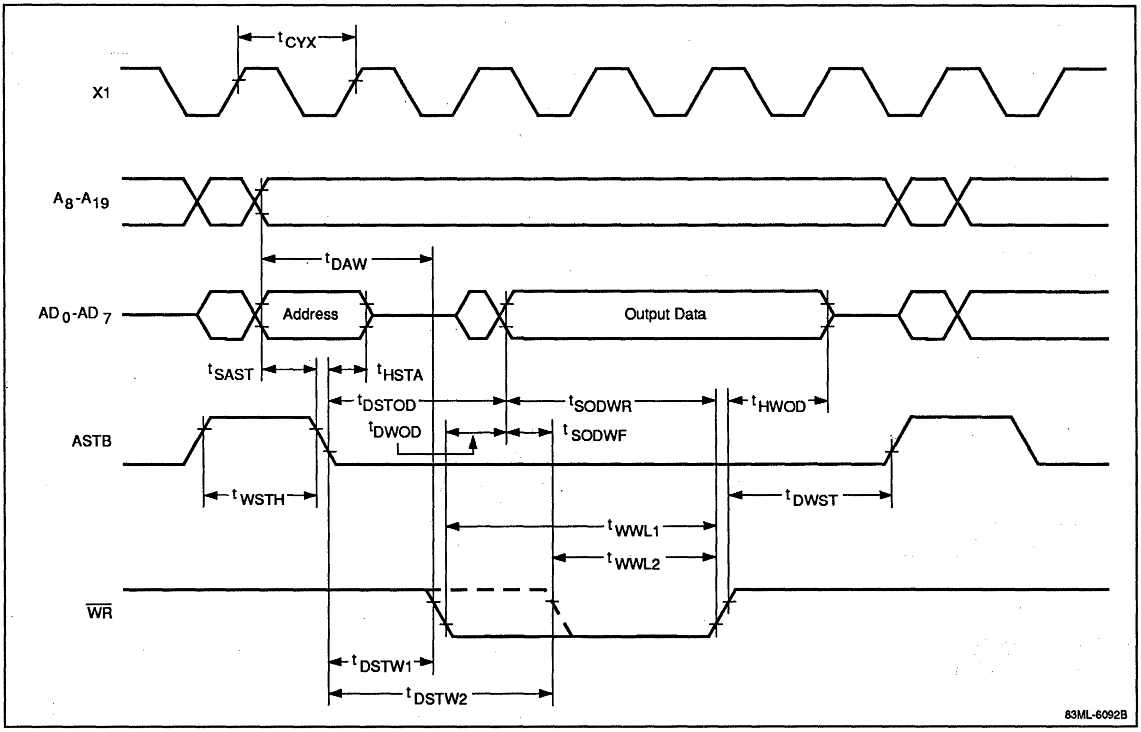
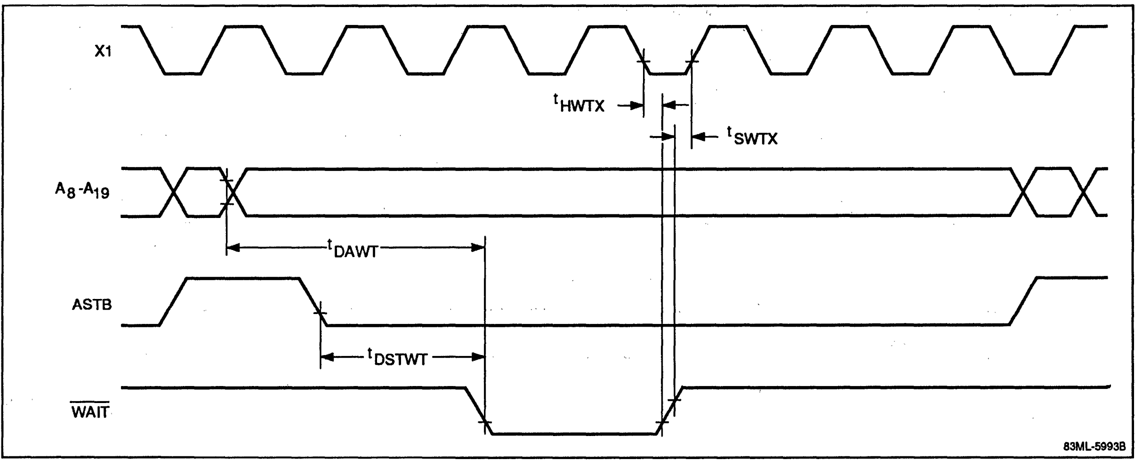


Figure 14. External $\overline{WAIT}$ Input Timing



Serial Port Operation

$T_A = -40$ to $+85°C$; $V_{DD} = +5\ V \pm 10\%$; $V_{SS} = 0\ V$; $f_{XX} = 12\ MHz$; $C_L = 100\ pF$.

Item	Symbol	Conditions		Min	Max	Unit
Serial clock cycle time	t_{CYSK}	Input	External clock	1.0		μs
		Output	Internal clock/16	1.3		μs
			Internal clock/64	5.3		μs
Serial clock low-level width	t_{WSKL}	Input	External clock	420		ns
		Output	Internal clock/16	556		ns
			Internal clock/64	2.5		μs
Serial clock high-level width	t_{WSKH}	Input	External clock	420		ns
		Output	Internal clock/16	556		ns
			Internal clock/64	2.5		μs
SI, SB0 setup time to $\overline{SCK}$ ↑	t_{SSSK}			150		ns
SI, SB0 hold time from $\overline{SCK}$ ↓	t_{HSSK}			400		ns
SO/SB0 output delay time from $\overline{SCK}$ ↓	t_{DSBSK1}	CMOS push-pull output (3-line serial I/O mode)		0	300	ns
	t_{DSBSK2}	Open-drain output (SBI mode), $R_L = 1\ k\Omega$		0	800	ns
SB0 high, hold time from $\overline{SCK}$ ↑	t_{HSBSK}	SBI mode		4		t_{CYX}
SB0 low, setup time to $\overline{SCK}$ ↓	t_{SSBSK}	SBI mode		4		t_{CYX}
SB0 low-level width	t_{WSBL}			4		t_{CYX}
SB0 high-level width	t_{WSBH}			4		t_{CYX}
RxD setup time to $\overline{SCK}$ ↑	t_{SRXSK}			80		ns
RxD hold time after $\overline{SCK}$ ↑	t_{HSKRX}			80		ns
$\overline{SCK}$ ↓ to TxD delay time	t_{DSKTX}				210	ns

Figure 15. Clock-Synchronized Serial Interface Timing; Three-Line I/O Mode

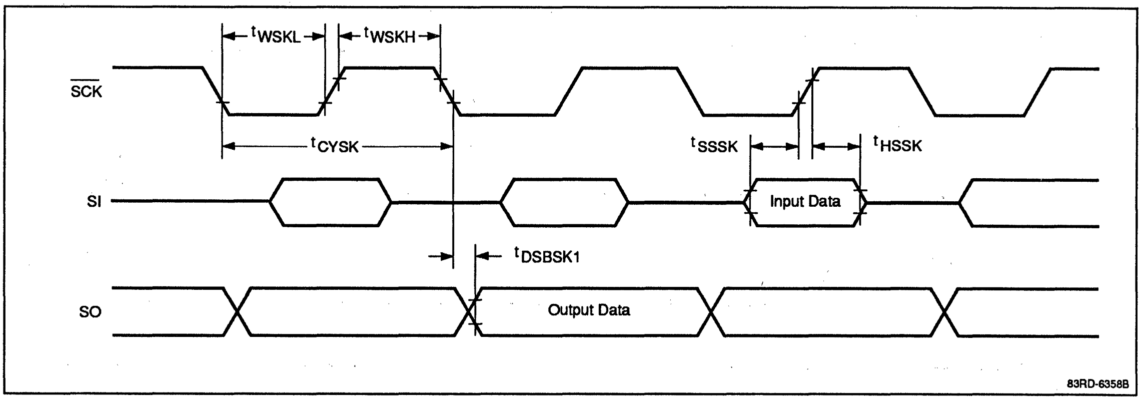


Figure 16. Clock-Synchronized Serial Interface Timing; SBI Mode

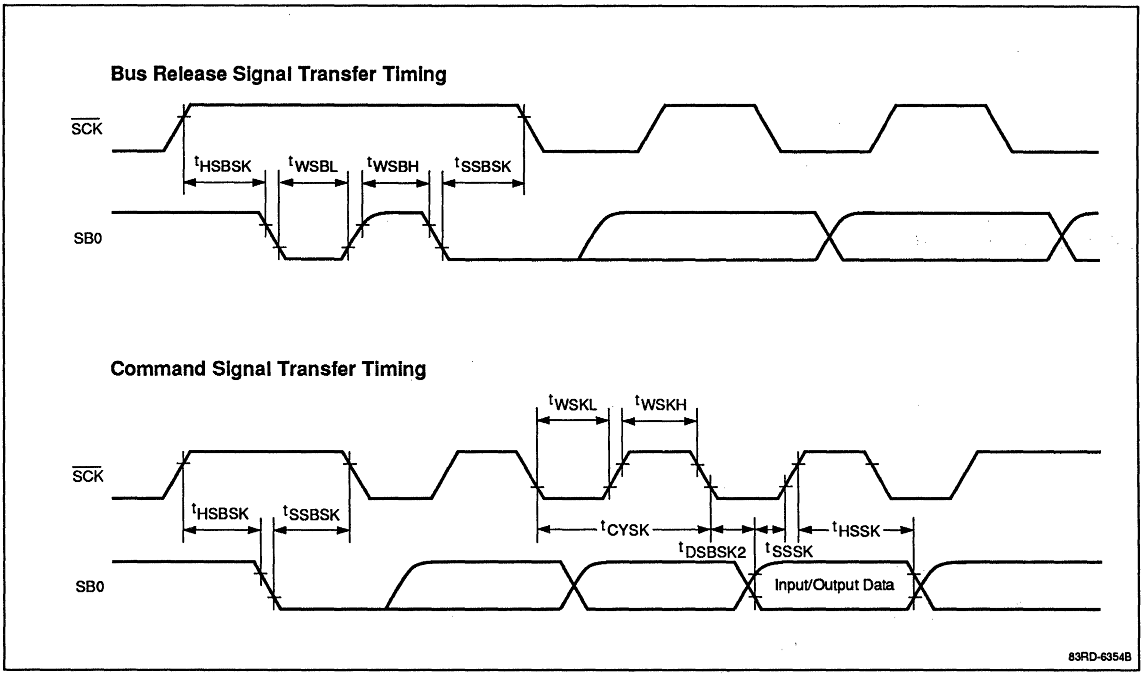
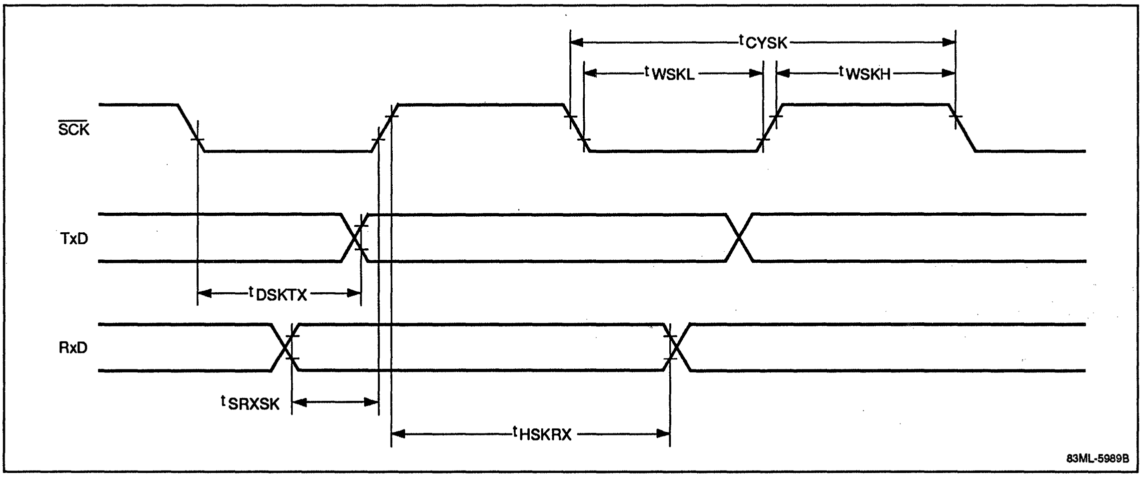


Figure 17. Asynchronous Mode Timing



Comparator Port Operation

Item	Symbol	Conditions	Min	Max	Unit
Comparison accuracy	V_{ACOMP}	–		100	mV
		μPD78P224		100	mV
Comparison time	t_{COMP}		128	256	t_{CYX}
Sampling time	t_{SAMP}		62		t_{CYX}
PT input voltage	V_{IPT}		0	V_{DD}	V

Interrupt Timing Operation

Item	Symbol	Conditions	Min	Max	Unit
NMI low-level width	t_{WNIL}		10		μs
NMI high-level width	t_{WNIH}		10		μs
INTP0-INTP6 low-level width	t_{WITL}		24		t_{CYX}
INTP0-INTP6 high-level width	t_{WITH}		24		t_{CYX}
$\overline{\text{RESET}}$ low-level width	t_{WRSL}		10		μs
$\overline{\text{RESET}}$ high-level width	t_{WRSH}		10		μs

Figure 18. Interrupt Input Timing

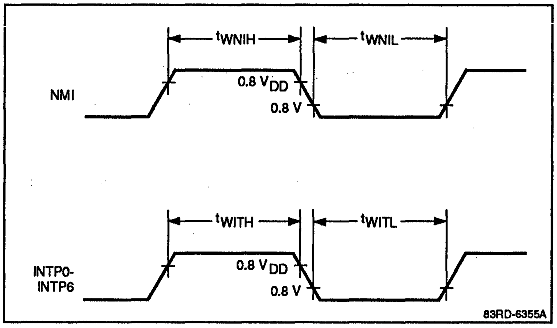
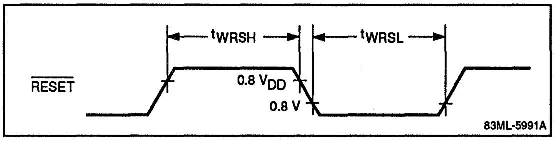


Figure 19. Reset Input Timing



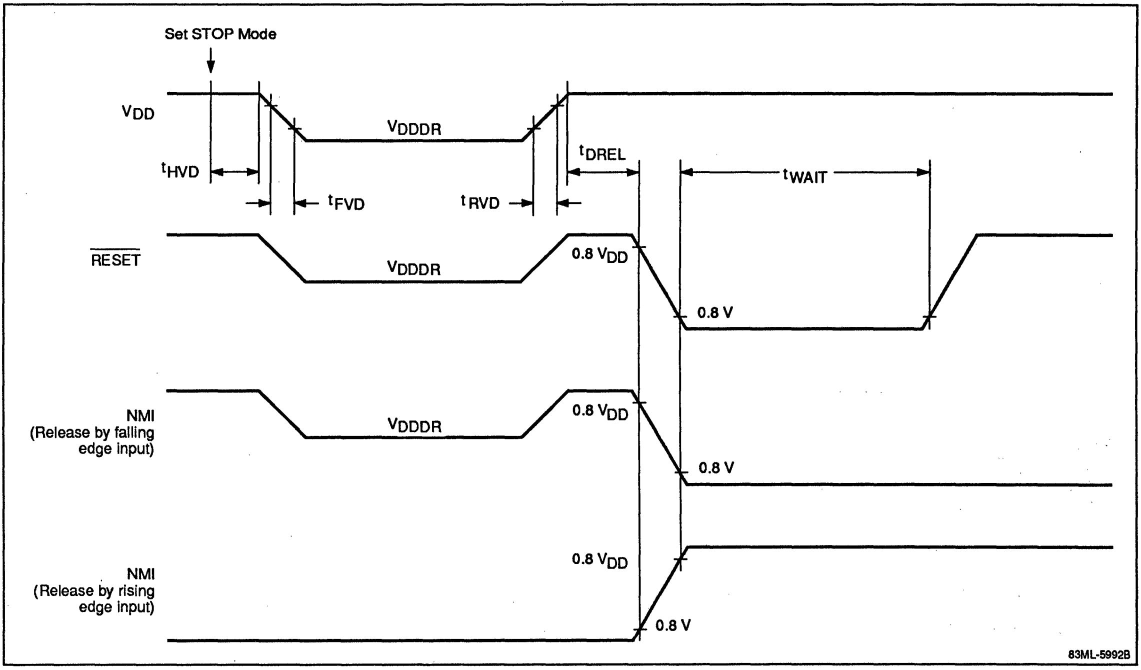
Data Retention Characteristics

Item	Symbol	Conditions	Min	Typ	Max	Unit
Data retention voltage	V_{DDDR}	STOP mode	2.5		5.5	V
Data retention current	I_{DDDR}	V_{DDDR} = 2.5 V		2	20	μA
		V_{DDDR} = 5 V ±10%		5	50	μA
V_{DD} rise time	t_{RVD}		200			μs
V_{DD} fall time	t_{FVD}		200			μs
V_{DD} retention time (for STOP mode setting)	t_{HVD}		0			ms
STOP release signal input time	t_{DREL}		0			ms
Oscillation stabilization wait time	t_{WAIT}	Crystal resonator	30			ms
		Ceramic resonator	5			ms
Low-level input voltage	V_{IL}	Note 1	0		0.1 V_{DDDR}	V
High-level input voltage	V_{IH}	Note 1	0.9 V_{DDDR}		V_{DDDR}	V

Notes:

(1) $\overline{\text{RESET}}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4, $P2_6$/INTP5, $P2_7$/INTP6/SI, $P3_2$/$\overline{\text{SCK}}$, $P3_3$/SO/SB0, and $\overline{\text{EA}}$ pins.

Figure 20. Data Retention Characteristics



Timing Dependent on t_{CYX}

Item	Symbol	Calculation Formula	Min/Max	12 MHz	Unit
X1 input clock cycle time	t_{CYX}		Min	82	ns
Address setup time to ASTB ↓	t_{SAST}	$t_{CYX}-30$	Min	52	ns
Address to $\overline{\text{RD}}$ ↓ delay time	t_{DAR}	$2t_{CYX}-35$	Min	129	ns
Address float time from $\overline{\text{RD}}$ ↓	t_{FAR}	$t_{CYX}/2-30$	Min	11	ns
Address to data input time	t_{DAID}	$(4+2n)\,t_{CYX}-100$	Max	228	ns
ASTB ↓ to data input time	t_{DSTID}	$(3+2n)\,t_{CYX}-65$	Max	181	ns
$\overline{\text{RD}}$ ↓ to data input time	t_{DRID}	$(2+2n)\,t_{CYX}-65$	Max	99	ns
ASTB ↓ to $\overline{\text{RD}}$ ↓ delay time	t_{DSTR}	$t_{CYX}-30$	Min	52	ns
$\overline{\text{RD}}$ ↑ to address active time	t_{DRA}	$2t_{CYX}-40$	Min	124	ns
$\overline{\text{RD}}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}-40$	Min	124	ns
$\overline{\text{RD}}$ low-level width	t_{WRL}	$(2+2n)\,t_{CYX}-40$	Min	124	ns
ASTB high-level width	t_{WSTH}	$t_{CYX}-30$	Min	52	ns
Address to $\overline{\text{WR}}$ ↓ delay time	t_{DAW}	$2t_{CYX}-35$	Min	129	ns
ASTB ↓ to data output time	t_{DSTOD}	$t_{CYX}+60$	Max	142	ns
ASTB ↓ to $\overline{\text{WR}}$ ↓ delay time	t_{DSTW1}	$t_{CYX}-30$	Min	52	ns
	t_{DSTW2}	$2t_{CYX}-35$ (refresh mode)	Min	129	ns
Data setup time to $\overline{\text{WR}}$ ↑	t_{SODWR}	$(3+2n)\,t_{CYX}-100$	Min	146	ns
Data setup time to $\overline{\text{WR}}$ ↓	t_{SODWF}	$t_{CYX}-60$ (refresh mode)	Min	22	ns
$\overline{\text{WR}}$ ↑ to ASTB ↑ delay time	t_{DWST}	$t_{CYX}-40$	Min	42	ns
$\overline{\text{WR}}$ low-level width	t_{WWL1}	$(3+2n)\,t_{CYX}-50$	Min	196	ns
	t_{WWL2}	$(2+2n)\,t_{CYX}-50$ (refresh mode)	Min	114	ns
Address to $\overline{\text{WAIT}}$ ↓ input time	t_{DAWT}	$3t_{CYX}-100$	Max	146	ns
ASTB ↓ to $\overline{\text{WAIT}}$ ↓ input time	t_{DSTWT}	$2t_{CYX}-80$	Max	84	ns

Notes:
(1) n indicates the number of wait states.

Figure 21. Recommended Oscillator Circuit

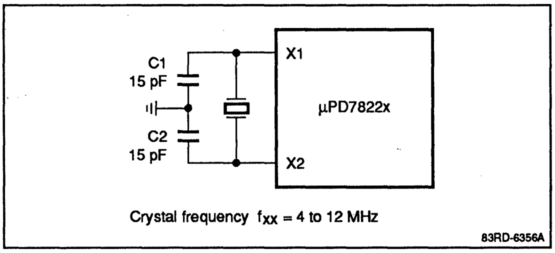
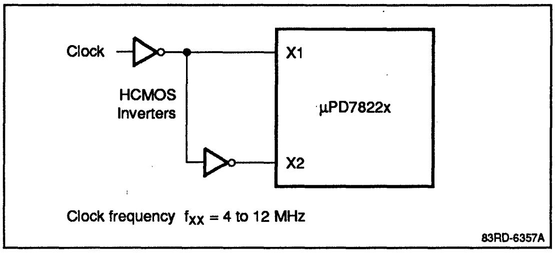


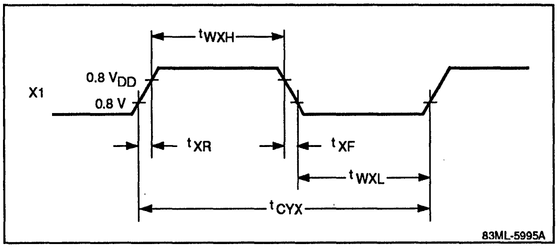
Figure 22. Recommended External Clock Circuit



External Clock Operation

Item	Symbol	Conditions	Min	Max	Unit
X1 input low-level width	t_{WXL}		30	130	ns
X1 input high-level width	t_{WXH}		30	130	ns
X1 input rise time	t_{XR}		0	30	ns
X1 input fall time	t_{XF}		0	30	ns
X1 input clock cycle time	t_{CYX}		82	250	ns

Figure 23. External Clock Timing



μPD78P224 PROGRAMMING

In the 78P224, the mask ROM of 78224 is replaced by a one-time programmable ROM (OTP ROM. The ROM is 16,384 × 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P224GJ/L are the socket adaptors used for configuring the μPD78P224 to fit a standard PROM socket.

Refer to tables 3 through 6 and figures 24 and 25 for special information applicable to PROM programming.

Table 3. Pin Functions During PROM Programming

Pin		Function
$P0_0$-$P0_7$	A_0-A_7	Input pins for PROM write/verify operations
$P5_0/A_8$	A_8	Input pin for PROM write/verify operation
$P2_1$/INTPO	A_9	Input pin for PROM write/verify operation
$P5_2$-$P5_6/A_{10}$-A_{14}	A_{10}-A_{14}	Input pins for PROM write/verify operations
$P4_0$-$P4_7/AD_0$-AD_7	D_0-D_7	Data pins for PROM write/verify operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobe data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enable a data read from the PROM
$P2_0$/NMI	NMI	PROM programming mode is entered by applying a high voltage to this pin
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
EA	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{ss}	V_{ss}	Ground

Table 4. Summary of Operation Modes for PROM Programming

Mode	NMI	$\overline{RESET}$	$\overline{CE}$	$\overline{OE}$	V_{pp}	V_{DD}	D_0-D_7
Program write	+12.5 V	L	L	H	+12.5 V	+6 V	Data input
Program verify	+12.5 V	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	+12.5 V	L	H	H	+12.5 V	+6 V	High Z
Read out	+12.5 V	L	L	L	+5 V	+5 V	Data output
Output disable	+12.5 V	L	L	H	+5 V	+5 V	High Z
Standby	+12.5 V	L	H	L/H	+5 V	+5 V	High Z

Notes:
When +12.5 V is applied to V_{pp} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

Table 5. DC Programming Characteristics

T_A = 25 ±5°C, V_{IP} = 12.5 ± 0.5 V applied to NMI pin, V_{ss} = 0 V.

Parameter	Symbol	Symbol*	Condition	Min	Typ	Max	Unit
High-level input voltage	V_{IH}	V_{IH}		2.4		V_{DDP} +0.3	V
Low-level input voltage	V_{IL}	V_{IL}		−0.3		0.8	V
Input leakage current	V_{LIP}	V_{LI}	V_I = 0 to V_{DDP}			10	μA
High-level output voltage	V_{OH1}	V_{OH}	I_{OH} = −400 μA	2.4			V
	V_{OH2}	V_{OH2}	I_{OH} = −100 μA	V_{DD} − 0.7			V
Low-level output voltage	V_{OL}	V_{OL}	I_{OH} = 2.1 mA			0.45	V
Output leakage current	I_{LO}		V_O = 0 to V_{DPP}; $\overline{OE}$ = V_{IH}			10	μA
NMI pin high-voltage input current	I_{IP}					±10	μA
V_{DDP} power voltage	V_{DDP}	V_{CC}	Program memory write mode	5.75	6.0	6.25	V
			Program memory read mode	4.5	5.0	5.5	V
V_{PP} power voltage	V_{PP}	V_{PP}	Program memory write mode	12.2	12.5	12.8	V
			Program memory read mode		V_{PP} = V_{DDP}		V
V_{DDP} power current	I_{DD}	I_{CC}	Program memory write mode		5	30	mA
			Program memory read mode $\overline{CE}$ = V_{IL}, V_I = V_{IH}		5	30	mA
V_{PP} power current	I_{PP}	I_{PP}	Program memory write mode $\overline{CE}$ = V_{IL}, $\overline{OE}$ = V_{IH}		5	30	mA
			Program memory read mode		1	100	μA

* Corresponding symbols of the μPD27C256A.

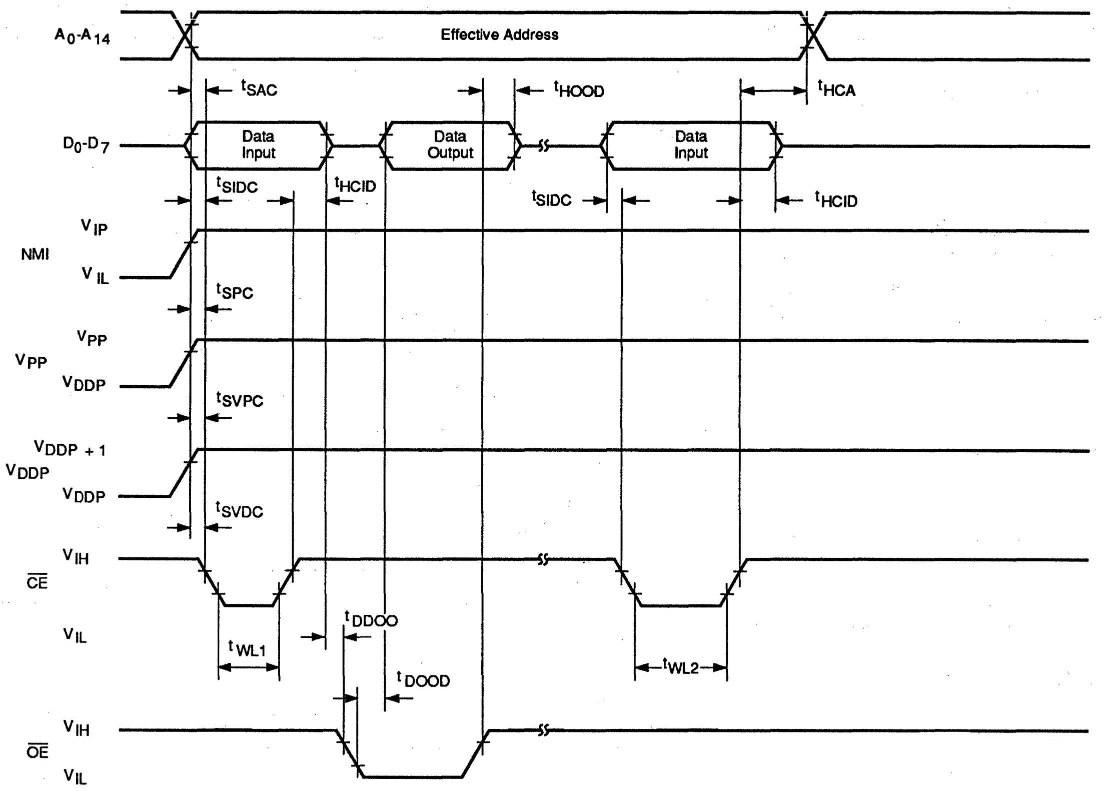
Table 6. AC Programming Characteristics

T_A = 25 ±5°C, V_{IP} = 12.5 ± 0.5 V applied to NMI pin, V_{ss} = 0 V, V_{DD} = 6 ±0.25 V, V_{PP} = 12.5 ±0.3 V.

Parameter	Symbol	Symbol*	Condition	Min	Typ	Max	Unit
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}		2			μs
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}		2			μs
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}		2			μs
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}		2			μs
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}		2			μs
Output data hold time from $\overline{OE}$ ↑	t_{HOOD}	t_{DF}		0		130	ns
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}		1			ms
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}		1			ms
Initial program pulse width	t_{WL1}	t_{PW}		0.95	1.0	1.05	ms
Additional program pulse width	t_{WL2}	t_{OPW}		2.85		78.75	ms
NMI high-voltage input setup time to $\overline{CE}$ ↓	t_{SPC}			2			μs
Address to data output time	t_{DAOD}	t_{ACC}	$\overline{CE}$ = $\overline{OE}$ = V_{IL}			200	ns
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}	$\overline{OE}$ = V_{IL}			200	ns
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}	$\overline{CE}$ = V_{IL}			75	ns
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	$\overline{CE}$ = V_{IL}	0		60	ns
Data hold time from address	t_{HAOD}	t_{OH}	$\overline{CE}$ = $\overline{OE}$ = V_{IL}	0			ns

* Corresponding symbols of the μPD27C256A.

Figure 24. PROM Write Mode Timing



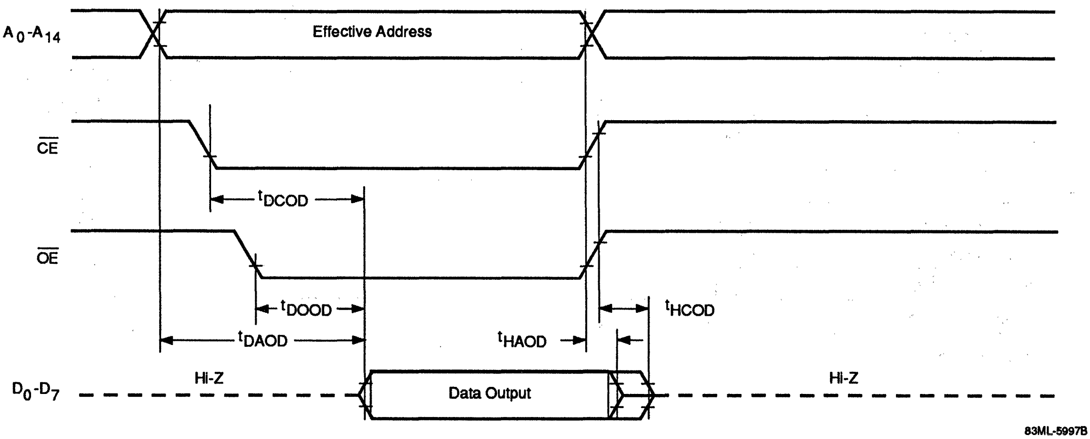
Notes:

(1) VDDP must be applied before applying V PP. It should be removed after removing V PP.

(2) VPP must not exceed +13 V, including overshoot.

83ML-5996B

Figure 25. PROM Read Mode Timing



83ML-5997B

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin to a low level and apply +12.5 V to the NMI pin.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{pp} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This bit is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps 4 to 6.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Fix the $\overline{\text{RESET}}$ pin to a low level and apply +12.5 V to the NMI pin.

(2) Apply +5 V to the V_{DD} and V_{pp} pins.

(3) Input the address of the data to be read to pins A_0-A_{14}.

(4) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(5) Data is output to the D_0 to D_7 pins.

INSTRUCTION SET

All microcomputers in the μPD7822x family have a 1-byte instruction lookahead buffer. This allows the first byte of the next opcode in program memory to be fetched while the current opcode is being executed. This pipeline architecture allows instruction fetch and excute cycles to overlap. An instruction can be fetched from program memory while data is being read from or written to RAM or an I/O port.

The advantage of the pipeline is that one instruction can be executed while another is being fetched, virtually halving the time required for these two operations and thereby reducing overall program execution time.

Operands and Operations

Refer to tables 7 and 8 for the meanings of symbols in the operand and operations columns of the Instruction Set table.

Specify operands in accordance with the rules of operand representation; for details, refer to the assembler specifications. If two or more description methods are available, select one. The symbols +, –, #, !, $, /, [ ], and & are keywords and must be used in conjunction with each instruction.

When describing immediate data as a label, use one of the following modifiers: +, –, #, !, $, /, [ ], and &. Symbols r and rp can be described in both the function name and absolute name.

Table 7. Operands

Symbol	Meaning
+	Autoincrement
–	Autodecrement
#	Immediate data
!	Absolute address
$	Relative address
/	Bit inversion
[ ]	Indirect addressing
&	Subbank; 1M-byte expansion space
r	Register Function name: X, A, C, B, E, D, L, H Absolute name: R0 to R7
r1	Register group 1: C, B
rp	Register pair Function name: AX, BC, DE, HL Absolute name: RP0 to RP3
sfr	Special function register: P0, P2-P7, P0H, P0L, RTPC, CR10, CR11, CR20, CR21, CR22, CR30, PM0, PM3, PM5, PM6, PMC3, PUO, CRC0-CRC2, TOC, TM1-TM3, TMC0, TMC1, PRM0, PRM1, ADM, ADCR, CSIM, SBIC, SIO, ASIM, ASIS, RxB, TxS, BRGC, STBC (dedicated instruction only), MM, PW, RFM, IF0L, IF0H, MK0L, MK0H, PR0L, PR0H, ISM0L, ISM0H, INTM0, INTM1, IST

Table 7. Operands (cont)

Symbol	Meaning
sfrp	Special function register pair: CR00-CR02, TM0, IF0, MK0, PR0, ISM0
mem	Memory address indirectly addressed Register indirect mode: [DE], [HL], [DE+], [HL+], [DE–], [HL–] Base mode: [DE + byte], [HL + byte], [SP + byte] Indexed mode: word[A], word[B], word[DE], word[HL]
mem1	Memory addressed by means of indirect addressing group 1: [DE], [HL]
saddr	Memory address indirectly addressed: FE20H-FF1FH immediate data or label
saddrp	Memory address addressed by means of direct addressing pair: FE20H-FF1EH immediate data (LSB=0; odd address) or label
addr16	16-bit address: 0000H-FEFFH immediate data or label
addr11	11-bit address: 800H-FFFH immediate data or label
addr5	5-bit address: 40H-7EH immediate data or label
word	16-bit data: 16-bit immediate data or label
byte	8-bit data: 8-bit immediate data or label
bit	3-bit data: 3-bit immediate data or label
n	Number of shift bits: 3-bit immediate data (0-7)
RBn	Register bank: RB0-RB3

Table 8. Registers and Flags

Symbol	Meaning
A	A register; 8-bit accumulator
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R7	Registers 0 to 7 (absolute names)
AX	Register pair (AX); 16-bit accumulator
BC	Register pair (BC)
DE	Register pair (DE)
HL	Register pair (HL)
RP0-RP3	Register pairs 0 to 3 (absolute names)
PC	Program counter
SP	Stack pointer
PSW	Program status word
CY	Carry flag
AC	Auxiliary flag
Z	Zero flag
RBS1-RBS0	Register bank select flags
IE	Interrupt enable flag
STBC	Standby control register
( )	Memory contents indicated by address or register contents in ( )
xxH	Hexadecimal number
x_H, x_L	Higher 8 bits and lower 8 bits of 16-bit register pair

Clocks

The clock field specifies the number of clocks required under the conditions defined by the four column headings as follows:

IROM	Program in internal ROM is executed.
IRAM	Program in external ROM is executed and internal RAM is accessed.
SFR	Program in external ROM is executed and special function register is accessed.
EMEM	Program in external ROM is executed and external memory is accessed.

In a shift-rotate instruction, n in the clock field indicates the number of bits by which data is shifted.

The hyphen (-) indicates a range of values; for example 10-13 means 10, 11, 12, or 13.

The virgule symbol (/) means either/or; for example, a/b means either a or b.

The number of clocks when execution is branched by a conditional branch instruction is shown after the symbol (/).

The number of clocks for instruction having the saddr or saddrp operand and when an SFR is accessed with FF00H to FFFFH described as saddr or saddrp is shown after the symbol (/).

Bytes and Clocks

The number of bytes and clocks for instructions with a mem or &mem operand depends on the particular instruction and the memory addressing mode (register indirect, base, or indexed). Table 9 is applicable when the program in internal ROM is executed (ROM clock column of the Instruction Set table). Table 10 is applicable when the program in external ROM is executed (IRAM, SFR, and EMEM clock columns).

Flags

The symbols in the flag field have the following meanings.

Blank	No change
0	Cleared to 0
1	Set to 1
x	Set or cleared depending on the result
R	Value previously saved is restored

Operation Codes

Table 11 defines the symbols used in the operation code field.

Registers and Register Pairs. The r, rl, and rp operands are specified in the opcode by one or more bits as shown in figure 26. For example, 001 as bits $R_2R_1R_0$ (or $R_6R_5R_4$) specifies register A.

In the first and second operands are registers or register pairs; the higher 4 bits of the register specification byte define the first operand and the lower 4 bits define the second operand. For example, in the MOV A,L instruction (transfer L register contents to register A), the second byte of the opcode is obtained from figure 26 as shown below.

Instruction	Opcode, Bytes 1 and 2
MOV r,r	0 0 1 0 0 1 0 0 0 $R_6R_5R_4$ 0 $R_2R_1R_0$
MOV A,L	0 0 1 0 0 1 0 0 0 0 0 1 0 1 1 0

Memory Addressing Modes. The 3-bit mem code and the 5-bit mod code are selected from figure 27 according to the description of mem in the operand field (table 7).

A MOV instruction with register indirect mode specified for mem is a special 1-byte instruction. When base mode or indexed mode is specified for mem, the 8-bit or 16-bit offset data corresponding to byte and word, respectively, is added from the third byte onward.

The opcode for an &mem or &mem1 operand is modified by inserting a 01H code as the first byte preceding the first-byte code listed in the Instruction Set table. Subsequent bytes are as shown in the table.

Figure 26. Opcodes for Registers (r, r1, rp)

r

R_2 R_6	R_1 R_5	R_0 R_4	reg	
0	0	0	R0	X
0	0	1	R1	A
0	1	0	R2	C
0	1	1	R3	B
1	0	0	R4	E
1	0	1	R5	D
1	1	0	R6	L
1	1	1	R7	H

r1

R_0	reg
0	C
1	B

rp

P_1 P_2 P_6	P_0 P_1 P_5	reg-pair	
0	0	RP0	AX
0	1	RP1	BC
1	0	RP2	DE
1	1	RP3	HL

83ML-5998A

Figure 27. Opcodes for Memory Addressing Modes (mem, mod)

Mod / Mem	1 0110	0 0110	0 1010
	Register Indirect Mode	Base Mode	Index Mode
0 0 0	[DE+]	[DE+byte]	word [DE]
0 0 1	[HL+]	[SP+byte]	word [A]
0 1 0	[DE-]	[HL+byte]	word [HL]
0 1 1	[HL-]	-	word [B]
1 0 0	[DE]	-	-
1 0 1	[HL]	-	-

83ML-5999A

Table 9. Bytes and Clocks for Instructions With "mem" and "&mem" Operands; Internal ROM (IROM)

			Register Indirect Mode		Base Mode		Indexed Mode
	Instruction		[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE+byte] [HL+byte]	[SP+byte]	word[A] word[B] word[DE] word[HL]
Bytes		mem	1/2*	1/2*	3	3	4
		&mem	2/3*	2/3*	4	4	5
Clock Cycles	MOV	A, mem	6/8	6/8	8-11	9-12	8-11
		mem, A					
		A, &mem	8/10	8/10	10-13	11-14	10-13
		&mem, A					
	XCH	A, mem	11-15	9-13	10-15	11-16	10-15
		A, &mem	13-17	11-15	12-17	13-18	12-17
	ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	10/12	8/12	9/12	10-13	9-12
		A, &mem	12/14	10/14	11/14	12-15	11-14

* When internal RAM is accessed with an instruction having a mem operand, the number of bytes is the number before the symbol (/). When the external memory (including the SFR area) is accessed, the number of bytes is the number after the symbol (/).

Table 10. Bytes and Clocks for Instructions With "mem" and "&mem" Operands; External ROM (IRAM, SFR, EMEM)

			Register Indirect Mode		Base Mode		Indexed Mode
Instruction			[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE+byte] [HL+byte]	[SP+byte]	word[A] word[B] word[DE] word[HL]
Bytes		mem	2*	2*	3	3	4
		&mem	3*	3*	4	4	5
Clock Cycles	MOV	A, mem	9/11	6/8	11/13	12/14	14/16
		mem, A					
		A, &mem	12/14	9/11	14/16	15/17	17/19
		&mem, A					
	XCH	A, mem	14/18	12/16	13/17	14/18	16/20
		A, &mem	17/21	15/19	16/20	17/21	19/23
	ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	13/15	11/13	12/14	13/15	15/17
		A, &mem	16/18	14/16	15/17	16/18	18/20

* When [DE], [HL], [DE+], [HL+], [DE–], or [HL–] is specified as the mem operand of a MOV instruction, the instruction is used as a dedicated 1-byte type. When the operand is &mem, the instruction is 2-byte.

Table 11. Opcode Symbols

Symbol	Meaning
Bn	nth bit of immediate data B
Nn	nth bit of immediate data N
Data	8-bit immediate data corresponding to byte
Low/High Byte	16-bit immediate data corresponding to word
Saddr-offset	Lower 8-bit offset data of 16-bit address corresponding to saddr
Sfr-offset	Lower 8-bit offset data of 16-bit address of special function register (sfr)
Low/High Offset	16-bit offset data corresponding to word in indexed addressing
Low/High Addr	16-bit immediate data corresponding to addr16
jdisp	Signed 2's complement data (8 bits) indicating relative address distance between first address of next instruction and branch destination address
fa	Lower 11 bits of immediate data corresponding to addr11
ta	Lower 5 bits of immediate data corresponding to (addr5xdis)

Instruction Set

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z AC CY	Bytes B1 thru B5
8-Bit Data Transfer									
MOV	r,#byte	r ← byte	2	2	6				1 0 1 1 1 R_2 R_1 R_0
									Data
	saddr,#byte	(saddr) ← byte	3	3/5	9	9	12		0 0 1 1 1 0 1 0
									Saddr-offset
									Data
	sfr,#byte	sfr ← byte	3	5		9	12		0 0 1 0 1 0 1 1
									Sfr-offset
									Data
	r,r	r ← r	2	2	6				0 0 1 0 0 1 0 0
									0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,r	A ← r	1	2	3				1 1 0 1 0 R_2 R_1 R_0
	A,saddr	A ← (saddr)	2	2/4	6	6	9		0 0 1 0 0 0 0 0
									Saddr-offset
	saddr,A	(saddr) ← A	2	3/5	6	8			0 0 1 0 0 0 1 0
									Saddr-offset
	saddr, saddr	(saddr) ← (saddr)	3	3-7	9				0 0 1 1 1 0 0 0
									Saddr-offset
									Saddr-offset
	A,sfr	A ← sfr	2	4		6			0 0 0 1 0 0 0 0
									Sfr-offset
	sfr,A	sfr ← A	2	5		6			0 0 0 1 0 0 1 0
									Sfr-offset
	A,mem	A ← (mem)	1-4	6-12	6-14	8-16	8-16		* 0 1 0 1 1 mem
									0 0 0 mod
									0 mem 0 0 0 0
									Low Offset
									High Offset
	A,&mem	A ← (&mem)	2-5	8-14	9-17	11-19	11-19		* 0 0 0 0 0 0 0 1
									0 1 0 1 1 mem
									0 0 0 0 0 0 0 1
									0 0 0 mod
									0 mem 0 0 0 0
									Low Offset
									High Offset

Note:

* If [DE], [HL], [DE+], [DE−], [HL+] or [HL−] is described as mem, these instructions are used as dedicated 1-byte codes. If the register name is described as &mem, the instructions are used as dedicated 2-byte codes.

Instruction Set (cont)

				Clocks				Flags			
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Operation Code (Bits 7-0) Bytes B1 thru B5**
8-Bit Data Transfer (cont)											
MOV	mem,A	(mem) ← A	1-4	6-12	6-14	8-16	8-16				* 0 1 0 1 0 mem
											0 0 0 mod
											1 mem 0 0 0 0
											Low Offset
											High Offset
	&mem,A	(&mem) ← A	2-5	8-14	9-17	11-19	11-19				* 0 0 0 0 0 0 0 1
											0 1 0 1 0 mem
											0 0 0 0 0 0 0 1
											0 0 0 mod
											1 mem 0 0 0 0
											Low Offset
											High Offset
	A,!addr16	A ← (!addr16)	4	6/8	14		16				0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 0
											Low Addr
											High Addr
	A,&!addr16	A ← (&laddr16)	5	8/10			19				0 0 0 0 0 0 0 1
											0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 0
											Low Addr
											High Addr
	!addr16, A	(!addr16) ← A	4	6/8	14		17				0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 1
											Low Addr
											High Addr
	&!addr16,A	(&laddr16) ← A	5	8/10			20				0 0 0 0 0 0 0 1
											0 0 0 0 1 0 0 1
											1 1 1 1 0 0 0 1
											Low Addr
											High Addr
	PSW,#byte	PSW ← byte	3	3	9	9	9	x	x	x	0 0 1 0 1 0 1 1
											1 1 1 1 1 1 1 0
											Data
	PSW,A	PSW ← A	2	2	6	6	6	x	x	x	0 0 0 1 0 0 1 0
											1 1 1 1 1 1 1 0
	A,PSW	A ← PSW	2	2	6	6	6				0 0 0 1 0 0 0 0
											1 1 1 1 1 1 1 0

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z AC CY**	**Bytes B1 thru B5**
8-Bit Data Transfer (cont)									
XCH	A,r	A ↔ r	1	4	4				1 1 0 1 1 R_2 R_1 R_0
	r,r	r ↔ r	2	3	6				0 0 1 0 0 1 0 1
									0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,mem	A ↔ (mem)	2-4	9-16	12-16		16-20		0 0 0 mod
									0 mem 0 1 0 0
									Low Offset
									High Offset
	A,&mem	A ↔ (&mem)	3-5	11-18	15-19		19-23		0 0 0 0 0 0 0 1
									0 0 0 mod
									0 mem 0 1 0 0
									Low Offset
									High Offset
	A,saddr	A ↔ (saddr)	2	4/8	6				0 0 1 0 0 0 0 1
									Saddr-offset
	A,sfr	A ↔ sfr	3	6/10		13			0 0 0 0 0 0 0 1
									0 0 1 0 0 0 0 1
									Sfr-offset
	saddr,saddr	(saddr) ↔ (saddr)	3	6-14		10			0 0 1 1 1 0 0 1
									Saddr-offset
									Saddr-offset
16-Bit Data Transfer									
MOVW	rp,#word	rp ← word	3	3	9				0 1 1 0 0 P_2 P_1 0
									Low Byte
									High Byte
	saddrp,#word	(saddrp) ← word	4	4/8	12	12	18		0 0 0 0 1 1 0 0
									Saddr-offset
									Low Byte
									High Byte
	sfrp,#word	sfrp ← word	4	8		12			0 0 0 0 1 0 1 1
									Saddr-offset
									Low Byte
									High Byte
	rp,rp	rp ← rp	2	4	6				0 0 1 0 0 1 0 0
									0 P_6 P_5 0 1 P_2 P_1 0
	AX,saddrp	AX ← (saddrp)	2	6/10	8	12			0 0 0 1 1 1 0 0
									Saddr-offset
	saddrp,AX	(saddrp) ← AX	2	5/9	8	12			0 0 0 1 1 0 1 0
									Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
16-Bit Data Transfer (cont)											
MOVW	AX,sfrp	AX ← sfrp	2	10		12					0 0 0 1 0 0 0 1
											Sfr-offset
	sfrp,AX	sfrp ← AX	2	9		12					0 0 0 1 0 0 1 1
											Sfr-offset
	AX,mem1	AX ← (mem1)	2	9-15	12	16	16				0 0 0 0 0 1 0 1
											1 1 1 0 0 0 1 R_0
	AX,&mem1	AX ← (&mem1)	3	11-17	15	19	19				0 0 0 0 0 0 0 1
											0 0 0 0 0 1 0 1
											1 1 1 0 0 0 1 R_0
	mem1, AX	(mem1) ← AX	2	8-14	11	15	15				0 0 0 0 0 1 0 1
											1 1 1 0 0 1 1 R_0
	&mem1,AX	(&mem1) ← AX	3	10-16	14	18	18				0 0 0 0 0 0 0 1
											0 0 0 0 0 1 0 1
											1 1 1 0 0 1 1 R_0
8-Bit Operation											
ADD	A,#byte	A,CY ← A + byte	2	2	6			x	x	x	1 0 1 0 1 0 0 0
											Data
	saddr,#byte	(saddr),CY ← (saddr) + byte	3	3/7	9	11		x	x	x	0 1 1 0 1 0 0 0
											Saddr-offset
											Data
	sfr,#byte	sfr,CY ← sfr + byte	4	9		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 0 0 0
											Sfr-offset
											Data
	r,r	r,CY ← r + r	2	3	7			x	x	x	1 0 0 0 1 0 0 0
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A + (saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 0 0
											Saddr-offset
	A,sfr	A,CY ← A + sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 0 0 0
											Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) + (saddr)	3	3-9	9	11		x	x	x	0 1 1 1 1 0 0 0
											Saddr-offset
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
ADD	A,mem	A,CY ← A + (mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 0 0 Low Offset High Offset
	A,&mem	A,CY ← A + (&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 0 0 Low Offset High Offset
ADDC	A,#byte	A,CY ← A + byte + CY	2	2	6			x	x	x	1 0 1 0 1 0 0 1 Data
	saddr,#byte	(saddr),CY ← (saddr) + byte + CY	3	3/7	9	11		x	x	x	0 1 1 0 1 0 0 1 Saddr-offset Data
	sfr,#byte	sfr,CY ← sfr + byte + CY	4	9		14		x	x	x	0 0 0 0 0 0 0 1 0 1 1 0 1 0 0 1 Sfr-offset Data
	r,r	r,CY ← r + r + CY	2	3	7			x	x	x	1 0 0 1 1 0 0 1 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A + (saddr) + CY	2	2/5	6	7	8	x	x	x	1 0 0 1 1 0 0 1 Saddr-offset
	A,sfr	A,CY ← A + sfr + CY	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 1 Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) + (saddr) + CY	3	3-9	9	11		x	x	x	0 1 1 1 1 0 0 1 Saddr-offset Saddr-offset
	A,mem	A,CY ← A + (mem) + CY	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 0 1 Low Offset High Offset
	A,&mem	A,CY ← A + (&mem) + CY	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 0 0 Low Offset High Offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Bytes B1 thru B5**
8-Bit Operation (cont)											
SUB	A,#byte	A,CY ← A−byte	2	2	6			x	x	x	1 0 1 0 1 0 1 0 Data
	saddr,#byte	(saddr),CY ← (saddr)−(byte)	3	3/7	9	11		x	x	x	0 1 1 0 1 0 1 0 Saddr-offset Data
	sfr,#byte	sfr,CY ← sfr−byte	4	9		14		x	x	x	0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 0 Sfr-offset Data
	r,r	r,CY ← r−r	2	3	7			x	x	x	1 0 0 0 1 0 1 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A−(saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 1 0 Saddr-offset
	A,sfr	A,CY ← A−sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 0 1 0 Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) − (saddr)	3	3-9	9	11		x	x	x	0 1 1 1 1 0 1 0 Saddr-offset Saddr-offset
	A,mem	A,CY ← A−(&mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 1 0 Low Offset High Offset
	A,&mem	A,CY ← A−(&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 1 0 Low Offset High Offset
SUBC	A,#byte	A,CY ← A−byte−CY	2	2	6			x	x	x	1 0 1 0 1 0 1 1 Data
	saddr,#byte	(saddr),CY ← (saddr)−byte−CY	3	3/7	9	11		x	x	x	0 1 1 0 1 0 1 1 Saddr-offset Data
	sfr,#byte	sfr,CY ← sfr−byte−CY	4	9		14		x	x	x	0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 1 Sfr-offset Data

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
SUBC	r,r	r,CY ← r−r−CY	2	3	7			x	x	x	1 0 0 0 1 0 1 1 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A−(saddr)−CY	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 1 1 Saddr-offset
	A,sfr	A,CY ← A−sfr−CY	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 0 1 1 Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr)−(saddr) −CY	3	3-9	9	11		x	x	x	0 1 1 1 1 0 1 1 Saddr-offset Saddr-offset
	A,mem	A,CY ← A−(mem)−CY	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 1 1 Low Offset High Offset
	A,&mem	A,CY ← A−(&mem)−CY	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 1 1 Low Offset High Offset
AND	A,#byte	A ← A∧byte	2	2	6			x			1 0 1 0 1 1 0 0 Data
	saddr,#byte	(saddr) ← (saddr)∧byte	3	3/7	9	11		x			0 1 1 0 1 1 0 0 Saddr-offset Data
	sfr,#byte	sfr ← sfr∧byte	4	9		14		x			0 0 0 0 0 0 0 1 0 1 1 0 1 1 0 0 Sfr-offset Data
	r,r	r ← r∧r	2	3	7			x			1 0 0 0 1 1 0 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∧(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 0 0 Saddr-offset
	A,sfr	A ← A∧(sfr)	3	7		10		x			0 0 0 0 0 0 0 1 1 0 0 1 1 1 0 0 Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∧(saddr)	3	3-9	9	11		x			0 1 1 1 1 1 0 0 Saddr-offset Saddr-offset

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z AC CY	Bytes B1 thru B5
8-Bit Operation (cont)									
AND	A,mem	A ← A∧(mem)	2-4	8-13	11-15	13-17	13-17	x	0 0 0 mod 0 mem 1 1 0 0 Low Offset High Offset
	A,&mem	A ← A∧(&mem)	3-5	10-15	14-18	16-20	16-20	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 0 0 Low Offset High Offset
OR	A,#byte	A ← A∨byte	2	2	6			x	1 0 1 0 1 1 1 0 Data
	saddr,#byte	(saddr) ← (saddr) ∨ byte	3	3/7	9	11		x	0 1 1 0 1 1 1 0 Saddr-offset Data
	sfr,#byte	sfr ← sfr ∨ byte	4	9		14		x	0 0 0 0 0 0 0 1 0 1 1 0 1 1 1 0 Sfr-offset Data
	r,r	r ← r∨r	2	3	7			x	1 0 0 0 1 1 1 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∨(saddr)	2	3/5	6	7	8	x	1 0 0 1 1 1 1 0 Saddr-offset
	A,sfr	A ← A∨sfr	3	7		10		x	0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 0 Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∨(saddr)	3	3-9	9	11		x	0 1 1 1 1 1 1 0 Saddr-offset Saddr-offset
	A,mem	A ← A∨(mem)	2-4	8-13	11-15	13-17	13-17	x	0 0 0 mod 0 mem 1 1 1 0 Low Offset High Offset
	A,&mem	A ← A∨(&mem)	3-5	10-15	14-18	16-20	16-20	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 1 0 Low Offset High Offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
XOR	A,#byte	A ← A∀byte	2	2	6			x			1 0 1 0 1 1 0 1
											Data
	saddr,#byte	(saddr) ← (saddr)∀byte	3	3/5	9	11		x			0 1 1 0 1 1 0 1
											Saddr-offset
											Data
	sfr,#byte	sfr ← sfr∀byte	4	7		14		x			0 0 0 0 0 0 0 1
											0 1 1 0 1 1 0 1
											Sfr-offset
											Data
	r,r	r ← r∀r	2	3	7			x			1 0 0 0 1 1 0 1
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∀(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 0 1
											Saddr-offset
	A,sfr	A ← A∀(sfr)	3	7		10		x			0 0 0 0 0 0 0 1
											1 0 0 1 1 1 0 1
											Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∀(saddr)	3	3-9	9	11		x			0 1 1 1 1 1 0 1
											Saddr-offset
											Saddr-offset
	A,mem	A ← A∀(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod
											0 mem 1 1 0 1
											Low Offset
											High Offset
	A,&mem	A ← A∀(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 1 0 1
											Low Offset
											High Offset
CMP	A,#byte	A − byte	2	2	6			x	x	x	1 0 1 0 1 1 1 1
											Data
	saddr,#byte	(saddr) − byte	3	3/5	9	11		x	x	x	0 1 1 0 1 1 1 1
											Saddr-offset
											Data
	sfr,#byte	sfr − byte	4	7		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 1 1 1
											Sfr-offset
											Data

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z AC CY	Bytes B1 thru B5
8-Bit Operation (cont)									
CMP	r,r	r − r	2	3	7			x x x	1 0 0 0 1 1 1 1
									0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A − (saddr)	2	3/5	6	7	8	x x x	1 0 0 1 1 1 1 1
									Saddr-offset
	A,sfr	A − sfr	3	7		10		x x x	0 0 0 0 0 0 0 1
									1 0 0 1 1 1 1 1
									Sfr-offset
	saddr,saddr	(saddr) − (saddr)	3	3-7	9	11		x x x	0 1 1 1 1 1 1 1
									Saddr-offset
									Saddr-offset
	A,mem	A − (mem)	2-4	8-13	11-15	13-17	13-17	x x x	0 0 0 mod
									0 mem 1 1 1 1
									Low Offset
									High Offset
	A,&mem	A − (&mem)	3-5	10-15	14-18	16-20	16-20	x x x	0 0 0 0 0 0 0 1
									0 0 0 mod
									0 mem 1 1 1 1
									Low Offset
									High Offset
16-Bit Operation									
ADDW	AX,#word	AX,CY ← AX + word	3	4	9			x x x	0 0 1 0 1 1 0 1
									Low Byte
									High Byte
	AX,rp	AX,CY ← AX + rp	2	6	8			x x x	1 0 0 0 1 0 0 0
									0 0 0 0 1 P_2 P_1 0
	AX,saddrp	AX,CY ← AX + (saddrp)	2	7/11	9	13		x x x	0 0 0 1 1 1 0 1
									Saddr-offset
	AX,sfrp	AX,CY ← AX + sfrp	3	13		16		x x x	0 0 0 0 0 0 0 1
									0 0 0 1 1 1 0 1
									Sfr-offset
SUBW	AX,#word	AX,CY ← AX − word	3	4	9			x x x	0 0 1 0 1 1 1 0
									Low Byte
									High Byte
	AX,rp	AX,CY ← AX − rp	2	6	8			x x x	1 0 0 0 1 0 1 0
									0 0 0 0 1 P_2 P_1 P_0
	AX,saddrp	AX,CY ← AX − (saddrp)	2	7/11	9	13		x x x	0 0 0 1 1 1 1 0
									Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Bytes B1 thru B5**
16-Bit Operation (cont)											
SUBW	AX,sfrp	AX,CY ← AX−sfrp	3	13		16		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 1 0
											Sfr-offset
CMPW	AX,#word	AX−word	3	3	9			x	x	x	0 0 1 0 1 1 1 1
											Low Byte
											High Byte
	AX,rp	AX−rp	2	5	7			x	x	x	1 0 0 0 1 1 1 1
											0 0 0 0 1 P_2 P_1 0
	AX,saddrp	AX−(saddrp)	2	6/10	8	12		x	x	x	0 0 0 1 1 1 1 1
											Saddr-offset
	AX,sfrp	AX−sfrp	3	12		15		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 1 1
											Sfr-offset
Multiplication/Division											
MULU	r	AX ← Axr	2	22	24						0 0 0 0 0 1 0 1
											0 0 0 0 1 R_2 R_1 R_0
DIVUW	r	AX(quotient), r(remainder) ← AX ÷ r	2	71	76						0 0 0 0 0 1 0 1
											0 0 0 1 1 R_2 R_1 R_0
Increment/Decrement											
INC	r	r ← r + 1	1	2	3			x	x		1 1 0 0 0 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) + 1	2	2/6	6	7		x	x		0 0 1 0 0 1 1 0
											Saddr-offset
DEC	r	r ← r − 1	1	2	3			x	x		1 1 0 0 1 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) − 1	2	2/6	6	7		x	x		0 0 1 0 0 1 1 1
											Saddr-offset
INCW	rp	rp ← rp + 1	1	3	3						0 1 0 0 0 1 P_1 P_0
DECW	rp	rp ← rp − 1	1	3	3						0 1 0 0 1 1 P_1 P_0
Shift/Rotate											
ROR	r,n	$(CY, r_7 \leftarrow r_0, r_{m-1} \leftarrow r_m)$ xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 0
											0 1 N_2 N_1 N_0 R_2 R_1 R_0
ROL	r,n	$(CY, r_0 \leftarrow r_7, r_{m+1} \leftarrow r_m)$ xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 1
											0 1 N_2 N_1 N_0 R_2 R_1 R_0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Shift/Rotate (cont)											
RORC	r,n	(CY ← r_0, r_7 ← CY, r_{m-1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 0 0 0 N_2 N_1 N_0 R_2 R_1 R_0
ROLC	r,n	(CY ← r_7, r_0 ← CY, r_{m+1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 1 0 0 N_2 N_1 N_0 R_2 R_1 R_0
SHR	r,n	(CY ← r_0, r_7 ← 0, r_{m-1} ← r_m) xn times, n=0-7	2	3+2n	5+2n			x	0	x	0 0 1 1 0 0 0 0 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHL	r,n	(CY ← r_7, r_0 ← 0, r_{m+1} ← r_m) xn times, n=0-7	2	3+2n	5+2n			x	0	x	0 0 1 1 0 0 0 1 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHRW	rp,n	(CY ← rp_0, rp_{15} ← 0, rp_{m-1} ← rp_m) xn times, n=0-7	2	3+3n	5+3n			x	0	x	0 0 1 1 0 0 0 0 1 1 N_2 N_1 N_0 R_2 R_1 R_0
SHLW	rp,n	(CY ← rp_{15}, rp_0 ← 0, rp_{m+1} ← rp_m) xn times, n=0-7	2	3+3n	5+3n			x	0	x	0 0 1 1 0 0 0 1 1 1 N_2 N_1 N_0 R_2 R_1 R_0
ROR4	mem1	A_{3-0} ← $(mem1)_{3-0}$, $(mem1)_{7-4}$ ← A_{3-0}, $(mem1)_{3-0}$ ← $(mem1)_{7-4}$	2	24	26	34	34				0 0 0 0 0 1 0 1 1 0 0 0 1 1 R_1 0
	&mem1	A_{3-0} ← $(\&mem1)_{3-0}$, $(\&mem1)_{7-4}$ ← A_{3-0}, $(\&mem1)_{3-0}$ ← $(\&mem1)_{7-4}$	3	26	29	37	37				0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 0 1 1 R_1 0
ROL4	mem1	A_{3-0} ← $(mem1)_{7-4}$, $(mem1)_{3-0}$ ← A_{3-0}, $(mem1)_{7-4}$ ← $(mem1)_{3-0}$	2	25	27	35	35				0 0 0 0 0 1 0 1 1 0 0 1 1 1 R_1 0
	&mem1	A_{3-0} ← $(\&mem1)_{7-4}$, $(\&mem1)_{3-0}$ ← A_{3-0}, $(\&mem1)_{7-4}$ ← $(\&mem1)_{3-0}$	3	27	30	38	38				0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 1 1 1 R_1 0

6

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
BCD Adjustment											
ADJBA		Decimal adjust accumulator after addition	1	3		3		x	x	x	0 0 0 0 1 1 1 0
ADJBS		Decimal adjust accumulator after addition	–1	3		3		x	x	x	0 0 0 0 1 1 1 1
Bit Manipulation											
MOV1	CY,saddr.bit	CY ← (saddr bit)	3	5/7	9	9	11			x	0 0 0 0 1 0 0 0
											0 0 0 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,sfr.bit	CY ← sfr.bit	3	7		9				x	0 0 0 0 1 0 0 0
											0 0 0 0 1 B_2 B_1 B_0
											Sfr-offset
	CY,A.bit	CY ← A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 0 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 0 0 0 B_2 B_1 B_0
											Sfr-offset
	CY,PSW.bit	CY ← PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 0 0 0 0 B_2 B_1 B_0
	saddr.bit,CY	(saddr bit) ← CY	3	8/12	12	14	14				0 0 0 0 1 0 0 0
											0 0 0 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit,CY	sfr.bit ← CY	3	12		14					0 0 0 0 1 0 0 0
											0 0 0 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit,CY	A.bit ← CY	2	8	10						0 0 0 0 0 0 1 1
											0 0 0 1 1 B_2 B_1 B_0
	X.bit,CY	X.bit ← CY	2	8	10						0 0 0 0 0 0 1 1
											0 0 0 1 0 B_2 B_1 B_0
											Sfr-offset
	PSW.bit,CY	PSW.bit ← CY	2	7		9		x	x		0 0 0 0 0 0 1 0
											0 0 0 1 0 B_2 B_1 B_0
AND1	CY,saddr.bit	CY ← CY ∧ (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 0 1 0 0 B_2 B_1 B_0
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Bytes B1 thru B5**
Bit Manipulation (cont)											
AND1	CY,/saddr.bit	CY ← CY ∧ $\overline{\text{(saddr.bit)}}$	3	5/7	9	11				x	0 0 0 0 1 0 0 0 0 0 1 1 0 B_2 B_1 B_0 Sfr-offset
	CY,sfr.bit	CY ← CY ∧ sfr.bit	3	7		11				x	0 0 0 0 1 0 0 0 0 0 1 0 1 B_2 B_1 B_0 Sfr-offset
	CY,/sfr.bit	CY ← CY ∧ $\overline{\text{sfr.bit}}$	3	7		11				x	0 0 0 0 1 0 0 0 0 0 1 1 1 B_2 B_1 B_0
	CY,A.bit	CY ← CY ∧ A.bit	2	5	7					x	0 0 0 0 0 0 1 1 0 0 1 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY ∧ $\overline{\text{A.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1 0 0 1 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY ∧ X.bit	2	5	7					x	0 0 0 0 0 0 1 1 0 0 1 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY ∧ $\overline{\text{X.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1 0 0 1 1 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY ∧ PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0 0 0 1 0 0 B_2 B_1 B_0
	CY,/PSW.bit	CY ← CY ∧ $\overline{\text{PSW.bit}}$	2	5		7				x	0 0 0 0 0 0 1 0 0 0 1 1 0 B_2 B_1 B_0
OR1	CY,saddr.bit	CY ← CY V (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0 0 1 0 0 0 B_2 B_1 B_0 Saddr-offset
	CY,/saddr.bit	CY ← CY V $\overline{\text{(saddr.bit)}}$	3	5/7	9	11				x	0 0 0 0 1 0 0 0 0 1 0 1 0 B_2 B_1 B_0 Sfr-offset
	CY,sfr.bit	CY ← CY V sfr.bit)	3	7		11				x	0 0 0 0 1 0 0 0 0 1 0 1 1 B_2 B_1 B_0 Sfr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Bytes B1 thru B5**
Bit Manipulation (cont)											
OR1	CY/sfr.bit	CY ← CY V $\overline{\text{sfr.bit}}$	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 0 1 1 B_2 B_1 B_0
	CY,A.bit	CY ← CY V A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY V $\overline{\text{A.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY V X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY V $\overline{\text{X.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 1 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY V PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 0 0 0 B_2 B_1 B_0
	CY,/PSW.bit	CY ← CY V $\overline{\text{PSW.bit}}$	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 0 1 0 B_2 B_1 B_0
XOR1	CY,saddr.bit	CY ← CY⊻(saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 1 1 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,sfr.bit	CY ← CY⊻sfr.bit	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 1 0 1 B_2 B_1 B_0
											Sfr-offset
	CY,A.bit	CY ← CY⊻A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 1 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY⊻X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 1 0 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY⊻PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 1 0 0 B_2 B_1 B_0
SET1	saddr.bit	(saddr.bit) ← 1	2	3/7	6						1 0 1 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← 1	3	10		14					0 0 0 0 1 0 0 0
											1 0 0 0 1 B_2 B_1 B_0
											Sfr-offset

Instruction Set (cont)

				Clocks				Flags			
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
Bit Manipulation (cont)											
SET1	A.bit	A.bit ← 1	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 0 1 B_2 B_1 B_0
	X.bit	X.bit ← 1	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 0 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← 1	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											1 0 0 0 0 B_2 B_1 B_0
CLR1	saddr.bit	(saddr.bit) ← 0	2	6/10	6						1 0 1 0 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← 0	3	10		14					0 0 0 0 1 0 0 0
											1 0 0 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit	A.bit ← 0	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 1 1 B_2 B_1 B_0
	X.bit	X.bit ← 0	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 1 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← 0	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											1 0 0 1 0 B_2 B_1 B_0
NOT1	saddr.bit	(saddr.bit) ← 0 ($\overline{\text{saddr.bit}}$)	3	6/10	10	14					0 0 0 0 1 0 0 0
											0 1 1 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3	10		14					0 0 0 0 1 0 0 0
											0 1 1 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2	6	8						0 0 0 0 0 0 1 1
											0 1 1 1 1 B_2 B_1 B_0
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2	6	8						0 0 0 0 0 0 1 1
											0 1 1 1 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← $\overline{\text{PSW.bit}}$	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											0 1 1 1 0 B_2 B_1 B_0
SET1	CY	CY ← 1	1	2		3				1	0 1 0 0 0 0 0 1
CLR1	CY	CY ← 0	1	2		3				0	0 1 0 0 0 0 0 0
NOT1	CY	CY ← $\overline{\text{CY}}$	1	2		3				x	0 1 0 0 0 0 1 0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Call/Return											
CALL	!addr16	(SP − 1) ← $(PC+3)_H$, (SP − 2) ← $(PC+3)_L$, PC ← !addr16, SP ← SP − 2	3	10-15	17		21				0 0 1 0 1 0 0 0 Low Addr High Addr
	rp	(SP − 1) ← $(PC+2)_H$, (SP − 2) ← $(PC+2)_L$, PC_H ← rp_H, PC_L ← rp_L, SP ← SP − 2	2	12-17	15		19				0 0 0 0 0 1 0 1 0 1 0 1 1 P_2 P_1 0
CALLF	!addr11	(SP − 1) ← $(PC+2)_H$, (SP − 2) ← $(PC+2)_L$, PC_{15+11} ← 00001, PC_{10-0} ← !addr11, SP ← SP − 2	2	10-15	14		18				1 0 0 1 0 ← fa →
CALLT	[addr5]	(SP − 1) ← $(PC+1)_H$, (SP − 2) ← $(PC+1)_L$, PC_H ← (00000000, addr5 + 1), PC_L ← (00000000, addr5), SP ← SP − 2	1	14-20	20		24				1 1 1 ← ta →
BRK		(SP − 1) ← PSW, (SP − 2) ← $(PC+1)_H$, (SP − 3) ← $(PC+1)_L$, PC_H ← (003FH), PC_H ← (003FH), SP ← SP − 3, IE ← 0	1	16-26	22		28				0 1 0 1 1 1 1 0
RET		PC_L ← (SP), PC_H ← (SP + 1), SP ← SP + 2	1	10-15	11		15				0 1 0 1 0 1 1 0
RETI		PC_L ← (SP), PC_H ← (SP + 1), PSW ← (SP + 2), SP ← SP + 3, NMIS ← 0	1	12-20	15		21	R	R	R	0 1 0 1 0 1 1 1
RETB		PC_L ← (SP), PC_H ← (SP + 1), PSW ← (SP + 2), SP ← SP + 3	1	12-20	13		19	R	R	R	0 1 0 1 1 1 1 1
Stack Manipulation											
PUSH	PSW	(SP − 1) ← PSW, SP ← SP − 1	1	4-8	5		7				0 1 0 0 1 0 0 1
	sfr	(SP − 1) ← sfr, SP ← SP − 1	2	7-9	9		12				0 0 1 0 1 0 0 1 Sfr-offset
	rp	(SP − 1) ← rp_H (SP − 2) ← rp_L, SP ← SP − 2	1	8-13	8		12				0 0 1 1 1 1 P_1 P_0

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z AC CY**	**Bytes B1 thru B5**
Stack Manipulation (cont)									
POP	PSW	PSW ← (SP), SP ← SP + 1	1	4-8	6		8	R R R	0 1 0 0 1 0 0 0
	sfr	sfr ← (SP), SP ← SP + 1	2	9-11	9		12		0 1 0 0 0 0 1 1
									Sfr-offset
	rp	rp_L ← (SP), rp_H ← (SP + 1), SP ← SP +2	1	10-15	11		15		0 0 1 1 0 1 P_1 P_0
MOVW	SP,#word	SP ← word	4	8		12			0 0 0 0 1 0 1 1
									1 1 1 1 1 1 0 0
									Low Byte
									High Byte
	SP,AX	SP ← AX	2	9		11			0 0 0 1 0 0 1 1
									1 1 1 1 1 1 0 0
	AX,SP	AX ← SP	2	10		12			0 0 0 1 0 0 0 1
									1 1 1 1 1 1 0 0
INCW	SP	SP ← SP + 1	2	5		7			0 0 0 0 0 1 0 1
									1 1 0 0 1 0 0 0
DECW	SP	SP ← SP − 1	2	5		7			0 0 0 0 0 1 0 1
									1 1 0 0 1 0 0 1

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z AC CY	Bytes B1 thru B5
Unconditional Branch								
BR	!addr16	PC ← !addr16	3	5	11			0 0 1 0 1 1 0 0
								Low Addr
	rp	PC_H ← rp_H, PC_L ← rp_L	2	6	10			0 0 0 0 0 1 0 1
								0 1 0 0 1 P_2 P_1 0
	$addr16	PC ← $addr16	2	4	9			0 0 0 1 0 1 0 0
								jdisp
Conditional Branch								
BC	$addr16	PC ← $addr16 if CY = 1	2	2/4	9	6		1 0 0 0 0 0 1 1
BL								jdisp
BNC	$addr16	PC ← $addr16 if CY = 0	2	2/4	9	6		1 0 0 0 0 0 1 0
BNL								jdisp
BZ	$addr16	PC ← $addr16 if Z = 1	2	2/4	9	6		1 0 0 0 0 0 0 1
BE								jdisp
BNZ	$addr16	PC ← $addr16 if Z = 0	2	2/4	9	6		1 0 0 0 0 0 0 0
BNE								jdisp
BT	saddr.bit, $addr16	PC ← $addr16 if (saddr.bit) = 1	3	5-9	12	9		0 1 1 1 0 B_2 B_1 B_0
								Saddr-offset
								jdisp
	sfr.bit, $addr16	PC ← $addr16 if sfr.bit = 1	4	7/9	16	13		0 0 0 0 1 0 0 0
								1 0 1 1 1 B_2 B_1 B_0
								Sfr-offset
								jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 1
								1 0 1 1 1 B_2 B_1 B_0
								jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 1
								1 0 1 1 0 B_2 B_1 B_0
								jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 0
								1 0 1 1 0 B_2 B_1 B_0
								jdisp

Instruction Set (cont)

				Clocks			Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z	AC	CY	Bytes B1 thru B5
Conditional Branch (cont)										
BF	saddr.bit,$addr16	PC ← $addr16 if (saddr.bit) = 0	4	5-9	15	12				0 0 0 0 1 0 0 0
										1 0 1 0 1 B_2 B_1 B_0
										Saddr-offset
										jdisp
	sfr.bit,$addr16	PC ← $addr16 if sfr.bit = 0	4	7/9	16	13				0 0 0 0 1 0 0 0
										1 0 1 0 1 B_2 B_1 B_0
										Sfr-offset
										jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 0	3	5/7	12	9				0 0 0 0 0 0 1 1
										1 0 1 0 1 B_2 B_1 B_0
										jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 0	3	5/7	12	9				0 0 0 0 0 0 1 1
										1 0 1 0 0 B_2 B_1 B_0
										jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 0	3	5/7	12	9				0 0 0 0 0 0 1 0
										1 0 1 0 0 B_2 B_1 B_0
										jdisp
BTCLR	saddr.bit,$addr16	PC ← $addr16 if (saddr.bit) = 1 then reset (saddr.bit)	4	5-13	15	12				0 0 0 0 1 0 0 0
										1 1 0 1 1 B_2 B_1 B_0
										Saddr-offset
										jdisp
	sfr.bit,$addr16	PC ← $addr16 if sfr.bit = 1 then reset sfr.bit	4	7/13	18	13				0 0 0 0 1 0 0 0
										1 1 0 1 1 B_2 B_1 B_0
										Sfr-offset
										jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 1 then reset A.bit	3	5/9	12	9				0 0 0 0 0 0 1 1
										1 1 0 1 1 B_2 B_1 B_0
										jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 1 then reset X.bit	3	5/9	12	9				0 0 0 0 0 0 1 1
										1 1 0 1 0 B_2 B_1 B_0
										jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 1 then reset PSW.bit	3	5/8	12	9	x	x	x	0 0 0 0 0 0 1 0
										1 1 0 1 0 B_2 B_1 B_0
										jdisp

6

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**Int ROM**	**Branch**	**No Branch**	**Z AC CY**	**Bytes B1 thru B5**
Conditional Branch (cont)								
DBNZ	rl,$addr16	rl ← rl – 1, then PC ← $addr16 if rl ≠ 0	2	3/5	9	6		0 0 1 1 0 0 1 R_0 jdisp
	saddr,$addr16	(saddr) ← (saddr) – 1, then PC ← $addr16 if (saddr) ≠ 0	3	4-10	12	9		0 0 1 1 1 0 1 1 Saddr-offset jdisp
CPU Control								
MOV	STBC,#byte	STBC ← byte	4	10	15			0 0 0 0 1 0 0 1 1 1 0 0 0 0 0 0 Data Data
SEL	RBn	RBS1–0 ← n, n = 0–3	2	2	6			0 0 0 0 0 1 0 1 1 0 1 0 1 0 N_1 N_0
NOP		No Operation	1	2	3			0 0 0 0 0 0 0 0
EI		IE ← 1 (Enable Interrupt)	1	2	3			0 1 0 0 1 0 1 1
DI		IE ← 0 (Disable Interrupt)	1	2	3			0 1 0 0 1 0 1 0

NEC
NEC Electronics Inc.

μPD7823x
Advanced, 8-Bit Real-Time Control Microcomputers With A/D and D/A Converters

Description

The μPD78233, μPD78234, and μPD78P238 are high-performance, 8-bit, single-chip microcomputers. They contain extended addressing capabilities for up to 1M byte of external memory. The devices also integrate sophisticated analog and digital peripherals as well as two low-power standby modes that make them ideal for low-power/battery backup applications.

The μPD7823x family focuses on embedded control with features like hardware multiply and divide, two levels of interrupt response, four banks of main registers for multitasking, and macroservice for processor-independent peripheral and memory DMA. Augmenting this high-performance core are advanced components like high-precision A/D and D/A converters, two independent serial interfaces, several counter/timers, PWM outputs as well as a real-time output port. On board memory includes up to 1K bytes of RAM and 32K bytes of mask ROM or OTP ROM.

The macroservice routine allows data to be transferred between any combination of memory and peripherals independent of the current program execution. The four banks of processor registers allow simplified context switching to be performed. Both features combined with powerful on-chip peripherals make this part ideal for a wide variety of embedded control applications.

Features

□ Complete single-chip microcomputer
- 8-bit ALU
- 16K ROM
- 640 bytes RAM
- Both 1-bit and 8-bit logic

□ Instruction prefetch queue

□ Hardware multiply and divide

□ Memory expansion
- 8085 bus-compatible
- 64K program address space
- 1M data address space

□ Large I/O capacity: up to 64 I/O port lines

□ Two 12-bit PWM outputs

□ Eight-input 8-bit A/D converters

□ Two-output 8-bit D/A converters

□ Extensive timer/counter functions
- One 16-bit timer/counter/event counter
- Three 8-bit timer/counter/event counter

□ Four timer-controlled PWM channels

□ Two 4-bit real-time output ports

□ Extensive interrupt handler
- Vectored interrupt handling
- Programmable priority
- Macroservice mode

□ Two independent serial ports

□ Software pullup options

□ Refresh output for pseudostatic RAM

□ On-chip clock generator
- 12-MHz maximum CPU clock frequency
- 0.33-μs instruction cycle

□ CMOS silicon gate technology

□ 5-volt power supply

Ordering Information

Part Number	ROM	Package
μPD78233GC-3B9	ROMless	80-pin plastic QFP
μPD78233L		84-pin PLCC
μPD78233GJ-5B6		94-pin plastic QFP
μPD78234GC-3B9	16K Mask ROM	80-pin plastic QFP
μPD78234L		84-pin PLCC
μPD78234GJ-5B6		94-pin plastic QFP
μPD78P238GC-3B9	32K OTP ROM	80-pin plastic QFP
μPD78P238L		84-pin PLCC
μPD78P238GJ-5B6		94-pin plastic QFP

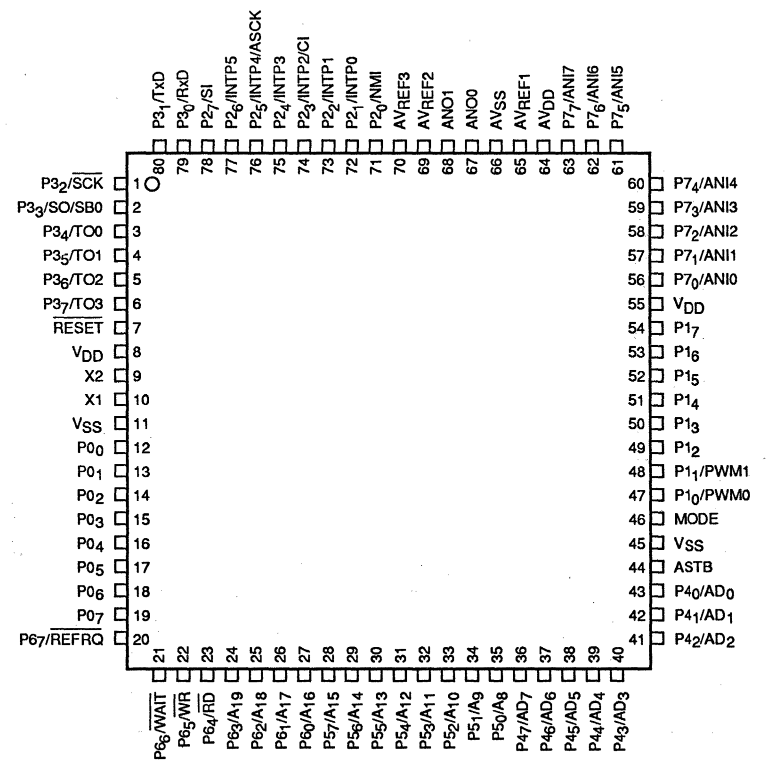
Pin Identification

Symbol	Function
$P0_0$-$P0_7$	Output port 0
$P1_0$-$P1_1$/PWM0-PWM1	I/O port 1/Pulse-width modulated outputs
$P1_2$-$P1_7$	I/O port 1
$P2_0$/NMI	Input port 2/Non-maskable interrupt input
$P2_1$-$P2_2$/INTP0-INTP1	Input port 2/External interrupt input/timer trigger
$P2_3$/INTP2/CI	Input port 2/External interrupt input/ Clock input
$P2_4$/INTP3	Input port 2/External interrupt input/timer trigger
$P2_5$/INTP4/ASCK	Input port 2/External interrupt input/ Asynchronous serial clock
$P2_6$/INTP5	Input port 2/External interrupt input
$P2_7$/SI	Input port 2/Serial input
$P3_0$/RxD	I/O port 3/Serial receive input
$P3_1$/TxD	I/O port 3/Serial transmit output
$P3_2/\overline{SCK}$	I/O port 3/Serial clock input/output
$P3_3$/SO/SB0	I/O port 3/Serial output/Serial bus I/O
$P3_4$-$P3_7$/TO0-TO3	I/O port 3/Timer output
$P4_0$-$P4_7$/AD_0-AD_7	I/O port 4/Lower address byte/data bus

Symbol	Function
$P5_0$-$P5_7$/A_8-A_{15}	I/O port 5/Upper address byte
$P6_0$-$P6_3$/A_{16}-A_{19}	Output port 6/Extended address nibble
$P6_4/\overline{RD}$	I/O port 6/Read strobe output
$P6_5/\overline{WR}$	I/O port 6/Write strobe output
$P6_6/\overline{WAIT}$	I/O port 6/Wait input
$P6_7/\overline{REFRQ}$	I/O port 6/Refresh output
$P7_0$-$P7_7$/ANI0-ANI7	Input port 7/A/D converter inputs
ANO0-ANO1	D/A converter output
ASTB	Address strobe output
$\overline{RESET}$	External reset input
$\overline{MODE}$	External memory access control input
X1, X2	External crystal or external clock input
AV_{REF1}	A/D converter reference voltage
AV_{REF2}, AV_{REF3}	D/A converter reference voltages
AV_{ss}	Analog ground
V_{DD}	Positive power supply input
AV_{DD}	Positive power supply input; analog section
V_{SS}	Power return; normally ground
NC	No connection

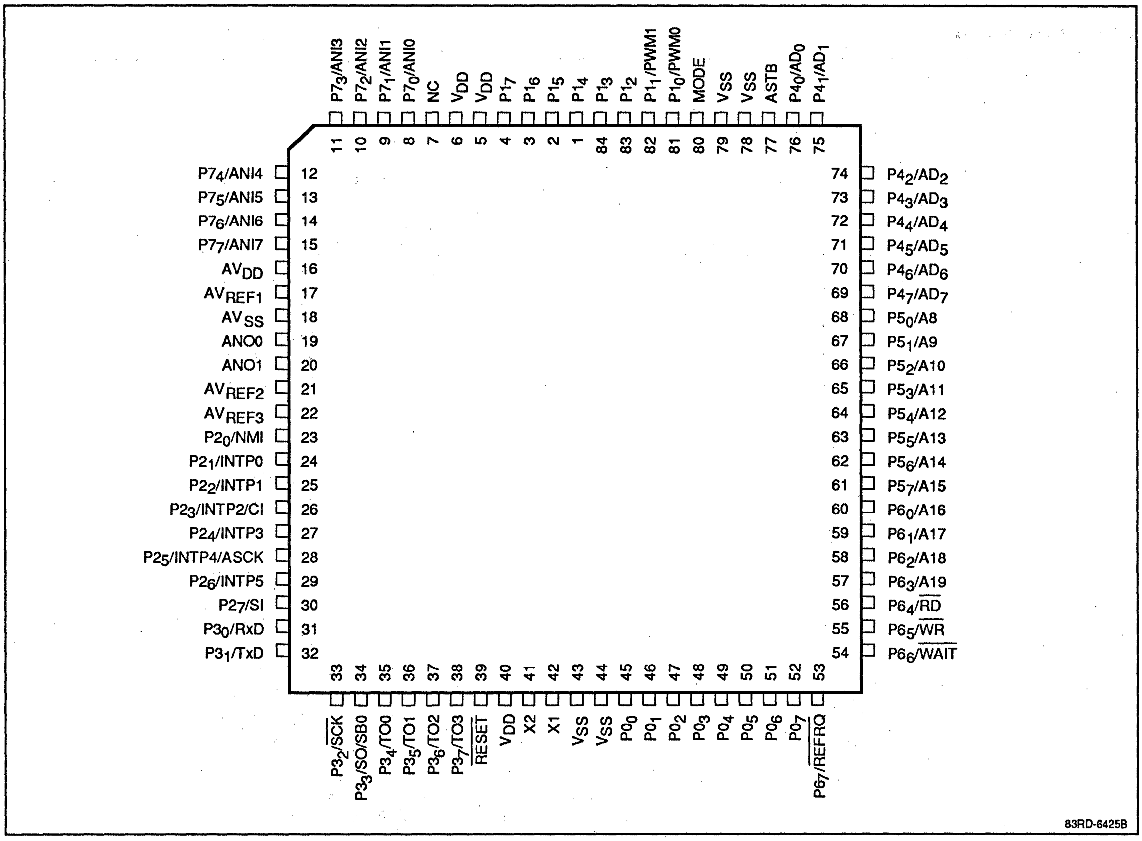
Pin Configurations

80-Pin Plastic QFP

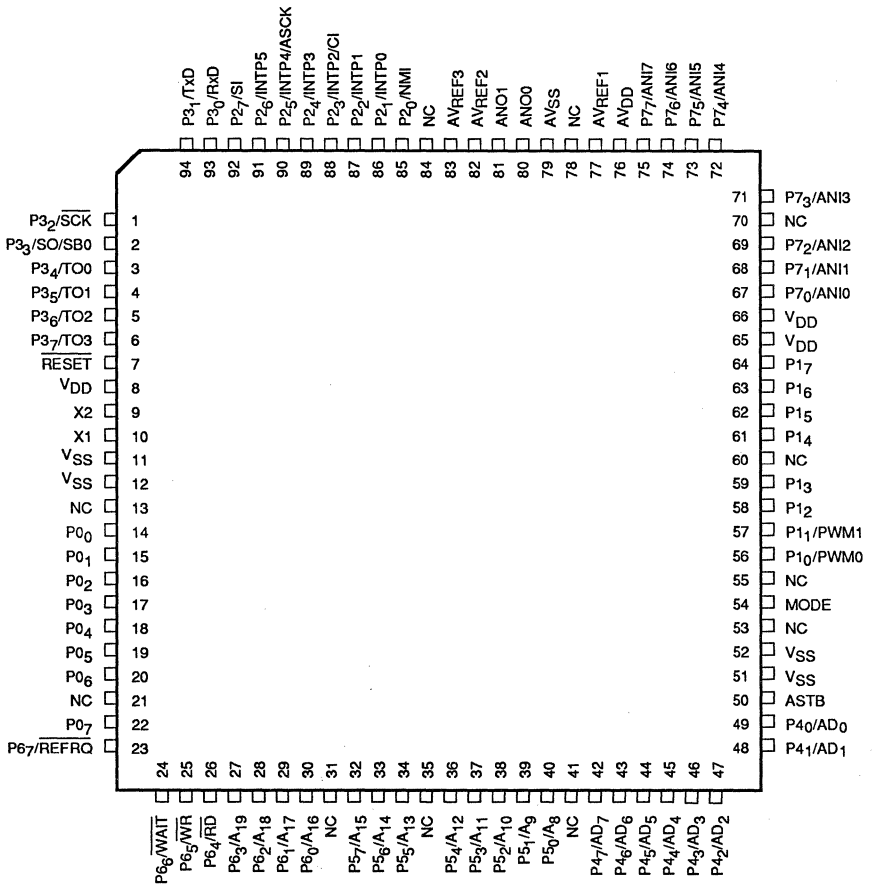


83RD-6424B

84-Pin PLCC (Plastic Leaded Chip Carrier)



94-Pin Plastic QFP



83RD-6426B

Pin Functions

P0$_0$-P0$_7$. Port 0 is an 8-bit, tristate output port with direct transistor drive capability. Port 0 can also be configured as two 4-bit, real-time (timer-controlled) output ports.

P1$_0$-P1$_7$. Port 1 is an 8-bit input/output port with the programmable pullup option. Port 1 has direct LED drive capability.

PWM0-PWM1. These are pulse-width modulated outputs for dc motor control.

P2$_0$-P2$_7$. Port 2 is an 8-bit input port with the programmable pullup option except for P2$_0$ and P2$_1$.

NMI. Non-maskable interrupt input.

INTP0-INTP5. External interrupt inputs. INTP0, INTP1, and INTP3 are timer capture trigger inputs.

CI. External clock input to the timer.

ASCK. Asynchronous serial clock input.

SI. Serial data input for three-wire serial I/O mode.

P3$_0$-P3$_7$. Port 3 is an 8-bit tristate I/O port with the programmable pullup option.

RxD. Receive serial data input.

TxD. Transmit serial data output.

$\overline{\text{SCK}}$. Serial shift clock output.

SO. Serial data output for three-wire serial I/O mode.

SB0. I/O bus for the clocked serial interface.

TO0-TO3. Timer flip-flop outputs.

P4$_0$-P4$_7$. Port 4 is an 8-bit, bidirectional tristate port with the programmable pullup option. Port 4 has direct LED drive capability.

AD$_0$-AD$_7$. Multiplexed address/data bus used with external memory or expanded I/O.

P5$_0$-P5$_7$. Port 5 is an 8-bit, bidirectional tristate port with the programmable pullup option. Port 5 has direct LED drive capability.

A$_8$-A$_{15}$. Upper-order address bus used with external memory or expanded I/O.

P6$_0$-P6$_3$. Pins P6$_0$-P6$_3$ of port 6 are outputs.

A$_{16}$-A$_{19}$. Extended-order address bus used with external memory.

P6$_4$-P6$_7$. Pins P6$_4$-P6$_7$ of port 6 are tristate I/Os with the programmable pullup option.

$\overline{\text{RD}}$. Read strobe output used by external memory (or data registers) to place data on the I/O bus during a read operation.

$\overline{\text{WR}}$. Write strobe output used by external memory (or data registers) to latch data from the I/O bus during a write operation.

$\overline{\text{WAIT}}$. Wait signal input.

$\overline{\text{REFRQ}}$. Refresh pulse output used by external pseudo-static memory.

P7$_0$-P7$_7$. Port 7 is an 8-bit input port.

ANI0-ANI7. Analog voltage inputs to A/D converter.

ANO1, ANO2. Analog voltage outputs from D/A converters.

ASTB. Address strobe output used by external circuitry to latch the low-order 8 address bits during the first part of a read or write cycle.

$\overline{\text{RESET}}$. A low level on this external reset input sets all registers to their specified reset values. This pin, together with P2$_0$/NMI, sets the μPD78P234 in the PROM programming mode.

MODE. Control signal input that selects external memory or internal ROM as the program memory. When MODE is low, μPD78234 is set in ROMless mode and external memory is accessed.

X1, X2. For frequency control of the internal clock oscillator, a crystal is connected to X1 and X2. If the clock is supplied by an external source, the clock signal is connected to X1 and the inverted clock signal is connected to X2.

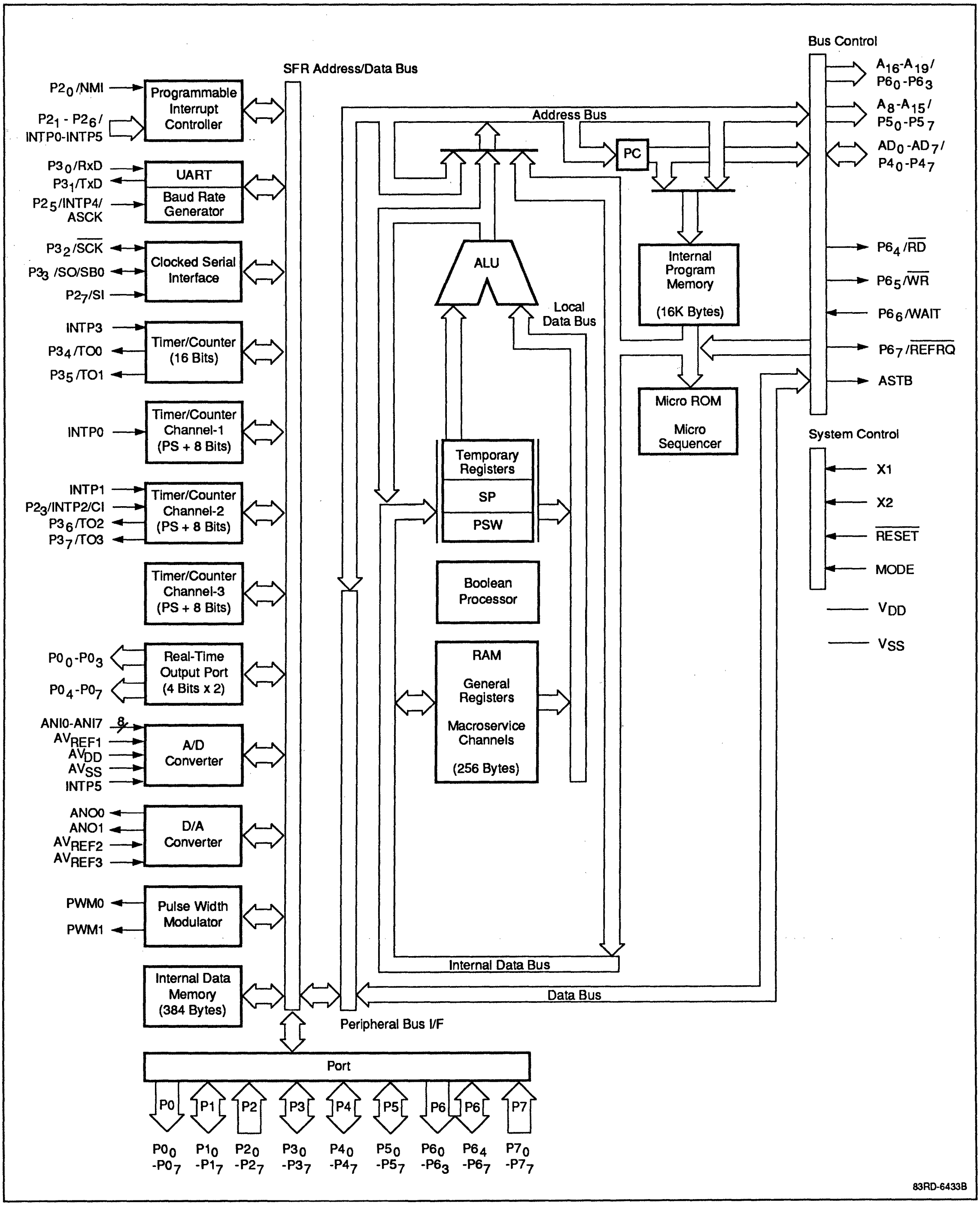
AV$_{REF1}$. A/D converter reference voltage.

AV$_{REF2}$, AV$_{REF3}$. D/A converter reference voltage.

AV$_{DD}$. A/D converter supply voltage.

AV$_{SS}$. A/D converter ground.

μPD7823x Block Diagram



FUNCTIONAL DESCRIPTION

Timing

The maximum clock frequency is 12 MHz. The clock is derived from an external crystal or an external oscillator. The internal processor clock is two-phase and the machine states are executed at a rate of 6 MHz. The shortest instructions require two states (333 ns). The CPU contains a one-byte instruction prefetch. This allows a subsequent instruction to be fetched during the execution of an instruction that does not reference memory.

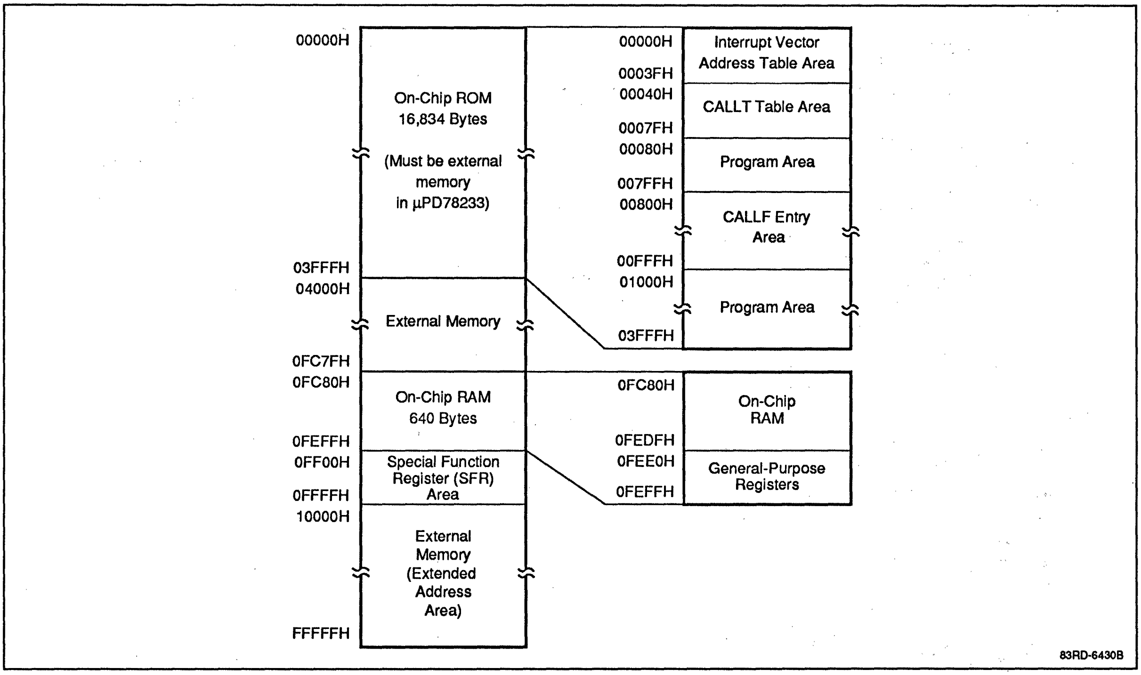
Memory Map

The μPD7823x has 1M byte of address space. This address space is partitioned into 64K bytes of program memory starting at address 00000H. (See figure 1.) The remainder of the 1M bytes can be accessed as data memory space.

External memory is supported by I/O port 4, an 8-bit multiplexed address/data bus. The memory mapping register controls the size of external memory as well as the number of added wait states. The upper address byte is derived from port 5, and the extended address nibble is derived from port 6.

The μPD78234 has on-chip mask ROM occupying the space from 00000H to 03FFFH. When the ROM is used and no other program or data space is required, ports 4, 5, and 6 are available as additional I/O ports.

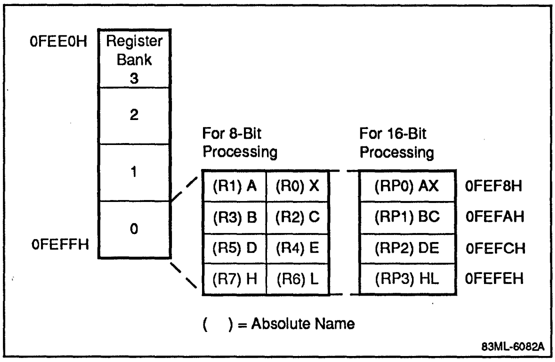
Figure 1. Memory Map



General-Purpose Registers

The general-purpose registers are mapped into specific addresses in data memory. They are made up of four banks, each bank consisting of eight 8-bit or four 16-bit registers. The register bank used is specified by a CPU instruction. This can be checked by reading RBS0 and RBS1 in the program status word (PSW).The general-purpose register configuration is shown in figure 2.

Figure 2. Register Mapping



Special Registers

There are three different special registers. The first is a 16-bit binary counter that holds the next program address to be executed and is named the program counter. The stack pointer is the second special 16-bit register. The stack pointer holds the address of the stack area (a last in, first out system). The third special register is an 8-bit program status word. This register contains various flags that are set or reset depending on the results of instruction execution. The program status word format is as follows:

7							0
IE	Z	RBS1	AC	RBS0	0	ISP	CY

CY	Carry flag
ISP	Interrupt priority status flag
RBS0, RBS1	Register bank selection flags
AC	Auxiliary carry flag
Z	Zero flag
IE	Interrupt request enable flag

Special Function Registers

These registers are assigned to special functions such as the mode and control registers for on-chip peripheral hardware. They are mapped into the 256-byte memory space from 0FF00H to 0FFFFH. Table 1 is a list of special function registers.

Table 1. Special Function Registers

				Handleable Bit Unit			
Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bit	16 Bit	On Reset
0FF00H	Port 0	P0	R/W	o	o	–	Indeterminate
0FF01H	Port 1	P1	R/W	o	o	–	Indeterminate
0FF02H	Port 2	P2	R	o	o	–	Indeterminate
0FF03H	Port 3	P3	R/W	o	o	–	Indeterminate
0FF04H	Port 4	P4	R/W	o	o	–	Indeterminate
0FF05H	Port 5	P5	R/W	o	o	–	Indeterminate
0FF06H	Port 6	P6	R/W	o	o	–	x0H
0FF07H	Port 7	P7	R	o	o	–	Indeterminate
0FF0AH	Port 0 buffer register (low)	P0L	R/W	o	o	–	Indeterminate
0FF0BH	Port 0 buffer register (high)	P0H	R/W	o	o	–	Indeterminate
0FF0CH	Real-time output port control register	RTPC	R/W	o	o	–	00H
0FF10H, 0FF11H	16-bit compare register 0 (16-bit timer/counter)	CR00	R/W	–	–	o	Indeterminate
0FF12H, 0FF13H	16-bit compare register 1 (16-bit timer/counter)	CR01	R/W	–	–	o	Indeterminate
0FF14H	8-bit compare register (8-bit timer/counter 1)	CR10	R/W	–	o	–	Indeterminate
0FF15H	8-bit compare register (8-bit timer/counter 2)	CR20	R/W	–	o	–	Indeterminate
0FF16H	8-bit compare register (8-bit timer/counter 2)	CR21	R/W	–	o	–	Indeterminate
0FF17H	8-bit compare register (8-bit timer/counter 3)	CR30	R/W	–	o	–	Indeterminate
0FF18H, 0FF19H	16-bit capture register (16-bit timer/counter)	CR02	R	–	–	o	Indeterminate
0FF1AH	8-bit capture register (8-bit timer/counter 2)	CR22	R	–	o	–	Indeterminate
0FF1CH	8-bit capture/compare register (8-bit timer/counter 1)	CR11	R/W	–	o	–	Indeterminate
0FF20H	Port 0 mode register	PM0	W	–	o	–	FFH
0FF21H	Port 1 mode register	PM1	R	–	o	–	FFH
0FF23H	Port 3 mode register	PM3	W	–	o	–	FFH
0FF25H	Port 5 mode register	PM5	W	–	o	–	FFH
0FF26H	Port 6 mode register	PM6	R/W	–	o	–	FxH
0FF30H	Capture/compare control register 0	CRC0	W	–	o	–	10H
0FF31H	Timer output control register	TOC	W	–	o	–	00H
0FF32H	Capture/compare control register 1	CRC1	W	–	o	–	00H
0FF34H	Capture/compare control register 2	CRC2	W	–	o	–	00H
0FF40H	Pull-up option register	PUO	R/W	o	o	–	00H
0FF43H	Port 3 mode control register	PMC3	R/W	o	o	–	00H
0FF50H, 0FF51H	16-bit timer register 0	TM0	R	–	–	o	0000H
0FF52H	8-bit timer register 1	TM1	R	–	o	–	00H

Table 1. Special Function Registers (cont)

					Handleable Bit Unit		
Address	Special Function Register (SFR) Name	Symbol	R/W	1 Bit	8 Bit	16 Bit	On Reset
0FF54H	8-bit timer register 2	TM2	R	–	o	–	00H
0FF56H	8-bit timer register 3	TM3	R	–	o	–	00H
0FF5CH	Prescaler mode register 0	PRM0	W	–	o	–	00H
0FF5DH	Timer control register 0	TMC0	R/W	–	o	–	00H
0FF5EH	Prescaler mode register 1	PRM1	W	–	o	–	00H
0FF5FH	Timer control register 1	TMC1	R/W	–	o	–	00H
0FF60H	D/A converter value setting register 0	DACS0	R/W	–	o	–	00H
0FF61H	D/A converter value setting register 1	DACS1	R/W	–	o	–	00H
0FF68H	A/D converter mode register	ADM	R/W	o	o	–	00H
0FF6AH	A/D conversion result register	ADCR	R	–	o	–	Indeterminate
0FF70	PWM control register	PWMC	R/W	–	o	–	05H
0FF72H, 0FF73H	PWM modulo register 0	PWM0	W	–	–	o	Indeterminate
0FF74H, 0FF75H	PWM modulo register 1	PWM1	W	–	–	o	Indeterminate
0FF7DH	One-shot pulse output control register	OSPC	R/W	o	o	–	00H
0FF80H	Clocked serial interface mode register	CSIM	R/W	o	o	–	00H
0FF82H	Serial bus interface control register	SBIC	R/W	o	o	–	00H
0FF86H	Serial shift register	SIO	R/W	–	o	–	Indeterminate
0FF88H	Asynchronous serial interface mode register	ASIM	R/W	o	o	–	80H
0FF8AH	Asynchronous serial interface status register	ASIS	R	o	o	–	00H
0FF8CH	Serial receive buffer: UART	RxB	R	–	o	–	Indeterminate
0FF8EH	Serial send shift register: UART	TxS	W	–	o	–	Indeterminate
0FF90H	Baud rate generator control register	BRGC	W	–	o	–	00H
0FFC0H	Standby control register	STBC	R/W	–	o	–	0000x000B
0FFC4H	Memory expansion mode register	MM	R/W	o	o	–	20H
0FFC5H	Programmable wait control register	PW	R/W	o	o	–	80H
0FFC6H	Refresh mode register	RFM	R/W	o	o	–	00H
0FFCFH	Memory size control register	IMS	W	–	o	–	Indeterminate
0FFE0H	Interrupt request flag register L	IF0L IF0	R/W	o	o	o	0000H
0FFE1H	Interrupt request flag register H	IF0H	R/W	o	o		0000H
0FFE4H	Interrupt mask flag register L	MK0L MK0	R/W	o	o	o	FFFFH
0FFE5H	Interrupt mask flag register H	MK0H	R/W	o	o		FFFFH
0FFE8H	Priority specification flag register L	PR0L PR0	R/W	o	o	o	FFFFH
0FFE9H	Priority specification flag register H	PR0H	R/W	o	o		FFFFH
0FFECH	Interrupt service mode specification flag register L	ISM0L ISM0	R/W	o	o	o	0000H
0FFECH	Interrupt service mode specification flag register L	ISM0L ISM0	R/W	o	o	o	0000H
0FFEDH	Interrupt service mode specification flag register H	ISM0H	R/W	o	o		0000H
0FFF4H	External interrupt mode register 0	INTM0	R/W	o	o	–	00H
0FFF5H	External interrupt mode register 1	INTM1	R/W	o	o	–	00H
0FFF8H	Interrupt status register	IST	R/W	o	o	–	00H

Input/Output Ports

Port 0 is a byte programmable tristate output port. Port 1 is bit programmable as input or output pins. Port 2 is bit selectable as input or control pins. Port 3 is bit programmable as input, output, or control pins. Port 4 is byte programmable as an I/O port or as the external address/data bus. Port 5 is bit programmable as I/O or the upper address byte. Port 6 is bit programmable as I/O, control pins, or the extended address nibble. Port 7 is an input only port.

Real-Time Output Port

The real-time output port (figure 3) shares pins with port 0. The high and low nibbles may be treated separately or together. In the real-time output function, data stored beforehand in the buffer register is transferred to the output latch simultaneously with the generation of either a timer interrupt or external interrupt. Using the real-time output function in conjunction with the macroservice function enables port 0 to output preprogrammed patterns at preprogrammed variable time intervals.

A/D Converter

The μPD7823x A/D converter (figure 4) uses the successive-approximation method of converting any or all of the eight multiplexed analog inputs into 8-bit digital data. This data is stored in a result register that can be accessed at any time. The conversion time is 30 μs at 12-MHz operation. Quantization error is ±1/2 LSB; maximum full-scale error is 0.4%.

There are two methods for starting the A/D conversion operation. Conversion may be started by hardware by using an external interrupt as a trigger. The second method of starting conversion is with a software command.

There are also two methods by which the μPD7823x will operate after conversion has begun. The first, the scan method, selects several analog input signals sequentially and obtains data from each pin producing an interrupt with each conversion. The converted data can be successively transferred to memory by using the macroservice function. The second, the select mode, chooses any one input and the result is updated continuously, with or without interrupt generation depending on the chosen start method.

Figure 3. Real-Time Output Port

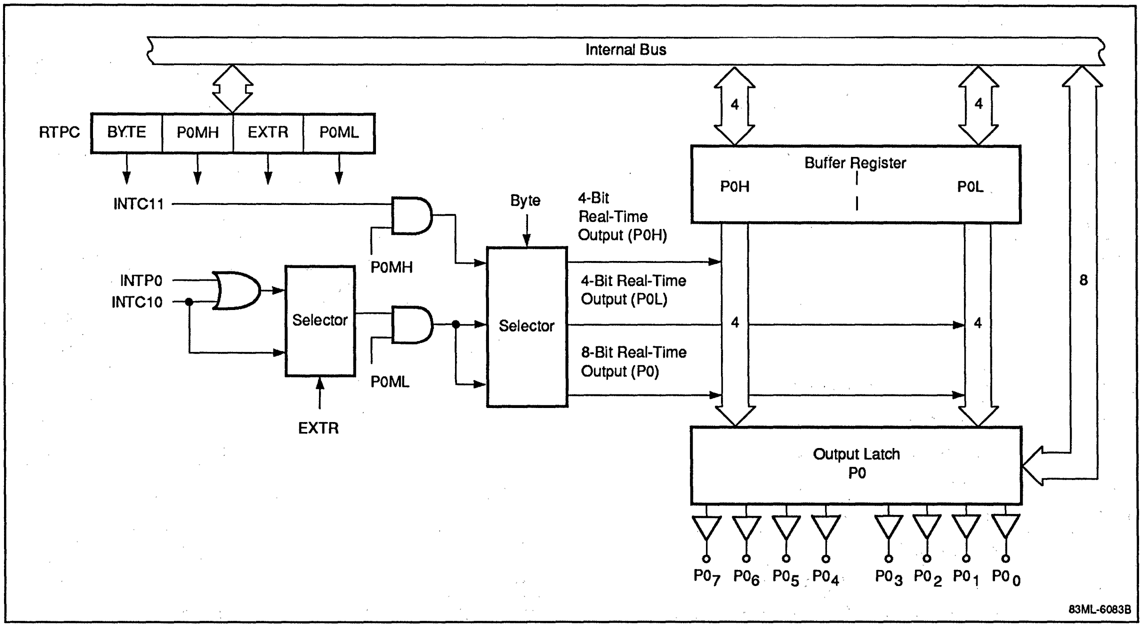
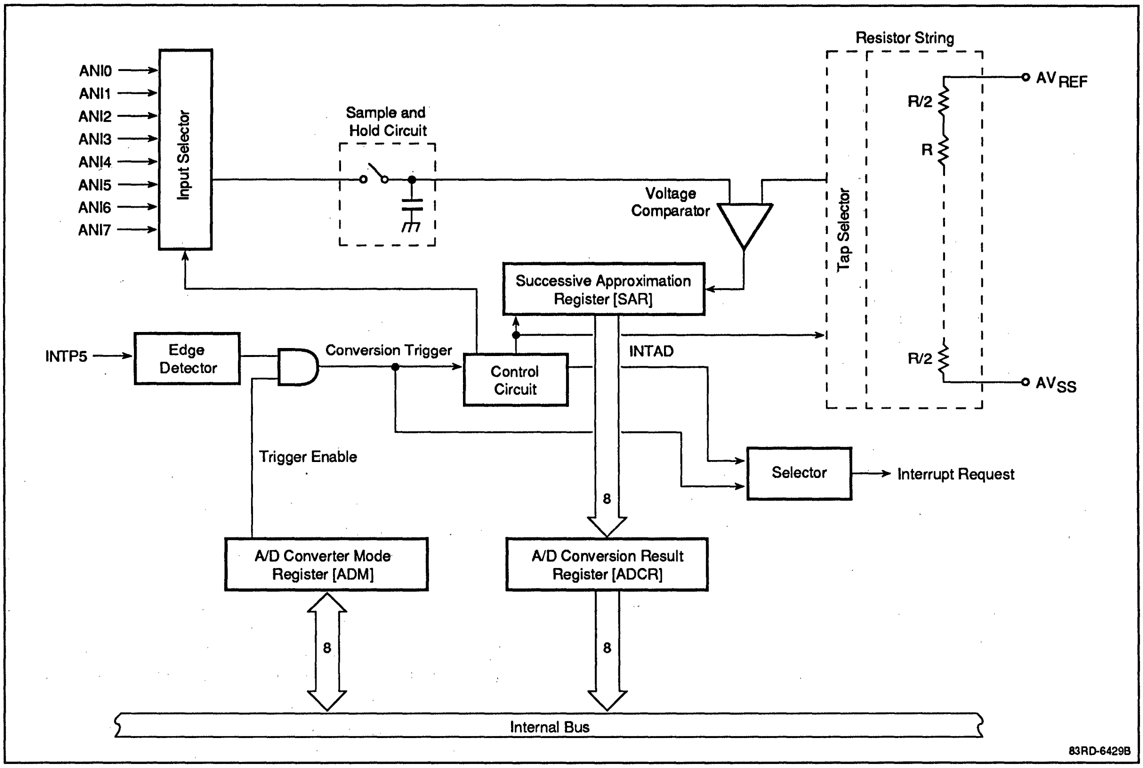


Figure 4. Analog-to-Digital Converter



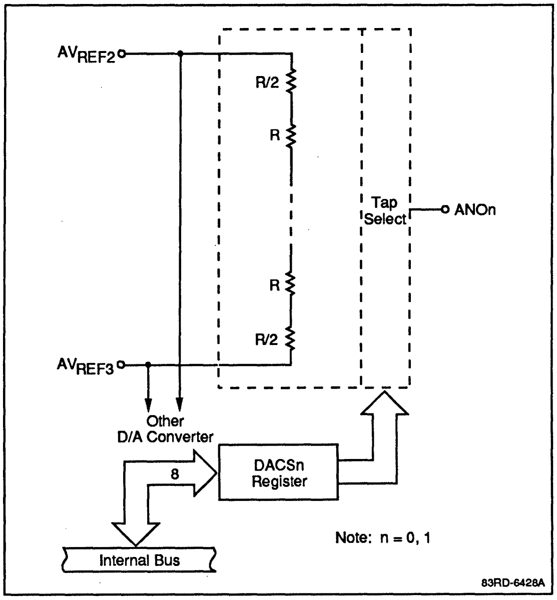
D/A Converter

The μPD7823x has two D/A converters as shown in figure 5. The 8-bit digital input, written to the DACSn register (n = 0, 1), selects one of 256 taps on a resistor ladder between reference voltages AV_{REF2} and AV_{REF3}. The selected voltage becomes the analog output at the ANOn pin.

Because of the high impedance at ANOn, an external buffer is required to drive a low-impedance load.

The ANOn pin is high impedance also while the RESET signal is active. After reset clears, the DACSn register is loaded with 0s.

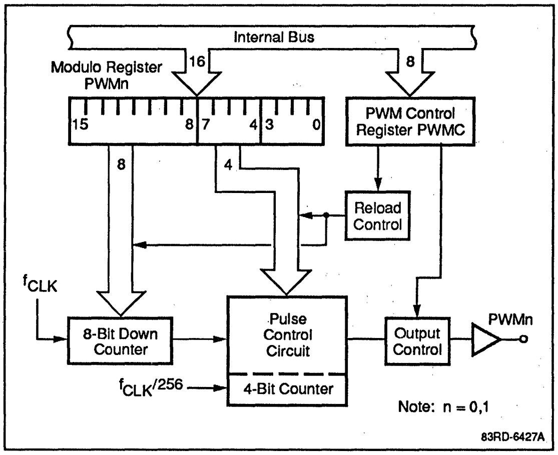
Figure 5. Digital-to-Analog Converter



PWM Output

The two pulse-width modulators of the μPD7823x (figure 6) have 12-bit resolution. Designed for dc motor speed control, the outputs at PWMn (n = 0, 1) are selectable independently as active low or high.

Figure 6. Pulse-Width Modulator



Serial Interface

The μPD7823x has two independent serial interfaces.

- Asynchronous serial interface (UART) (figure 7)
- Clock-synchronized serial interface (figure 8)

A universal asynchronous receiver transmitter (UART) is used as an asynchronous serial interface. This interface transfers one byte of data following a start bit. The μPD7823x contains a baud rate generator. This allows data to be transferred over a wide range of transfer rates. Transfer rates may also be defined by dividing the clock input to the ASCK pin. Transfer rates may also be generated by 8-bit timer counter 3.

The clock-synchronized serial interface has two different modes of operation:

- Three-line serial I/O mode.
  In this mode, data 8 bits long is transferred along three lines: a serial clock (SCK) line and two serial bus lines (SO and SI). This mode is convenient when the μPD7823x is connected to peripheral I/Os and display controllers that have the conventional clock-synchronized serial interface.
- Serial bus interface mode (SBI).
  In this mode the μPD7823x can communicate data with several devices using the serial clock (SCK) and the serial data bus (SBO) lines. This mode conforms to NEC's serial bus format. In SBI mode, addresses that select a device to communicate with, commands that direct the device, and actual data are output to the serial data bus. A handshake line, which was required for connecting several devices in the conventional clock-synchronized serial interface, is not needed.

Figure 7. Asynchronous Serial Interface

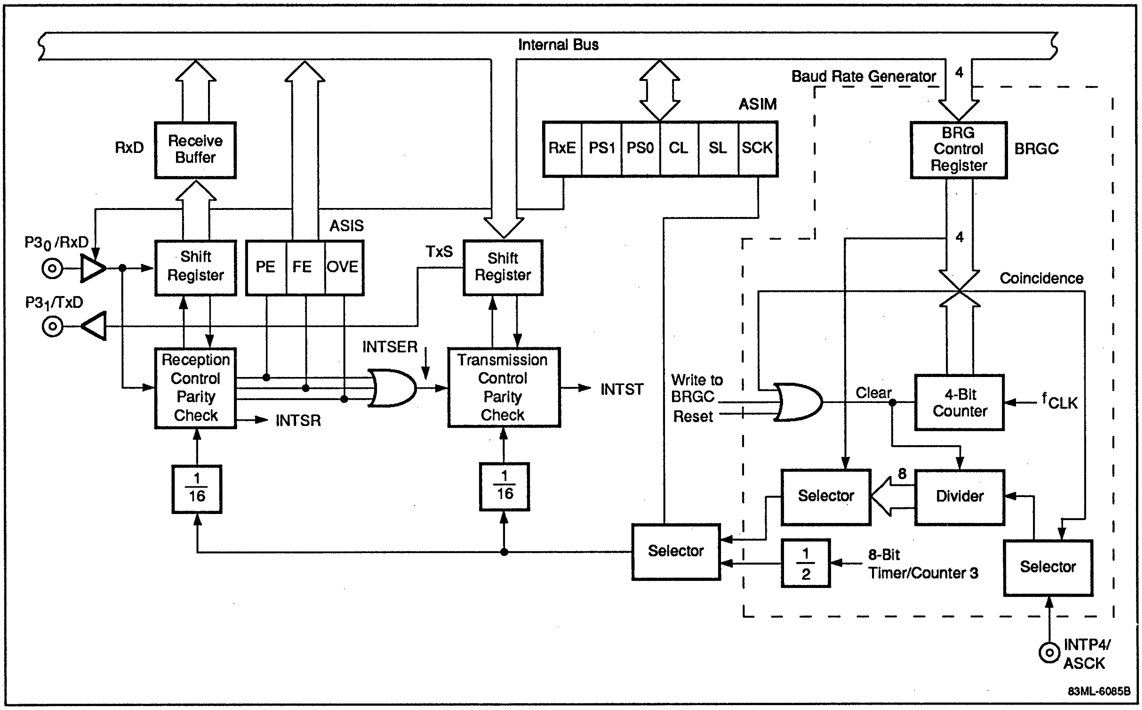
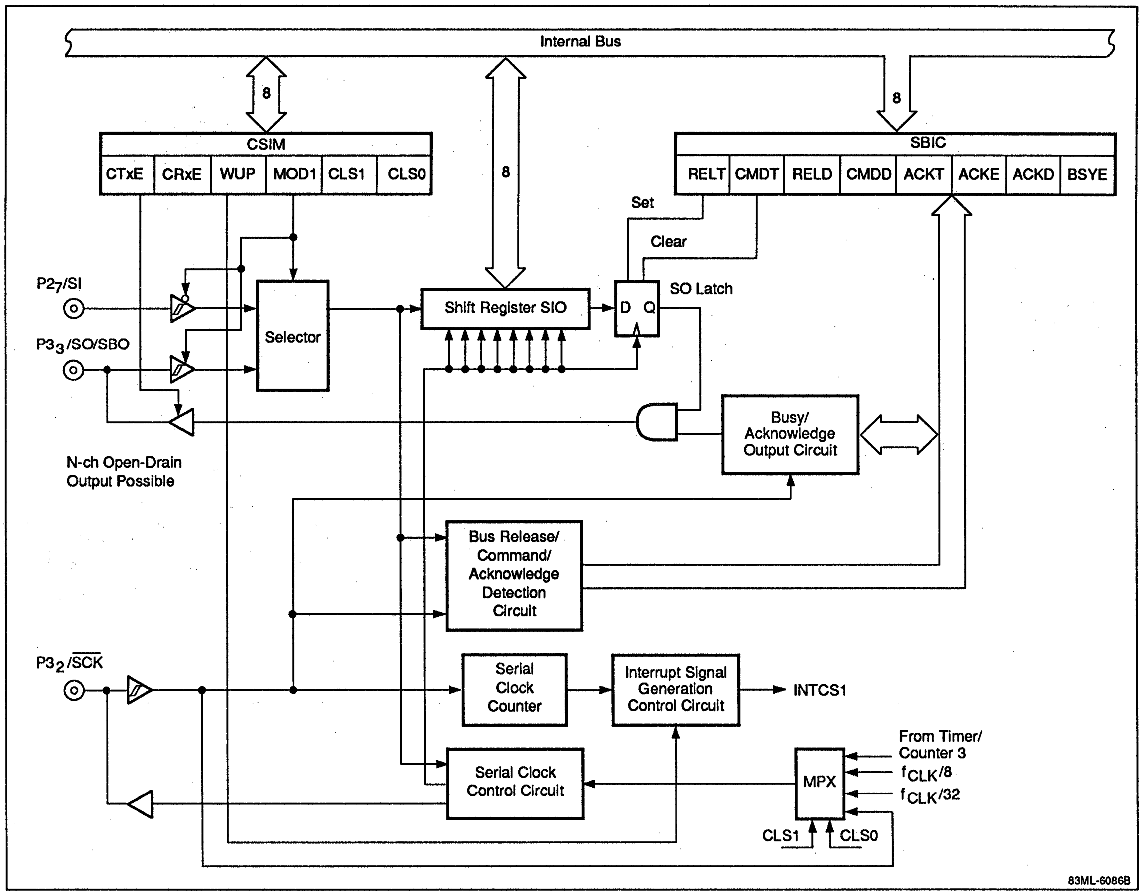


Figure 8. Clock-Synchronized Serial Interface



Timer/Counters

The μPD7823x has four timer/counters: one 16-bit and three 8-bit. The 16-bit timer/counter (figure 9) has the basic functionality of an interval timer, a programmable square-wave output, and a pulse width measurer. These functions can provide a digital delayed one-shot output, a pulse width modulated output, and a cycle measurer.

The first two 8-bit timer/counters can provide the basic functions of an interval timer and a pulse width measurer. Timer/counter 1 can also be used as a timer for output trigger generation for the real-time output port. Timer/counter 2 can also provide an external event counter, a one-shot timer, a programmable square-wave output, a pulse-width modulated output, and a cycle measurer. Timer/counter 3 can operate as an internal timer or as a counter to generate clocks for a baud rate generator. See figures 10, 11, and 12.

Figure 9. 16-Bit Timer/Counter

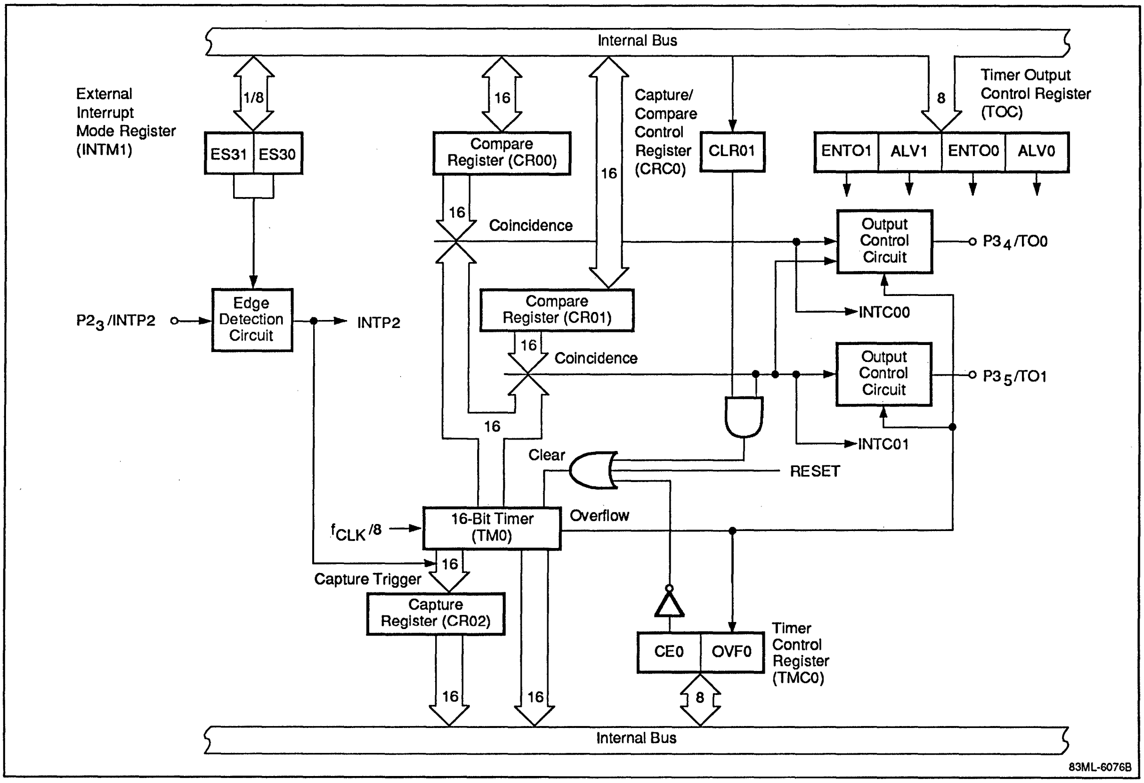


Figure 10. 8-Bit Timer/Counter 1

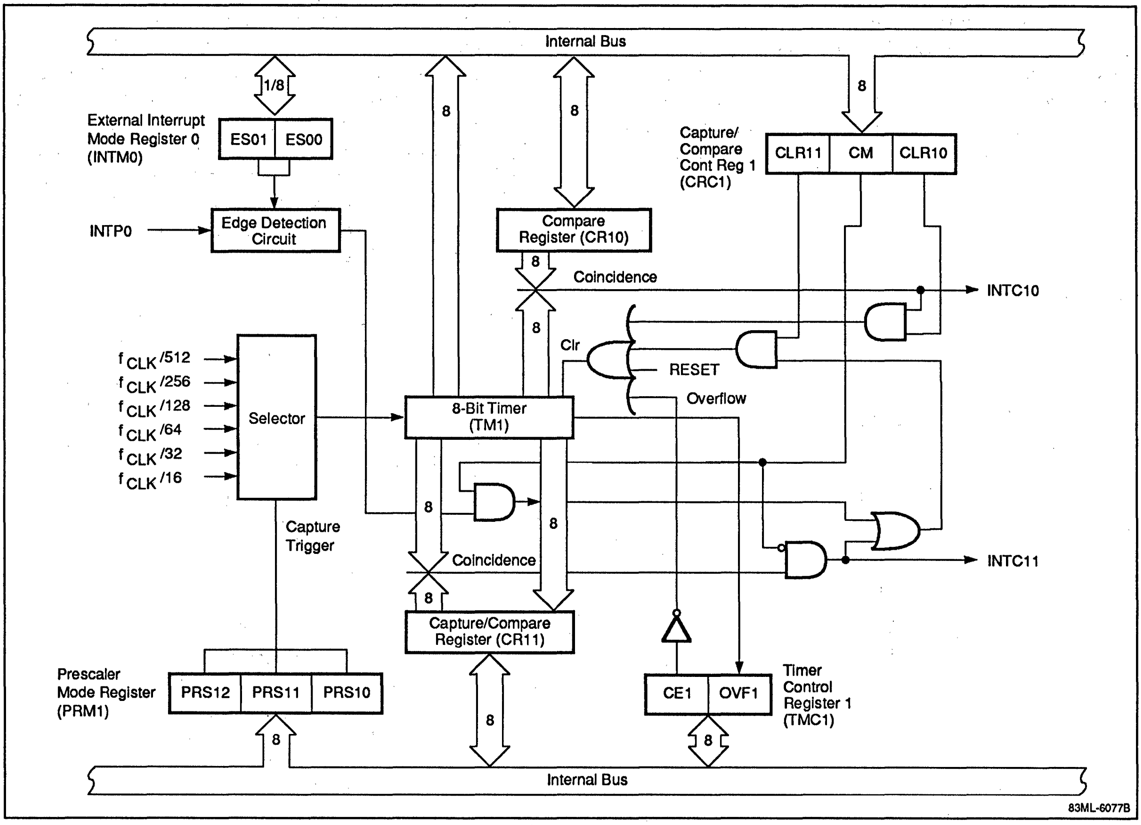
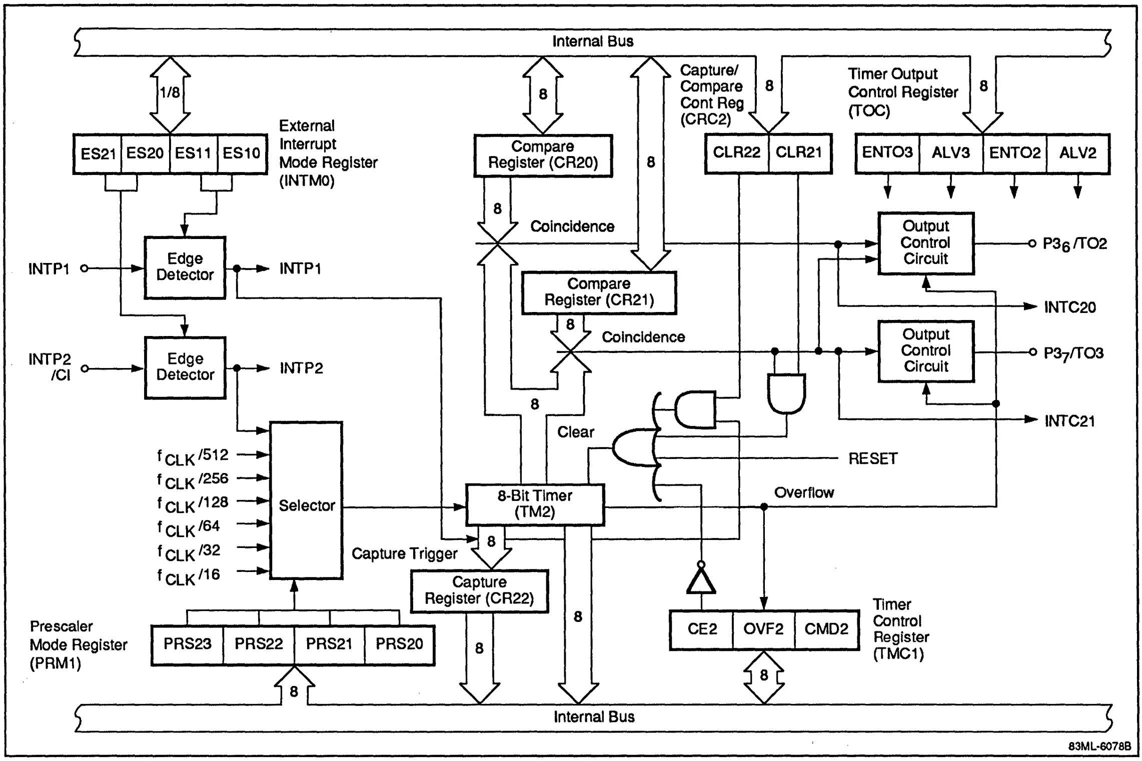


Figure 11. 8-Bit Timer/Counter 2



Interrupts

There are 20 interrupt request sources; each source is allocated a location in the vector table. (See table 2.) There is one software interrupt request and one of the remaining 19 interrupts is non-maskable. The software interrupt and the non-maskable interrupt are unconditionally received even in the DI state. These two interrupts possess the maximum priority. The maskable interrupt requests are subject to mask control by the setting of the interrupt mask flag.

There are default priorities associated with each maskable interrupt and these can be assigned to either of two programmable priority levels. Interrupts may be serviced by the vectored interrupt method where a branch to a desired service program is executed. Interrupts may also be handled by the macroservice function where a preassigned process is performed without program intervention.

Figure 12. 8-Bit Timer/Counter 3

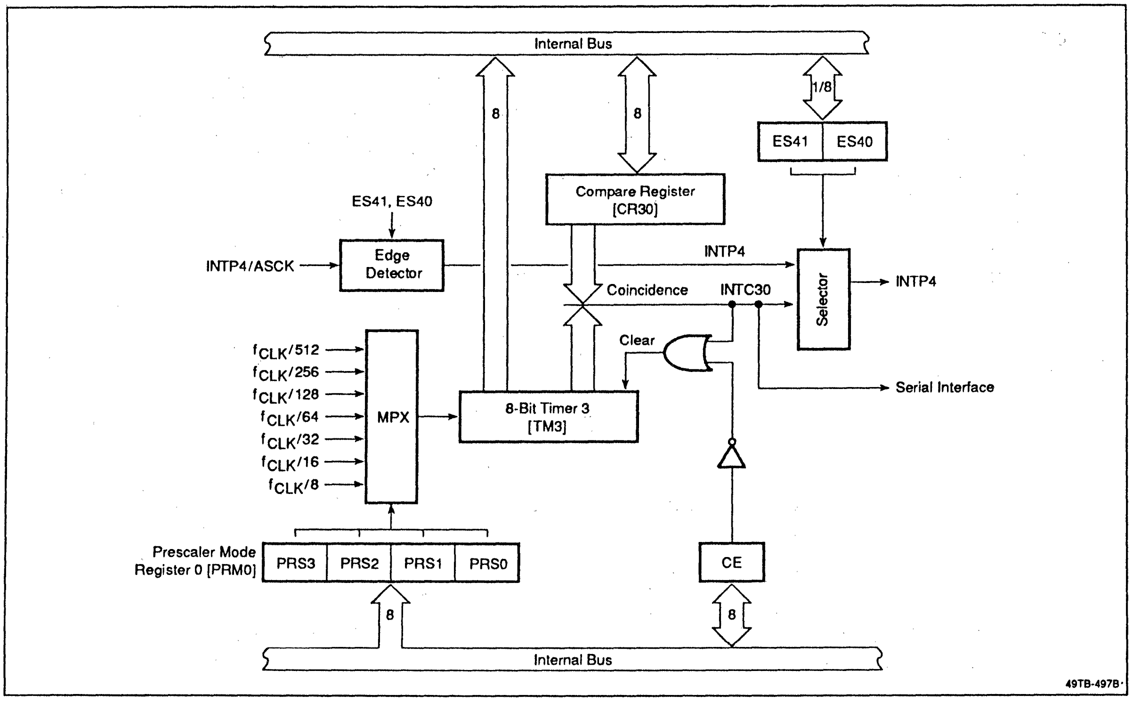


Table 2. Interrupt Sources and Vector Addresses

Interrupt Request Type	Default Priority	Interrupt Request Generation Source	Macroservice Mode	Vector Table Address
Software	None	BRK instruction execution	–	003EH
Non-maskable	None	NMI (pin input edge detection)	–	0002H
Maskable	0	INTP0 (pin input edge detection)	Yes	0006H
	1	INTP1 (pin input edge detection)	Yes	0008H
	2	INTP2 (pin input edge detection)	Yes	000AH
	3	INTP3 (pin input edge detection)	Yes	000CH
	4	INTC00 (TM0-CR00 coincidence signal generation)	Yes	0014H
	5	INTC01 (TM0-CR01 coincidence signal generation)	Yes	0016H
	6	INTC10 (TM1-CR10 coincidence signal generation)	Yes	0018H
	7	INTC11 (TM1-CR11 coincidence signal generation)	Yes	001AH
	8	INTC21 (TM2-CR21 coincidence signal generation)	Yes	001CH
	9	INTP4 (pin input edge detection)/INTC30 (TM3-CR30 coincidence signal generation)	Yes	000EH
	10	INTP5 (pin input edge detection)/INTAD (end of A/D conversion)	Yes	0010H
	11	INTC20 (TM2-CR20 coincidence signal generation)	Yes	0012H
	12	INTSER (generation of asynchronous serial interface receive error)	–	0020H
	13	INTSR (end of asynchronous serial interface reception)	Yes	0022H
	14	INTST (end of asynchronous serial interface transmission)	Yes	0024H
	15	INTCSI (end of clocked serial interface transmission)	Yes	0026H

Macroservice

The macroservice function can be programmed to transfer data from a special function register to memory or from memory to a special function register. Transfer events are triggered by interrupt requests and take place without software intervention. There are 17 interrupt requests where macroservicing can be executed. The macroservice function is controlled by the macroservice mode register and the macroservice channel pointer. The macroservice mode register assigns the macroservicing mode and the macroservice channel pointer indicates the address of the memory location pointers. The location of each register and its corresponding interrupt is shown in figure 13.

Refresh

The refresh signal is used with a pseudostatic RAM. The refresh cycle can be set to one of four intervals ranging from 2.6 to 21.3 μs. The refresh is timed to follow a read or write operation so there is no interference.

Standby Modes

Halt and stop functions reduce system power consumption. In the halt mode, the CPU stops and the system clock continues to run. A release of the halt mode is initiated by an unmasked interrupt request, an NMI, or a $\overline{\text{RESET}}$ input. In the stop mode, the CPU and system clock are both stopped, reducing the power consumption even further. The stop mode is released by an NMI input or a $\overline{\text{RESET}}$ input.

Figure 13. Macroservice Control Word Map

0FEDFH	Channel Pointer	INTSR
0FEDEH	Mode Register	
0FEDDH	Channel Pointer	INTST
0FEDCH	Mode Register	
0FEDBH	Channel Pointer	INTCSI
0FEDAH	Mode Register	
0FED9H	Channel Pointer	INTC10
0FED8H	Mode Register	
0FED7H	Channel Pointer	INTC11
0FED6H	Mode Register	
0FED5H	Channel Pointer	INTP4/INTC30
0FED4H	Mode Register	
0FED3H	Channel Pointer	INTP5/INTAD
0FED2H	Mode Register	
0FED1H	Channel Pointer	INTC00
0FED0H	Mode Register	
0FECFH	Channel Pointer	INTC01
0FECEH	Mode Register	
0FECDH	Channel Pointer	INTC20
0FECCH	Mode Register	
0FECBH	Channel Pointer	INTC21
0FECAH	Mode Register	
0FEC9H	Channel Pointer	INTP0
0FEC8H	Mode Register	
0FEC7H	Channel Pointer	INTP1
0FEC6H	Mode Register	
0FEC5H	Channel Pointer	INTP2
0FEC4H	Mode Register	
0FEC3H	Channel Pointer	INTP3
0FEC2H	Mode Register	

83ML-6087A

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

$T_A = +25°C$.

Item	Symbol	Conditions	Rating	Unit
Power supply voltages	V_{DD}		−0.5 to +7.0	V
	AV_{DD}		AV_{SS} to V_{DD} + 0.5	V
	AV_{SS}		−0.5 to +0.5	V
Input voltage	V_{I1}		−0.5 to AV_{REF1} + 0.5	V
Output voltage	V_O		−0.5 to V_{DD} + 0.5	V
Low-level output current	I_{OL}	One output pin	15	mA
		All output pins total	100	mA
High-level output current	I_{OH}	One output pin	−10	mA
		All output pins total	−50	mA
A/D converter reference input voltage	AV_{REF1}		−0.5 to AV_{DD} +0.3	V
D/A converter reference input voltage	AV_{REF2}		−0.5 to V_{DD} + 0.3	V
	AV_{REF3}		−0.5 to V_{DD} + 0.3	V
Operating temperature	T_{OPT}		−40 to +85	°C
Storage temperature	T_{STG}		−65 to +150	°C

Operating Frequency

Oscillation Frequency	T_A	V_{DD}
f_{XX} = 4 to 12 MHz	−40 to +85°C	+5V ± 10%

Capacitance

$T_A = +25°C$; $V_{DD} = V_{SS} = 0$ V.

Item	Symbol	Typ	Max	Unit	Conditions
Input capacitance	C_I		20	pF	f = 1 MHz; pins not used for measurement are at 0 V
Output capacitance	C_O		20	pF	
Input/output capacitance	C_{IO}		20	pF	

DC Characteristics

$T_A = -40$ to $+85°C$; $V_{DD} = +5\ V \pm 10\%$; $V_{SS} = AV_{SS} = 0\ V$.

Item	Symbol	Conditions	Min	Typ	Max	Unit
Low-level input voltage	V_{IL}		0		0.8	V
High-level input voltage	V_{IH1}	Except pins in Note 1	2.2		V_{DD}	V
	V_{IH2}	Pins in Note 1	0.8 V_{DD}		V_{DD}	V
Low-level output voltage	V_{OL1}	$I_{OL} = 2.0$ mA			0.45	V
	V_{OL2}	$I_{OL} = 8.0$ mA (pins in Note 2)			1.0	V
High-level output voltage	V_{OH1}	$I_{OH} = -1.0$ mA	$V_{DD} - 1.0$			V
	V_{OH2}	$I_{OH} = -100\ \mu A$	$V_{DD} - 0.5$			V
	V_{OH3}	$I_{OH} = -5.0$ mA (pins in Note 3)	2.0			V
Input leakage current	I_{LI}	$0V \leq V_I \leq V_{DD}$			±10	μA
Output leakage current	I_{LO}	$0V \leq V_O \leq V_{DD}$			±10	μA
AV_{REF} current	AI_{REF}	Operating mode, $f_{XX} = 12$ MHz		1.5	5.0	mA
V_{DD} power supply current	I_{DD1}	Operating mode, $f_{XX} = 12$ MHz		20	40	mA
	I_{DD2}	HALT mode, $f_{XX} = 12$ MHz		7	20	mA
Data retention voltage	V_{DDDR}	STOP mode	2.5		5.5	V
Data retention current	I_{DDDR}	STOP mode $V_{DDDR} = 2.5$ V		2	20	μA
		$V_{DDDR} = 5\ V \pm 10\%$		5	50	μA
Pullup resistor	R_L	$V_I = 0$ V	15	40	80	kΩ

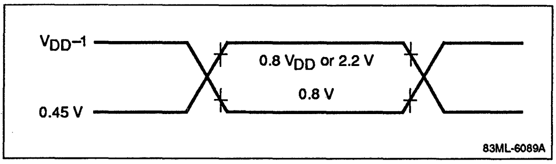
Notes:

(1) X1, X2, $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/$\overline{CI}$, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/$\overline{SCK}$, $P3_3$/SO/SB0, and $\overline{EA}$ pins.

(2) Pins $P1_0$-$P1_7$, $P4_0$-$P4_7$/AD_0-AD_7 and $P5_0$-$P5_7$/A_8-A_{15}.

(3) Pins $P0_0$-$P0_7$.

Figure 14. Voltage Thresholds for Timing Measurements



Read/Write Operation

$T_A = -40$ to $+85°C$; $V_{DD} = +5\ V \pm 10\%$; $V_{SS} = 0\ V$; $f_{XX} = 12\ MHz$; $C_L = 100\ pF$. See figures 15, 16, and 17.

Item	Symbol	Conditions	Min	Max	Unit
X1 input clock cycle time	t_{CYX}		82	250	ns
Address setup time to ASTB ↓	t_{SAST}		52		ns
Address hold time from ASTB ↓ (Note 1)	t_{HSTA}	$R_L = 5\,k\Omega$, $C_L = 50\,pF$	25		ns
Address to $\overline{\text{RD}}$ ↓ delay time	t_{DAR}		129		ns
Address float time from $\overline{\text{RD}}$ ↓	t_{FAR}		11		ns
Address to data input time	t_{DAID}			228	ns
ASTB ↓ to data input time	t_{DSTID}			181	ns
$\overline{\text{RD}}$ ↓ to data input time	t_{DRID}			99	ns
ASTB ↓ to $\overline{\text{RD}}$ ↓ delay time	t_{DSTR}		52		ns
Data hold time from $\overline{\text{RD}}$ ↑	t_{HRID}		0		ns
$\overline{\text{RD}}$ ↑ to address active time	t_{DRA}		124		ns
$\overline{\text{RD}}$ ↑ to ASTB ↑ delay time	t_{DRST}		124		ns
$\overline{\text{RD}}$ low-level width	t_{WRL}		124		ns
ASTB high-level width	t_{WSTH}		52		ns
Address to $\overline{\text{WR}}$ ↓ delay time	t_{DAW}		129		ns
$\overline{\text{ASTB}}$ ↓ to data output time	t_{DSTOD}			142	ns
$\overline{\text{WR}}$ ↓ to data output time	t_{DWOD}			60	ns
ASTB ↓ to $\overline{\text{WR}}$ ↓ delay time	t_{DSTW1}		52		ns
	t_{DSTW2}	Refresh mode	129		ns
Data setup time to WR ↑	t_{SODWR}		146		ns
Data setup time to $\overline{\text{WR}}$ ↓ (Note 1)	t_{SODWF}	Refresh mode	22		ns
Data hold time from $\overline{\text{WR}}$ ↑	t_{HWOD}		20		ns
$\overline{\text{WR}}$ ↑ to ASTB ↑ delay time	t_{DWST}		42		ns
WR low-level width	t_{WWL1}		196		ns
	t_{WWL2}	Refresh mode	114		ns
Address to $\overline{\text{WAIT}}$ ↓ input time	t_{DAWT}			146	ns
ASTB ↓ to $\overline{\text{WAIT}}$ ↓ input time	t_{DSTWT}			84	ns
$\overline{\text{WAIT}}$ hold time from X1 ↓	t_{HWTX}		0		ns
$\overline{\text{WAIT}}$ setup time to X1 ↑	t_{SWTX}		0		ns

Notes:

(1) The hold time includes the time during which V_{OH} and V_{OL} are retained under the following load conditions: $C_L = 100\ pF$ and $R_L = 2\ k\Omega$.

Figure 15. Read Operation Timing

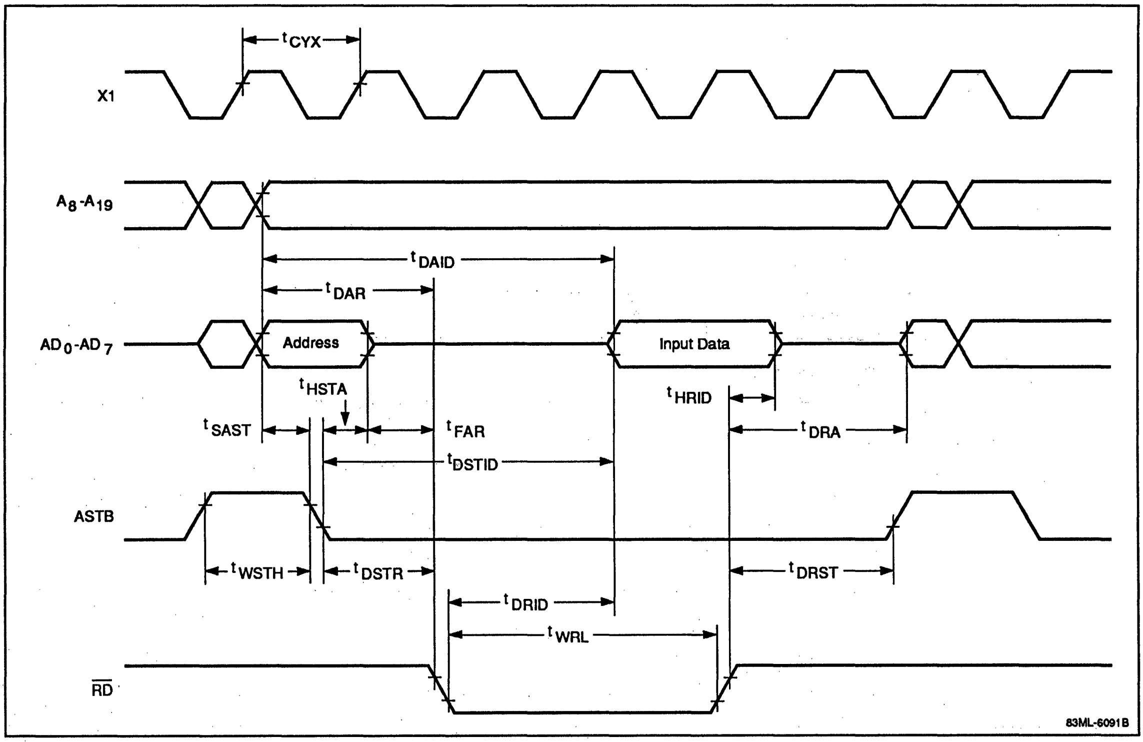


Figure 16. Write Operation Timing

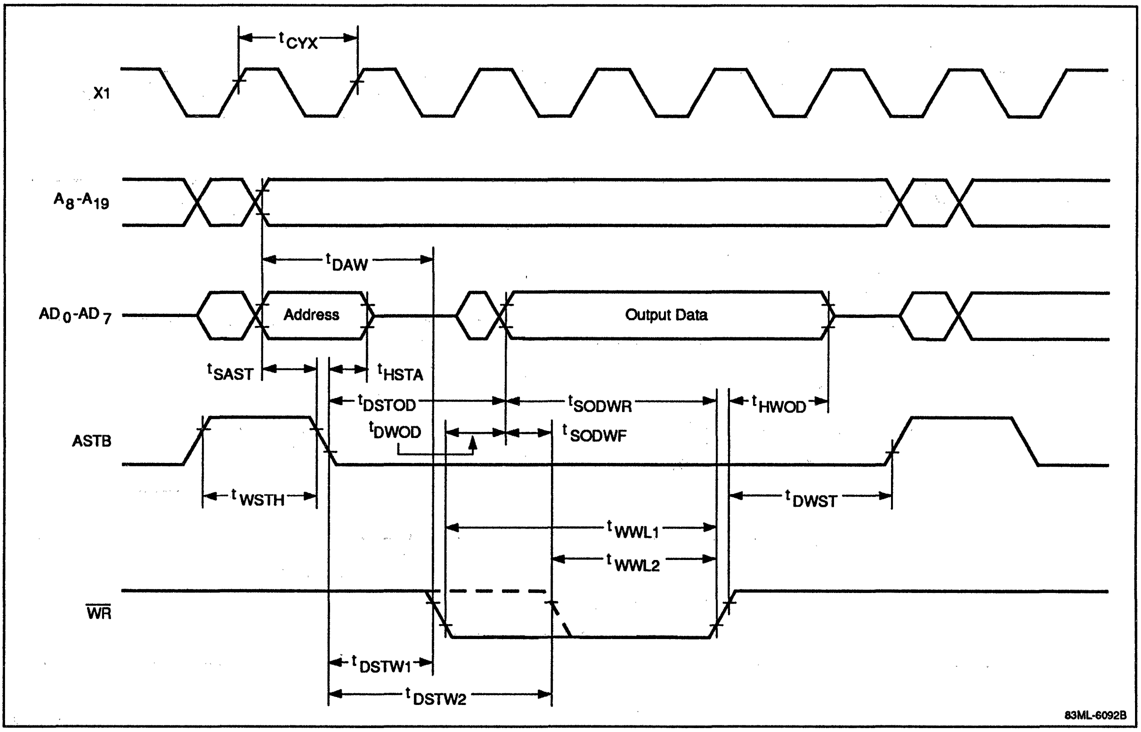
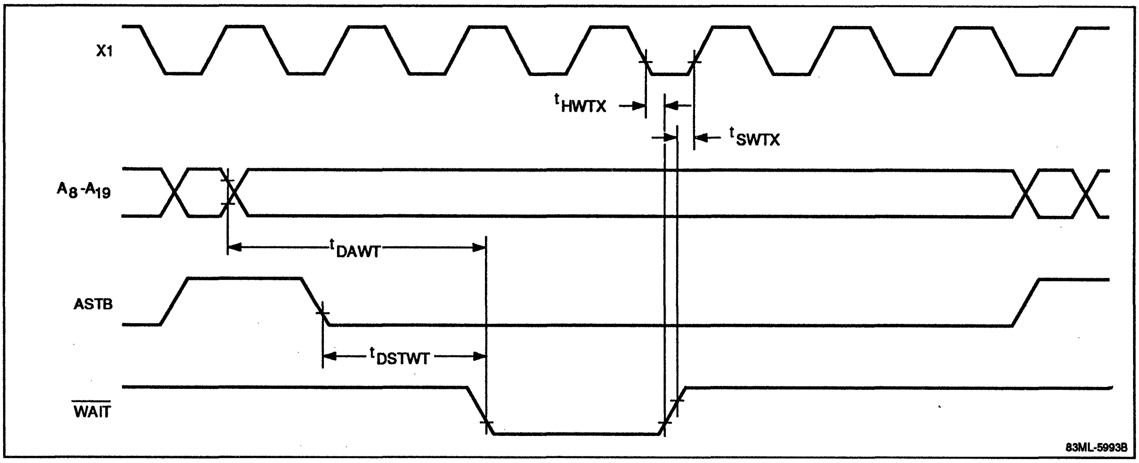


Figure 17. External $\overline{WAIT}$ Input Timing



Serial Port Operation

T_A = −40 to +85°C; V_{DD} = +5 V ± 10%; V_{SS} = 0 V; f_{XX} = 12 MHz; C_L = 100 pF. See figures 18, 19, and 20.

Item	Symbol	Conditions		Min	Max	Unit
Serial clock cycle time	t_{CYSK}	Input	External clock	1.0		μs
		Output	Internal clock/16	1.3		μs
			Internal clock/64	5.3		μs
Serial clock low-level width	t_{WSKL}	Input	External clock	420		ns
		Output	Internal clock/16	556		ns
			Internal clock/64	2.5		μs
Serial clock high-level width	t_{WSKH}	Input	External clock	420		ns
		Output	Internal clock/16	556		ns
			Internal clock/64	2.5		μs
SI, SB0 setup time to $\overline{\text{SCK}}$ ↑	t_{SSSK}			150		ns
SI, SB0 hold time from $\overline{\text{SCK}}$ ↑	t_{HSSK}			400		ns
SO/SB0 output delay time from $\overline{\text{SCK}}$ ↓	t_{DSBSK1}	CMOS push-pull output (3-line serial I/O mode)		0	300	ns
	t_{DSBSK2}	Open-drain output (SBI mode), R_L = 1 kΩ		0	800	ns
SB0 high, hold time from $\overline{\text{SCK}}$ ↑	t_{HSBSK}	SBI mode		4		t_{CYX}
SB0 low, setup time to $\overline{\text{SCK}}$ ↓	t_{SSBSK}	SBI mode		4		t_{CYX}
SB0 low-level width	t_{WSBL}			4		t_{CYX}
SB0 high-level width	t_{WSBH}			4		t_{CYX}
RxD setup time to $\overline{\text{SCK}}$ ↑	t_{SRXSK}			80		ns
RxD hold time after $\overline{\text{SCK}}$ ↑	t_{HSKRX}			80		ns
$\overline{\text{SCK}}$ ↓ to TxD delay time	t_{DSKTX}				210	ns

Figure 18. Clock-Synchronized Serial Interface Timing; Three-Line I/O Mode

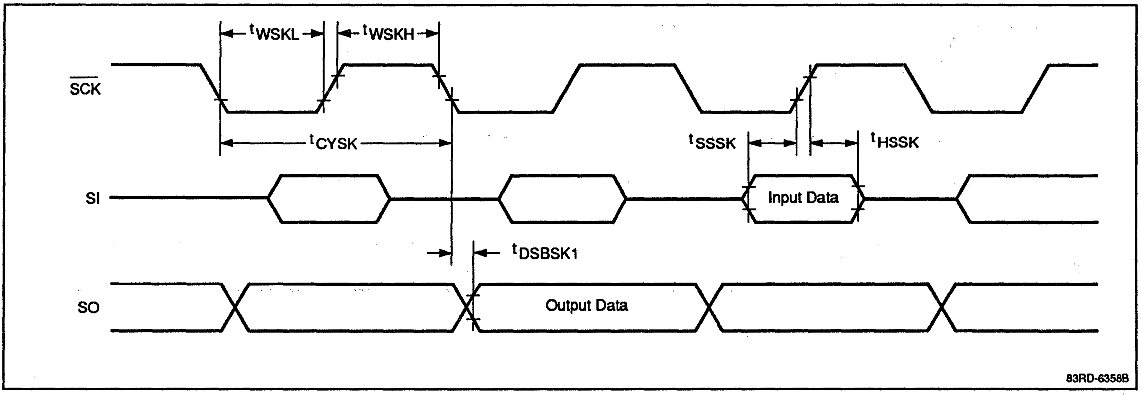


Figure 19. Clock-Synchronized Serial Interface Timing; SBI Mode

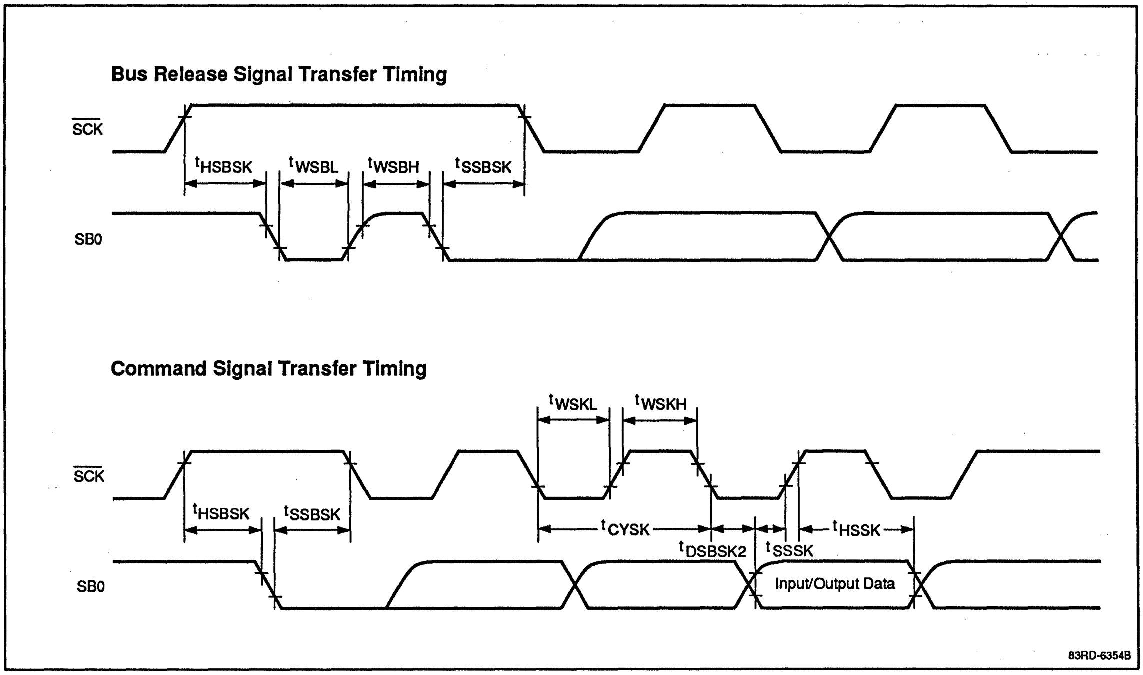
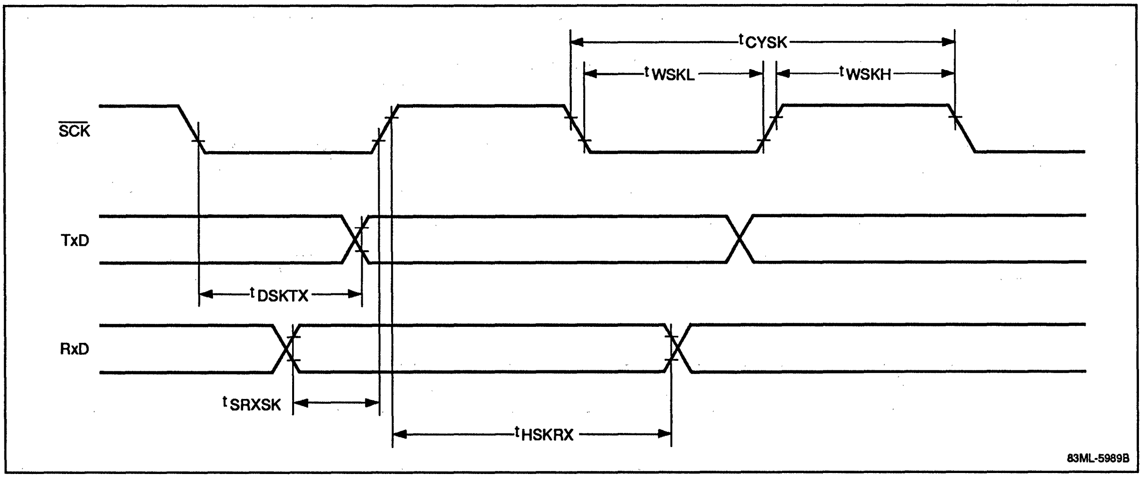


Figure 20. Asynchronous Mode Timing



A/D Converter Operation

$T_A = -40$ to $+85$°C; $V_{DD} = +5$ V ±10%; $V_{SS} = AV_{SS} = 0$ V.

Item	Symbol	Conditions	Min	Typ	Max	Unit
Resolution			8			Bit
Full-scale error		$AV_{REF} = 4.0$ V to V_{DD}; $T_A = -10$ to $+70$°C			0.4	%
		$AV_{REF} = 3.4$ V to V_{DD}; $T_A = -10$ to $+70$°C			0.8	%
		$AV_{REF} = 4.0$ V to V_{DD}			0.8	%
Quantization error					±1/2	LSB
Conversion time	t_{CONV}	83 ns ≦ t_{CYX} ≦ 125 ns	360			t_{CYX}
		125 ns ≦ t_{CYX} ≦ 250 ns	240			t_{CYX}
Sampling time	t_{SAMP}	83 ns ≦ t_{CYX} ≦ 125 ns	72			t_{CYX}
		125 ns ≦ t_{CYX} ≦ 250 ns	48			t_{CYX}
Analog input voltage	V_{IAN}		0		AV_{REF}	V
Input impedance	R_{AN}			1000		MΩ
Analog reference voltage	AV_{REF}		3.4		V_{DD}	V
AV_{REF} current	AI_{REF}	Operating mode, $f_{XX} = 12$ MHz		1.5	5.0	mA
		STOP mode		0.2	1.5	mA

D/A Converter Operation

$T_a = -40$ to $+85$°C; $AV_{REF2} = V_{DD} = +5$ V ±10%; $AV_{REF3} = V_{SS} = 0$ V.

Item	Symbol	Conditions	Min	Typ	Max	Unit
Resolution					8	Bit
Absolute accuracy		$AV_{REF2} = V_{DD} = 5$ V; $AV_{REF3} = V_{SS} = 0$ V; Load conditions: 2 MΩ, 30 pF			1	LSB
		$AV_{REF2} = 0.75$ V_{DD}; $AV_{REF3} = 0.25$ V_{DD}; Load conditions: 2 MΩ, 30 pF				LSB
Settling time	Undefined				10	μs
Analog reference voltage	V_{AVREF2}		0.75 V_{DD}		V_{DD}	V
Analog reference voltage	V_{AVREF3}		0		0.25 V_{DD}	V
Reference power input current	AI_{REF2}		0		5	mA
Reference power input current	AI_{REF3}		−5.0		0	mA
Output resistance	R_O			24		kΩ

Interrupt Timing Operation

Item	Symbol	Conditions	Min	Max	Unit
NMI low-level width	t_{WNIL}		10		μs
NMI high-level width	t_{WNIH}		10		μs
INTP0-INTP5 low-level width	t_{WITL}		24		t_{CYX}
INTP0-INTP5 high-level width	t_{WITH}		24		t_{CYX}
$\overline{RESET}$ low-level width	t_{WRSL}		10		μs
$\overline{RESET}$ high-level width	t_{WRSH}		10		μs

Note: See figures 21 and 22.

Figure 21. Interrupt Input Timing

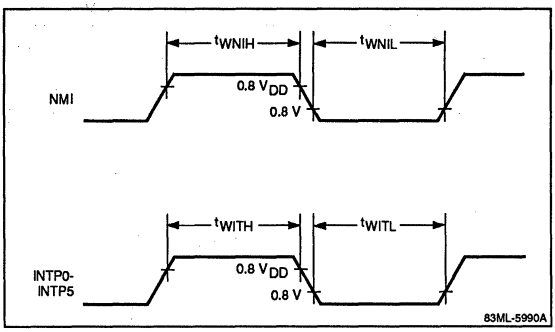
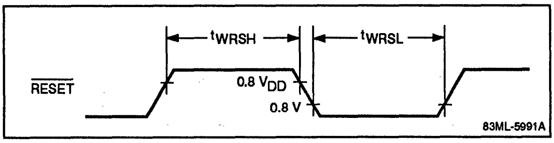


Figure 22. Reset Input Timing



Data Retention Characteristics

$T_A = -40$ to $+85$°C.

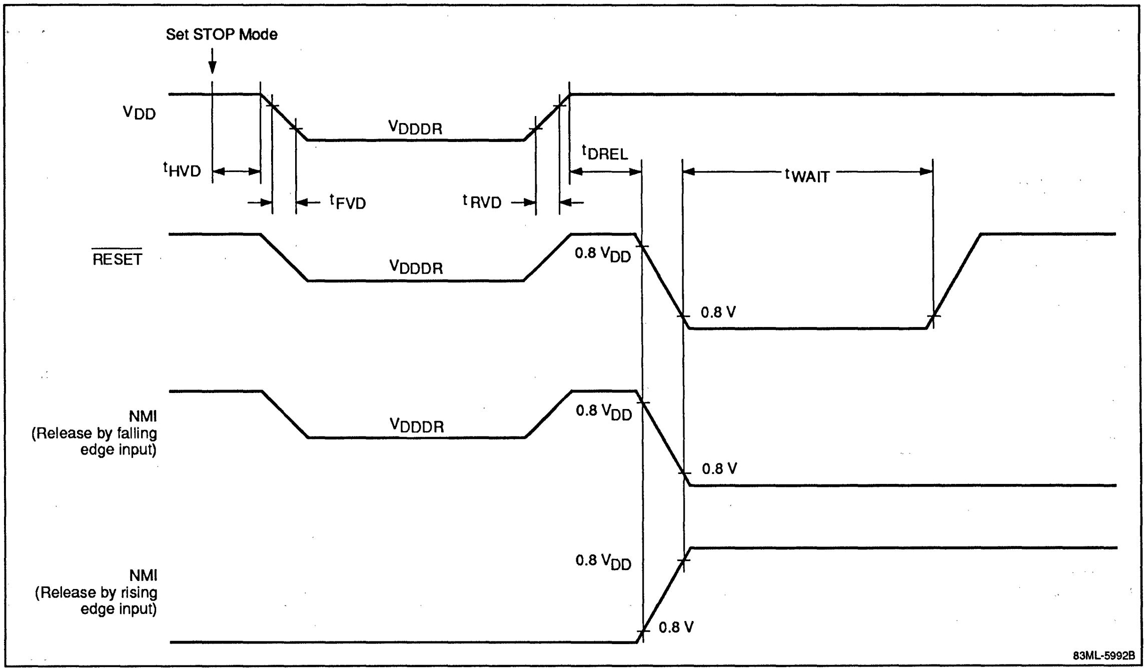
Item	Symbol	Conditions	Min	Typ	Max	Unit
Data retention voltage	V_{DDDR}	STOP mode	2.5		5.5	V
Data retention current	I_{DDDR}	$V_{DDDR} = 2.5$ V		2	15	μA
		$V_{DDDR} = 5$ V ±10%		5	20	μA
V_{DD} rise time	t_{RVD}		200			μs
V_{DD} fall time	t_{FVD}		200			μs
V_{DD} retention time (for STOP mode setup)	t_{HVD}		0			ms
STOP release signal input time	t_{DREL}		0			ms
Oscillation stabilization wait time	t_{WAIT}	Crystal oscillator	30			ms
		Ceramic resonator	5			ms
Low-level input voltage	V_{IL}	Specified pins (Note 1)	0		0.1 V_{DDDR}	V
High-level input voltage	V_{IH}		0.9 V_{DDDR}		V_{DDDR}	V

Notes:

(1) $\overline{RESET}$, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2/CI, $P2_4$/INTP3, $P2_5$/INTP4/ASCK, $P2_6$/INTP5, $P2_7$/SI, $P3_2$/$\overline{SCK}$, $P3_3$/SO/SB0, and $\overline{EA}$ pins.

(2) See figure 23.

Figure 23. Data Retention Characteristics



Timing Dependent on t_{CYX}

Item	Symbol	Calculation Formula	Min/Max	12 MHz	Unit
X1 input clock cycle time	t_{CYX}		Min	82	ns
Address setup time to ASTB ↓	t_{SAST}	$t_{CYX}-30$	Min	52	ns
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	$2t_{CYX}-35$	Min	129	ns
Address float time from $\overline{RD}$ ↓	t_{FAR}	$t_{CYX}/2-30$	Min	11	ns
Address to data input time	t_{DAID}	$(4+2n)\,t_{CYX}-100$	Max	228	ns
ASTB ↓ to data input time	t_{DSTID}	$(3+2n)\,t_{CYX}-65$	Max	181	ns
$\overline{RD}$ ↓ to data input time	t_{DRID}	$(2+2n)\,t_{CYX}-65$	Max	99	ns
ASTB ↓ to $\overline{RD}$ ↓ delay time	t_{DSTR}	$t_{CYX}-30$	Min	52	ns
$\overline{RD}$ ↑ to address active time	t_{DRA}	$2t_{CYX}-40$	Min	124	ns
$\overline{RD}$ ↑ to ASTB ↑ delay time	t_{DRST}	$2t_{CYX}-40$	Min	124	ns
$\overline{RD}$ low-level width	t_{WRL}	$(2+2n)\,t_{CYX}-40$	Min	124	ns
ASTB high-level width	t_{WSTH}	$t_{CYX}-30$	Min	52	ns
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	$2t_{CYX}-35$	Min	129	ns
ASTB ↓ to data output time	t_{DSTOD}	$t_{CYX}+60$	Max	142	ns
ASTB ↓ to $\overline{WR}$ ↓ delay time	t_{DSTW1}	$t_{CYX}-30$	Min	52	ns
	t_{DSTW2}	$2t_{CYX}-35$ (refresh mode)	Min	129	ns
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	$(3+2n)\,t_{CYX}-100$	Min	146	ns
Data setup time to $\overline{WR}$ ↓	t_{SODWF}	$t_{CYX}-60$ (refresh mode)	Min	22	ns
$\overline{WR}$ ↑ to ASTB ↑ delay time	t_{DWST}	$t_{CYX}-40$	Min	42	ns
$\overline{WR}$ low-level width	t_{WWL1}	$(3+2n)\,t_{CYX}-50$	Min	196	ns
	t_{WWL2}	$(2+2n)\,t_{CYX}-50$ (refresh mode)	Min	114	ns
Address to $\overline{WAIT}$ ↓ input time	t_{DAWT}	$3t_{CYX}-100$	Max	146	ns
ASTB ↓ to $\overline{WAIT}$ ↓ input time	t_{DSTWT}	$2t_{CYX}-80$	Max	84	ns

Notes:

(1) n indicates the number of wait states.

Figure 24. Recommended Oscillator Circuit

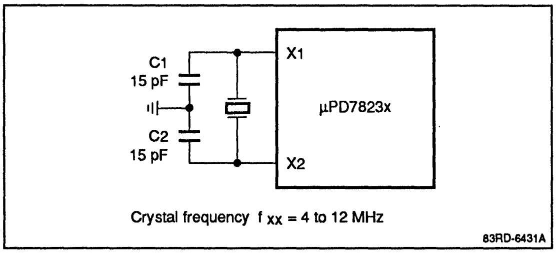
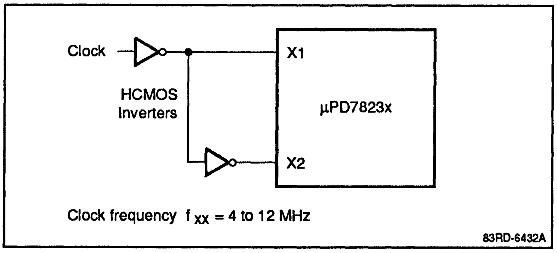


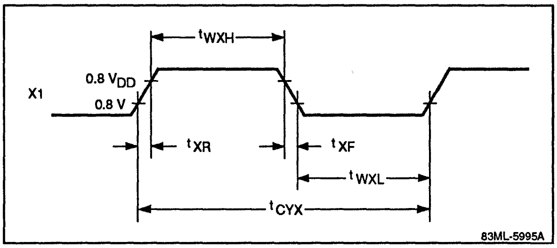
Figure 25. Recommended External Clock Circuit



External Clock Operation

Item	Symbol	Conditions	Min	Max	Unit
X1 input low-level width	t_{WXL}		30	130	ns
X1 input high-level width	t_{WXH}		30	130	ns
X1 input rise time	t_{XR}		0	30	ns
X1 input fall time	t_{XF}		0	30	ns
X1 input clock cycle time	t_{CYX}		82	250	ns

Figure 26. External Clock Timing



μPD78P238 PROGRAMMING

In the 78P238, the mask ROM of 78234 is replaced by a one-time programmable ROM (OTP ROM). The ROM is 32K × 8 bits and can be programmed using a general-purpose PROM writer with a μPD27C256A programming mode.

The PA-78P238GC/GJ/L are the socket adaptors used for configuring the μPD78P238 to fit a standard PROM socket.

Refer to tables 3 through 6 and figures 27 and 28 for special information applicable to PROM programming.

Table 3. Pin Functions During EPROM Programming

Pin		Function
$P0_0$-$P0_7$	A_0-A_7	Input pins for PROM write/verify operations
$P5_0/A_8$	A_8	Input pin for PROM write/verify operation
$P2_1$/INTP0	A_9	Input pin for PROM write/verify operation
$P5_2$-$P5_6/A_{10}$-A_{14}	A_{10}-A_{14}	Input pins for PROM write/verify operations
$P4_0$-$P4_7/AD_0$-AD_7	D_0-D_7	Data pins for PROM write/verify operations
$P6_5/\overline{WR}$	$\overline{CE}$	Strobe data into the PROM
$P6_4/\overline{RD}$	$\overline{OE}$	Enable a data read from the PROM
$P2_0$/NMI	NMI	PROM programming mode is entered by applying a high voltage to this pin
$\overline{RESET}$	$\overline{RESET}$	PROM programming mode requires applying a low voltage to this pin
$\overline{EA}$	V_{PP}	High voltage applied to this pin for program write/verify
V_{DD}	V_{DD}	Positive power supply pin
V_{SS}	V_{SS}	Ground

Table 4. Summary of Operation Modes for PROM Programming

Mode	NMI	RESET	CE	OE	V_{pp}	V_{DD}	D_0-D_7
Program write	+12.5 V	L	L	H	+12.5 V	+6 V	Data input
Program verify	+12.5 V	L	H	L	+12.5 V	+6 V	Data output
Program inhibit	+12.5 V	L	H	H	+12.5 V	+6 V	High Z
Read out	+12.5 V	L	L	L	+5 V	+5 V	Data output
Output disable	+12.5 V	L	L	H	+5 V	+5 V	High Z
Standby	+12.5 V	L	H	L/H	+5 V	+5 V	High Z

Notes:
When +12.5 V is applied to V_{pp} and +6 V to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ cannot be set to low level (L) simultaneously.

Table 5. DC Programming Characteristics

$T_A = 25 \pm 5°C$, $V_{IP} = 12.5 \pm 0.5$ V applied to NMI pin, $V_{SS} = 0$ V.

Parameter	Symbol	Symbol*	Condition	Min	Typ	Max	Unit
High-level input voltage	V_{IH}	V_{IH}		2.4		$V_{DDP} + 0.3$	V
Low-level input voltage	V_{IL}	V_{IL}		−0.3		0.8	V
Input leakage current	V_{LIP}	V_{LI}	$0 \le V_1 \le V_{DDP}$			10	µA
High-level output voltage	V_{OH1}	V_{OH}	$I_{OH} = -400$ µA	2.4			V
	V_{OH2}	V_{OH2}	$I_{OH} = -100$ µA	$V_{DD} - 0.7$			V
Low-level output voltage	V_{OL}	V_{OL}	$I_{OH} = 2.1$ mA			0.45	V
Output leakage current	I_{LO}		$0 \le V_O \le V_{DPP}$, $\overline{OE} = V_{IH}$			10	µA
NMI pin high-voltage input current	I_{IP}					±10	µA
V_{DDP} power voltage	V_{DDP}	V_{CC}	Program memory write mode	5.75	6.0	6.25	V
			Program memory read mode	4.5	5.0	5.5	V
V_{PP} power voltage	V_{PP}	V_{PP}	Program memory write mode	12.2	12.5	12.8	V
			Program memory read mode		$V_{PP} = V_{DDP}$		V
V_{DDP} power current	I_{DD}	I_{CC}	Program memory write mode		5	30	mA
			Program memory read mode $\overline{CE} = V_{IL}$, $V_I = V_{IH}$		5	30	mA
V_{PP} power current	I_{PP}	I_{PP}	Program memory write mode $\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$		5	30	mA
			Program memory read mode		1	100	µA

* Corresponding symbols of the µPD27C256A.

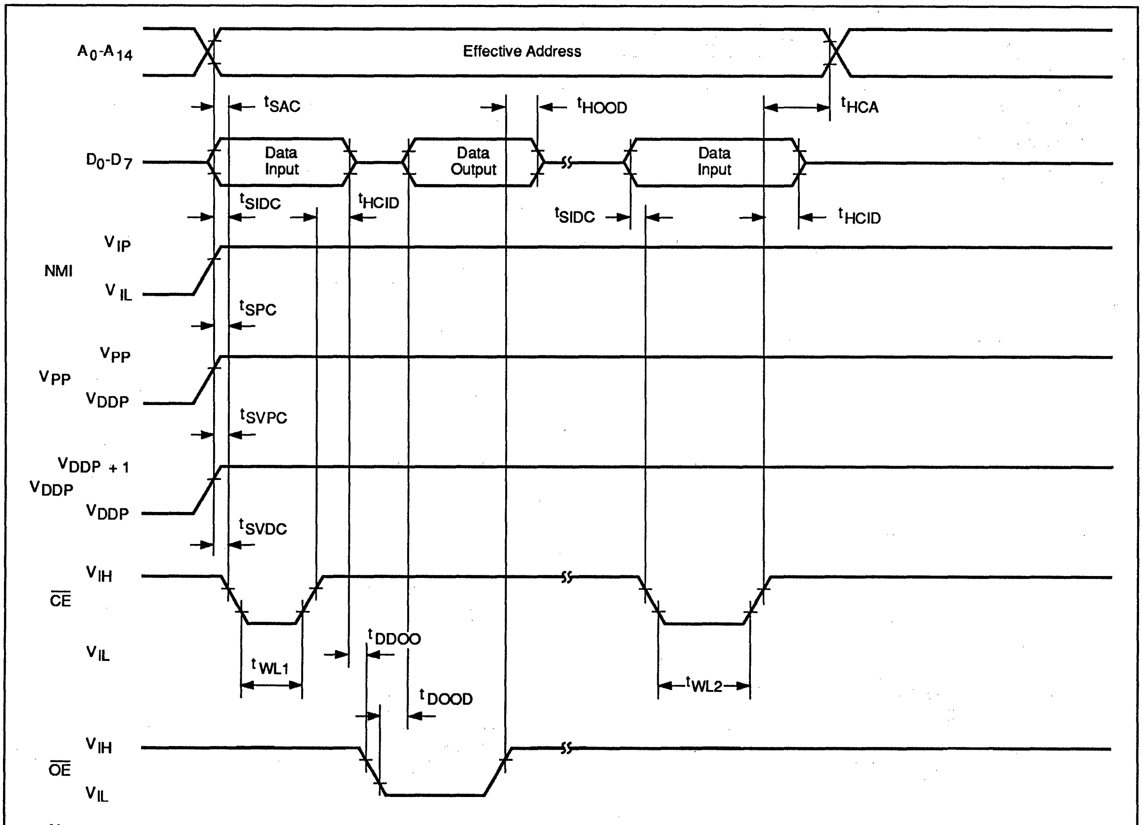
Table 6. AC Programming Characteristics

$T_A = 25 \pm 5°C$, $V_{IP} = 12.5 \pm 0.5$ V applied to NMI pin, $V_{SS} = 0$ V, $V_{DD} = 6 \pm 0.25$ V, $V_{PP} = 12.5 \pm 0.3$ V.

Parameter	Symbol	Symbol*	Condition	Min	Typ	Max	Unit
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}		2			µs
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}		2			µs
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}		2			µs
Address hold time from $\overline{CE}$ ↑	t_{HCA}	t_{AH}		2			µs
Input data hold time from $\overline{CE}$ ↑	t_{HCID}	t_{DH}		2			µs
Output data hold time to $\overline{OE}$ ↑	t_{HOOD}	t_{DF}		0		130	ns
V_{PP} setup time to $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}		1			ms
V_{DDP} setup time to $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}		1			ms
Initial program pulse width	t_{WL1}	t_{PW}		0.95	1.0	1.05	ms
Additional program pulse width	t_{WL2}	t_{OPW}		2.85		78.75	ms
NMI high-voltage input setup time (vs. $\overline{CE}$ ↓)	t_{SPC}			2			µs
Address to data output time	t_{DAOD}	t_{ACC}	$\overline{CE} = \overline{OE} = V_{IL}$			200	ns
$\overline{CE}$ ↓ to data output time	t_{DCOD}	t_{CE}	$\overline{OE} = V_{IL}$			200	ns
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}	$\overline{CE} = V_{IL}$			75	ns
Data hold time from $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	$\overline{CE} = V_{IL}$	0		60	ns
Data hold time from address	t_{HAOD}	t_{OH}	$\overline{CE} = \overline{OE} = V_{IL}$	0			ns

* Corresponding symbols of the µPD27C256A.

Figure 27. PROM Write Mode Timing



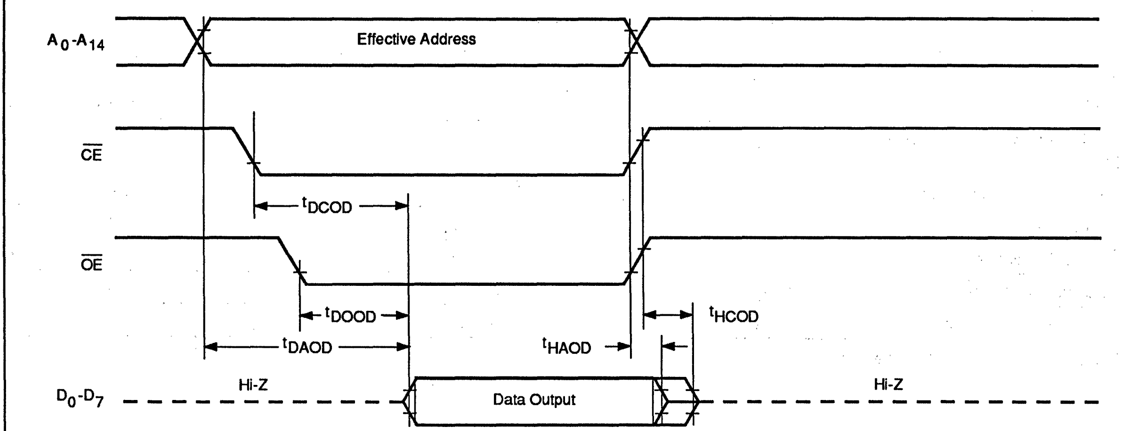
Notes:

(1) V_{DDP} must be applied before applying V_{PP}. It should be removed after removing V_{PP}.

(2) V_{PP} must not exceed +13 V, including overshoot.

83ML-5996B

Figure 28. PROM Read Mode Timing



83ML-5997B

PROM Write Procedure

(1) Connect the $\overline{\text{RESET}}$ pin to a low level and apply +12.5 V to the NMI pin.

(2) Apply +6 V to the V_{DD} pin and +12.5 V to the V_{pp} pin.

(3) Provide the initial address.

(4) Provide write data.

(5) Provide 1-ms program pulse (active low) to the $\overline{\text{CE}}$ pin.

(6) This bit is now verified with a pulse (active low) to the $\overline{\text{OE}}$ pin. If the data has been written, proceed to step 8; if not, repeat steps 4 to 6. If the data cannot be correctly written after 25 attempts, go to step 7.

(7) Classify as defective and stop write operation.

(8) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps 4 to 6.

(9) Increment the address.

(10) Repeat steps 4 to 9 until the end address.

PROM Read Procedure

(1) Fix the $\overline{\text{RESET}}$ pin to a low level and apply +12.5 V to the NMI pin.

(2) Apply +5 V to the V_{DD} and V_{pp} pins.

(3) Input the address of the data to be read to pins A_0-A_{14}.

(4) Read mode is entered with a pulse (active low) on both the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

(5) Data is output to the D_0 to D_7 pins.

INSTRUCTION SET

All microcomputers in the μPD7823x family have a 1-byte instruction lookahead buffer. This allows the first byte of the next opcode in program memory to be fetched while the current opcode is being executed. This pipeline architecture allows instruction fetch and excute cycles to overlap. An instruction can be fetched from program memory while data is being read from or written to RAM or an I/O port.

The advantage of the pipeline is that one instruction can be executed while another is being fetched, virtually halving the time required for these two operations and thereby reducing overall program execution time.

Operands and Operations

Refer to tables 7 and 8 for the meanings of symbols in the operand and operations columns of the Instruction Set table.

Specify operands in accordance with the rules of operand representation; for details, refer to the assembler specifications. If two or more description methods are available, select one. The symbols +, –, #, !, $, /, [ ], and & are keywords and must be used in conjunction with each instruction.

When describing immediate data as a label, use one of the following modifiers: +, –, #, !, $, /, [ ], and &. Symbols r and rp can be described in both the function name and absolute name.

Table 7. Operands

Symbol	Meaning
+	Autoincrement
–	Autodecrement
#	Immediate data
!	Absolute address
$	Relative address
/	Bit inversion
[ ]	Indirect addressing
&	Subbank
r	Register Function name: X, A, C, B, E, D, L, H Absolute name: R0 to R7
r1	Register group 1: C, B
rp	Register pair Function name: AX, BC, DE, HL Absolute name: RP0 to RP3
sfr	Special function register: P0, P2-P7, P0H, P0L, RTPC, CR10, CR11, CR20, CR21, CR22, CR30, PM0, PM3, PM5, PM6, PMC3, PUO, CRC0-CRC2, TOC, TM1-TM3, TMC0, TMC1, PRM0, PRM1, ADM, ADCR, CSIM, SBIC, SIO, ASIM, ASIS, RxB, TxS, BRGC, STBC (dedicated instruction only), MM, PW, RFM, IF0L, IF0H, MK0L, MK0H, PR0L, PR0H, ISM0L, ISM0H, INTM0, INTM1, IST
sfrp	Special function register pair: CR00-CR02, TM0, IF0, MK0, PR0, ISM0
mem	Memory address indirectly addressed Register indirect mode: [DE], [HL], [DE+], [HL+], [DE–], [HL–] Base mode: [DE+byte], [HL+byte], [SP+byte] Indexed mode: word[A], word[B], word[DE], word [HL]
mem1	Memory address addressed by means of indirect addressing group 1: [DE], [HL]
saddr	Memory address indirectly addressed: FE20H-FF1FH immediate data or label
saddrp	Memory address addressed by means of direct addressing pair: FE20H-FF1EH immediate data (LSB=0; odd address) or label
addr16	16-bit address: 0000H-FEFFH immediate data or label
addr11	11-bit address: 800H-FFFH immediate data or label
addr5	5-bit address: 40H-7EH immediate data or label
word	16-bit data: 16-bit immediate data or label
byte	8-bit data: 8-bit immediate data or label
bit	3-bit data: 3-bit immediate data or label
n	Number of shift bits: 3-bit immediate data (0-7)
RBn	Register bank: RB0-RB3

Table 8. Registers and Flags

Symbol	Meaning
A	A register; 8-bit accumulator
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R7	Registers 0 to 7 (absolute names)
AX	Register pair (AX); 16-bit accumulator
BC	Register pair (BC)
DE	Register pair (DE)
HL	Register pair (HL)
RP0-RP3	Register pairs 0 to 3 (absolute names)
PC	Program counter
SP	Stack pointer
PSW	Program status word
CY	Carry flag
AC	Auxiliary flag
Z	Zero flag
RBS1-RBS0	Register bank select flags
IE	Interrupt enable flag
STBC	Standby control register

Table 8. Registers and Flags (cont)

Symbol	Meaning
( )	Memory contents indicated by address or register contents in ( )
xxH	Hexadecimal number
x_H, x_L	Higher 8 bits and lower 8 bits of 16-bit register pair

Clocks

The clock field specifies the number of clocks required under the conditions defined by the four column headings as follows:

IROM	Program in internal ROM is executed.
IRAM	Program in external ROM is executed and internal RAM is accessed.
SFR	Program in external ROM is executed and special function register is accessed.
EMEM	Program in external ROM is executed and external memory is accessed.

In a shift/rotate instruction, n in the clock field indicates the number of bits by which data is shifted.

The hyphen (-) indicates a range of values; for example 10-13 means 10, 11, 12, or 13.

The virgule symbol (/) means either/or; for example, a/b means either a or b.

The number of clocks when execution is branched by a conditional branch instruction is shown after the symbol (/).

The number of clocks for instruction having the saddr or saddrp operand and when an SFR is accessed with FF00H to FFFFH described as saddr or saddrp is shown after the symbol (/).

Bytes and Clocks

The number of bytes and clocks for instructions with a mem or &mem operand depends on the particular instruction and the memory addressing mode (register indirect, base, or indexed). Table 9 is applicable when the program in internal ROM is executed (ROM clock column of the Instruction Set table). Table 10 is applicable when the program in external ROM is executed (IRAM, SFR, and EMEM clock columns).

Flags

The symbols in the flag field have the following meanings.

Blank	No change
0	Cleared to 0
1	Set to 1
x	Set or cleared depending on the result
R	Value previously saved is restored

Operation Codes

Table 11 defines the symbols used in the operation code field.

Registers and Register Pairs. The r, rl, and rp operands are specified in the opcode by one or more bits as shown in figure 29. For example, 001 as bits $R_2R_1R_0$ (or $R_6R_5R_4$) specifies register A.

In the first and second operands are registers or register pairs; the higher 4 bits of the register specification byte define the first operand and the lower 4 bits define the second operand. For example, in the MOV A,L instruction (transfer L register contents to register A), the second byte of the opcode is obtained from figure 29 as shown below.

Instruction	Opcode, Bytes 1 and 2
MOV r,r	0 0 1 0 0 1 0 0
	0 $R_6R_5R_4$ 0 $R_2R_1R_0$
MOV A,L	0 0 1 0 0 1 0 0
	0 0 0 1 0 1 1 0

Memory Addressing Modes. The 3-bit mem code and the 5-bit mod code are selected from figure 30 according to the description of mem in the operand field (table 7).

A MOV instruction with register indirect mode specified for mem is a special 1-byte instruction. When base mode or indexed mode is specified for mem, the 8-bit or 16-bit offset data corresponding to byte and word, respectively, is added from the third byte onward.

The opcode for an &mem or &mem1 operand is modified by inserting a 01H code as the first byte preceding the first-byte code listed in the Instruction Set table. Subsequent bytes are as shown in the table.

Figure 29. Opcodes for Registers (r, r1, rp)

r

R_2	R_1	R_0	reg	
R_6	R_5	R_4		
0	0	0	R0	X
0	0	1	R1	A
0	1	0	R2	C
0	1	1	R3	B
1	0	0	R4	E
1	0	1	R5	D
1	1	0	R6	L
1	1	1	R7	H

r1

R_0	reg
0	C
1	B

rp

P_1	P_0	reg-pair	
P_2	P_1		
P_6	P_5		
0	0	RP0	AX
0	1	RP1	BC
1	0	RP2	DE
1	1	RP3	HL

83ML-5998A

Figure 30. Opcodes for Memory Addressing Modes (mem, mod)

Mod / Mem	1 0110 Register Indirect Mode	0 0110 Base Mode	0 1010 Index Mode
0 0 0	[DE+]	[DE+byte]	word [DE]
0 0 1	[HL+]	[SP+byte]	word [A]
0 1 0	[DE-]	[HL+byte]	word [HL]
0 1 1	[HL-]	-	word [B]
1 0 0	[DE]	-	-
1 0 1	[HL]	-	-

83ML-5999A

Table 9. Bytes and Clocks for Instructions With "mem" and "&mem" Operands; Internal ROM (IROM)

			Register Indirect Mode		Base Mode		Indexed Mode
	Instruction		[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE+byte] [HL+byte]	[SP+byte]	word[A] word[B] word[DE] word[HL]
Bytes		mem	1/2*	1/2*	3	3	4
		&mem	2/3*	2/3*	4	4	5
Clock Cycles	MOV	A, mem	6/8	6/8	8-11	9-12	8-11
		mem, A					
		A, &mem	8/10	8/10	10-13	11-14	10-13
		&mem, A					
	XCH	A, mem	11-15	9-13	10-15	11-16	10-15
		A, &mem	13-17	11-15	12-17	13-18	12-17
	ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	10/12	8/12	9/12	10-13	9-12
		A, &mem	12/14	10/14	11/14	12-15	11-14

* When internal RAM is accessed with an instruction having a mem operand, the number of bytes is the number before the symbol (/). When the external memory (including the SFR area) is accessed, the number of bytes is the number after the symbol (/).

Table 10. Bytes and Clocks for Instructions With "mem" and "&mem" Operands; External ROM (IRAM, SFR, EMEM)

			Register Indirect Mode		Base Mode		Indexed Mode
Instruction			[DE+] [HL+] [DE–] [HL–]	[DE] [HL]	[DE + byte] [HL + byte]	[SP + byte]	word[A] word[B] word[DE] word[HL]
Bytes		mem	2*	2*	3	3	4
		&mem	3*	3*	4	4	5
Clock Cycles	MOV	A, mem	9/11	6/8	11/13	12/14	14/16
		mem, A					
		A, &mem	12/14	9/11	14/16	15/17	17/19
		&mem, A					
	XCH	A, mem	14/18	12/16	13/17	14/18	16/20
		A, &mem	17/21	15/19	16/20	17/21	19/23
	ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP	A, mem	13/15	11/13	12/14	13/15	15/17
		A, &mem	16/18	14/16	15/17	16/18	18/20

* When [DE], [HL], [DE+], [HL+], [DE–], or [HL–] is specified as the mem operand of a MOV instruction, the instruction is used as a dedicated 1-byte type. When the operand is &mem, the instruction is 2-byte.

Table 11. Opcode Symbols

Symbol	Meaning
Bn	Immediate data corresponding to bit
Nn	Immediate data corresponding to n
Data	8-bit immediate data corresponding to byte
Low/High Byte	16-bit immediate data corresponding to word
Saddr-offset	Lower 8-bit offset data of 16-bit address corresponding to saddr
Sfr-offset	Lower 8-bit offset data of 16-bit address of special function register (sfr)
Low/High Offset	16-bit offset data corresponding to word in indexed addressing
Low/High Addr	16-bit immediate data corresponding to addr16
jdisp	Signed 2's complement data (8 bits) indicating relative address distance between first address of next instruction and branch destination address
fa	Lower 11 bits of immediate data corresponding to addr11
ta	Lower 5 bits of immediate data corresponding to (addr5xdis)

Instruction Set

				Clocks				Flags	
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z AC CY**	**Operation Code (Bits 7-0) Bytes B1 thru B5**
8-Bit Data Transfer									
MOV	r,#byte	r ← byte	2	2	6				1 0 1 1 1 R_2 R_1 R_0
									Data
	saddr,#byte	(saddr) ← byte	3	3/5	9	9	12		0 0 1 1 1 0 1 0
									Saddr-offset
									Data
	sfr,#byte	sfr ← byte	3	5		9	12		0 0 1 0 1 0 1 1
									Sfr-offset
									Data
	r,r	r ← r	2	2	6				0 0 1 0 0 1 0 0
									0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,r	A ← r	1	2	3				1 1 0 1 0 R_2 R_1 R_0
	A,saddr	A ← (saddr)	2	2/4	6	6	9		0 0 1 0 0 0 0 0
									Saddr-offset
	saddr,A	(saddr) ← A	2	3/5	6	8			0 0 1 0 0 0 1 0
									Saddr-offset
	saddr, saddr	(saddr) ← (saddr)	3	3-7	9				0 0 1 1 1 0 0 0
									Saddr-offset
									Saddr-offset
	A,sfr	A ← sfr	2	4		6			0 0 0 1 0 0 0 0
									Sfr-offset
	sfr,A	sfr ← A	2	5		6			0 0 0 1 0 0 1 0
									Sfr-offset
	A,mem	A ← (mem)	1-4	6-12	6-14	8-16	8-16		* 0 1 0 1 1 mem
									0 0 0 mod
									0 mem 0 0 0 0
									Low Offset
									High Offset
	A,&mem	A ← (&mem)	2-5	8-14	9-17	11-19	11-19		* 0 0 0 0 0 0 0 1
									0 1 0 1 1 mem
									0 0 0 0 0 0 0 1
									0 0 0 mod
									0 mem 0 0 0 0
									Low Offset
									High Offset

Note:

* If [DE], [HL], [DE+], [DE–], [HL+] or [HL–] is described as mem, these instructions are used as dedicated 1-byte codes. If the register name is described as &mem, the instructions are used as dedicated 2-byte codes.

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Data Transfer (cont)											
MOV	mem,A	(mem) ← A	1-4	6-12	6-14	8-16	8-16			*	0 1 0 1 0 mem 0 0 0 mod 1 mem 0 0 0 0 Low Offset High Offset
	&mem,A	(&mem) ← A	2-5	8-14	9-17	11-19	11-19			*	0 0 0 0 0 0 0 1 0 1 0 1 0 mem 0 0 0 0 0 0 0 1 0 0 0 mod 1 mem 0 0 0 0 Low Offset High Offset
	A,!addr16	A ← (!addr16)	4	6/8	14		16				0 0 0 0 1 0 0 1 1 1 1 1 0 0 0 0 Low Addr High Addr
	A,&!addr16	A ← (&laddr16)	5	8/10			19				0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 1 1 1 1 1 0 0 0 0 Low Addr High Addr
	!addr16, A	(!addr16) ← A	4	6/8	14		17				0 0 0 0 1 0 0 1 1 1 1 1 0 0 0 1 Low Addr High Addr
	&!addr16,A	(&laddr16) ← A	5	8/10			20				0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 1 1 1 1 1 0 0 0 1 Low Addr High Addr
	PSW,#byte	PSW ← byte	3	3	9	9	9	x	x	x	0 0 1 0 1 0 1 1 1 1 1 1 1 1 1 0 Data
	PSW,A	PSW ← A	2	2	6	6	6	x	x	x	0 0 0 1 0 0 1 0 1 1 1 1 1 1 1 0
	A,PSW	A ← PSW	2	2	6	6	6				0 0 0 1 0 0 0 0 1 1 1 1 1 1 1 0

6

Instruction Set (cont)

				Clocks				Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z AC CY	Bytes B1 thru B5
8-Bit Data Transfer (cont)									
XCH	A,r	A ⟷ r	1	4	4				1 1 0 1 1 R_2 R_1 R_0
	r,r	r ⟷ r	2	3	6				0 0 1 0 0 1 0 1
									0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,mem	A ⟷ (mem)	2-4	9-16	12-16		16-20		0 0 0 mod
									0 mem 0 1 0 0
									Low Offset
									High Offset
	A,&mem	A ⟷ (&mem)	3-5	11-18	15-19		19-23		0 0 0 0 0 0 0 1
									0 0 0 mod
									0 mem 0 1 0 0
									Low Offset
									High Offset
	A,saddr	A ⟷ (saddr)	2	4/8	6				0 0 1 0 0 0 0 1
									Saddr-offset
	A,sfr	A ⟷ sfr	3	6/10		13			0 0 0 0 0 0 0 1
									0 0 1 0 0 0 0 1
									Sfr-offset
	saddr,saddr	(saddr) ⟷ (saddr)	3	6-14		10			0 0 1 1 1 0 0 1
									Saddr-offset
									Saddr-offset
16-Bit Data Transfer									
MOVW	rp,#word	rp ← word	3	3	9				0 1 1 0 0 P_2 P_1 0
									Low Byte
									High Byte
	saddrp,#word	(saddrp) ← word	4	4/8	12	12	18		0 0 0 0 1 1 0 0
									Saddr-offset
									Low Byte
									High Byte
	sfrp,#word	sfrp ← word	4	8		12			0 0 0 0 1 0 1 1
									Saddr-offset
									Low Byte
									High Byte
	rp,rp	rp ← rp	2	4	6				0 0 1 0 0 1 0 0
									0 P_6 P_5 0 1 P_2 P_1 0
	AX,saddrp	AX ← (saddrp)	2	6/10	8	12			0 0 0 1 1 1 0 0
									Saddr-offset
	saddrp,AX	(saddrp) ← AX	2	5/9	8	12			0 0 0 1 1 0 1 0
									Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
16-Bit Data Transfer (cont)											
MOVW	AX,sfrp	AX ← sfrp	2	10		12					0 0 0 1 0 0 0 1
											Sfr-offset
	sfrp,AX	sfrp ← AX	2	9		12					0 0 0 1 0 0 1 1
											Sfr-offset
	AX,mem1	AX ← (mem1)	2	9-15	12	16	16				0 0 0 0 0 1 0 1
											1 1 1 0 0 0 1 R_0
	AX,&mem1	AX ← (&mem1)	3	11-17	15	19	19				0 0 0 0 0 0 0 1
											0 0 0 0 0 1 0 1
											1 1 1 0 0 0 1 R_0
	mem1, AX	(mem1) ← AX	2	8-14	11	15	15				0 0 0 0 0 1 0 1
											1 1 1 0 0 1 1 R_0
	&mem1,AX	(&mem1) ← AX	3	10-16	14	18	18				0 0 0 0 0 0 0 1
											0 0 0 0 0 1 0 1
											1 1 1 0 0 1 1 R_0
8-Bit Operation											
ADD	A,#byte	A,CY ← A + byte	2	2	6			x	x	x	1 0 1 0 1 0 0 0
											Data
	saddr,#byte	(saddr),CY ← (saddr) + byte	3	3/7	9	11		x	x	x	0 1 1 0 1 0 0 0
											Saddr-offset
											Data
	sfr,#byte	sfr,CY ← sfr + byte	4	9		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 0 0 0
											Sfr-offset
											Data
	r,r	r,CY ← r + r	2	3	7			x	x	x	1 0 0 0 1 0 0 0
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A + (saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 0 0
											Saddr-offset
	A,sfr	A,CY ← A + sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 0 0 0
											Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) + (saddr)	3	3-9	9	11		x	x	x	0 1 1 1 1 0 0 0
											Saddr-offset
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
ADD	A,mem	A,CY ← A + (mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 0 0 Low Offset High Offset
	A,&mem	A,CY ← A + (&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 0 0 Low Offset High Offset
ADDC	A,#byte	A,CY ← A + byte + CY	2	2	6			x	x	x	1 0 1 0 1 0 0 1 Data
	saddr,#byte	(saddr),CY ← (saddr) + byte + CY	3	3/7	9	11		x	x	x	0 1 1 0 1 0 0 1 Saddr-offset Data
	sfr,#byte	sfr,CY ← sfr + byte + CY	4	9		14		x	x	x	0 0 0 0 0 0 0 1 0 1 1 0 1 0 0 1 Sfr-offset Data
	r,r	r,CY ← r + r + CY	2	3	7			x	x	x	1 0 0 1 1 0 0 1 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A + (saddr) + CY	2	2/5	6	7	8	x	x	x	1 0 0 1 1 0 0 1 Saddr-offset
	A,sfr	A,CY ← A + sfr + CY	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 1 Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) + (saddr) + CY	3	3-9	9	11		x	x	x	0 1 1 1 1 0 0 1 Saddr-offset Saddr-offset
	A,mem	A,CY ← A + (mem) + CY	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 0 1 Low Offset High Offset
	A,&mem	A,CY ← A + (&mem) + CY	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 0 0 Low Offset High Offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
SUB	A,#byte	A,CY ← A−byte	2	2	6			x	x	x	1 0 1 0 1 0 1 0
											Data
	saddr,#byte	(saddr),CY ← (saddr)−(byte)	3	3/7	9	11		x	x	x	0 1 1 0 1 0 1 0
											Saddr-offset
											Data
	sfr,#byte	sfr,CY ← sfr−byte	4	9		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 0 1 0
											Sfr-offset
											Data
	r,r	r,CY ← r−r	2	3	7			x	x	x	1 0 0 0 1 0 1 0
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A−(saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 1 0
											Saddr-offset
	A,sfr	A,CY ← A−sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 0 1 0
											Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) − (saddr)	3	3-9	9	11		x	x	x	0 1 1 1 1 0 1 0
											Saddr-offset
											Saddr-offset
	A,mem	A,CY ← A−(&mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod
											0 mem 1 0 1 0
											Low Offset
											High Offset
	A,&mem	A,CY ← A−(&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 0 1 0
											Low Offset
											High Offset
SUBC	A,#byte	A,CY ← A−byte−CY	2	2	6			x	x	x	1 0 1 0 1 0 1 1
											Data
	saddr,#byte	(saddr),CY ← (saddr)−byte−CY	3	3/7	9	11		x	x	x	0 1 1 0 1 0 1 1
											Saddr-offset
											Data
	sfr,#byte	sfr,CY ← sfr−byte−CY	4	9		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 0 1 1
											Sfr-offset
											Data

6

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
SUBC	r,r	r,CY ← r − r − CY	2	3	7			x	x	x	1 0 0 0 1 0 1 1 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A,CY ← A − (saddr) − CY	2	3/5	6	7	8	x	x	x	1 0 0 1 1 0 1 1 Saddr-offset
	A,sfr	A,CY ← A − sfr − CY	3	7		10		x	x	x	0 0 0 0 0 0 0 1 1 0 0 1 1 0 1 1 Sfr-offset
	saddr,saddr	(saddr),CY ← (saddr) − (saddr) − CY	3	3-9	9	11		x	x	x	0 1 1 1 1 0 1 1 Saddr-offset Saddr-offset
	A,mem	A,CY ← A − (mem) − CY	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod 0 mem 1 0 1 1 Low Offset High Offset
	A,&mem	A,CY ← A − (&mem) − CY	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 0 1 1 Low Offset High Offset
AND	A,#byte	A ← A ∧ byte	2	2	6			x			1 0 1 0 1 1 0 0 Data
	saddr,#byte	(saddr) ← (saddr) ∧ byte	3	3/7	9	11		x			0 1 1 0 1 1 0 0 Saddr-offset Data
	sfr,#byte	sfr ← sfr ∧ byte	4	9		14		x			0 0 0 0 0 0 0 1 0 1 1 0 1 1 0 0 Sfr-offset Data
	r,r	r ← r ∧ r	2	3	7			x			1 0 0 0 1 1 0 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A ∧ (saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 0 0 Saddr-offset
	A,sfr	A ← A ∧ (sfr)	3	7		10		x			0 0 0 0 0 0 0 1 1 0 0 1 1 1 0 0 Sfr-offset
	saddr,saddr	(saddr) ← (saddr) ∧ (saddr)	3	3-9	9	11		x			0 1 1 1 1 1 0 0 Saddr-offset Saddr-offset

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	Clocks IROM	IRAM	SFR	EMEM	Flags Z	AC	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
8-Bit Operation (cont)											
AND	A,mem	A ← A∧(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod 0 mem 1 1 0 0 Low Offset High Offset
	A,&mem	A ← A∧(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 0 0 Low Offset High Offset
OR	A,#byte	A ← AV byte	2	2	6			x			1 0 1 0 1 1 1 0 Data
	saddr,#byte	(saddr) ← (saddr) V byte	3	3/7	9	11		x			0 1 1 0 1 1 1 0 Saddr-offset Data
	sfr,#byte	sfr ← sfr V byte	4	9		14		x			0 0 0 0 0 0 0 1 0 1 1 0 1 1 1 0 Sfr-offset Data
	r,r	r ← rVr	2	3	7			x			1 0 0 0 1 1 1 0 0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← AV(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 1 0 Saddr-offset
	A,sfr	A ← AV sfr	3	7		10		x			0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 0 Sfr-offset
	saddr,saddr	(saddr) ← (saddr)V (saddr)	3	3-9	9	11		x			0 1 1 1 1 1 1 0 Saddr-offset Saddr-offset
	A,mem	A ← AV(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod 0 mem 1 1 1 0 Low Offset High Offset
	A,&mem	A ← AV(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1 0 0 0 mod 0 mem 1 1 1 0 Low Offset High Offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
8-Bit Operation (cont)											
XOR	A,#byte	A ← A∀byte	2	2	6			x			1 0 1 0 1 1 0 1
											Data
	saddr,#byte	(saddr) ← (saddr)∀byte	3	3/5	9	11		x			0 1 1 0 1 1 0 1
											Saddr-offset
											Data
	sfr,#byte	sfr ← sfr∀byte	4	7		14		x			0 0 0 0 0 0 0 1
											0 1 1 0 1 1 0 1
											Sfr-offset
											Data
	r,r	r ← r∀r	2	3	7			x			1 0 0 0 1 1 0 1
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A ← A∀(saddr)	2	3/5	6	7	8	x			1 0 0 1 1 1 0 1
											Saddr-offset
	A,sfr	A ← A∀(sfr)	3	7		10		x			0 0 0 0 0 0 0 1
											1 0 0 1 1 1 0 1
											Sfr-offset
	saddr,saddr	(saddr) ← (saddr)∀(saddr)	3	3-9	9	11		x			0 1 1 1 1 1 0 1
											Saddr-offset
											Saddr-offset
	A,mem	A ← A∀(mem)	2-4	8-13	11-15	13-17	13-17	x			0 0 0 mod
											0 mem 1 1 0 1
											Low Offset
											High Offset
	A,&mem	A ← A∀(&mem)	3-5	10-15	14-18	16-20	16-20	x			0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 1 0 1
											Low Offset
											High Offset
CMP	A,#byte	A−byte	2	2	6			x	x	x	1 0 1 0 1 1 1 1
											Data
	saddr,#byte	(saddr)−byte	3	3/5	9	11		x	x	x	0 1 1 0 1 1 1 1
											Saddr-offset
											Data
	sfr,#byte	sfr−byte	4	7		14		x	x	x	0 0 0 0 0 0 0 1
											0 1 1 0 1 1 1 1
											Sfr-offset
											Data

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Bytes B1 thru B5**
8-Bit Operation (cont)											
CMP	r,r	r−r	2	3	7			x	x	x	1 0 0 0 1 1 1 1
											0 R_6 R_5 R_4 0 R_2 R_1 R_0
	A,saddr	A−(saddr)	2	3/5	6	7	8	x	x	x	1 0 0 1 1 1 1 1
											Saddr-offset
	A,sfr	A−sfr	3	7		10		x	x	x	0 0 0 0 0 0 0 1
											1 0 0 1 1 1 1 1
											Sfr-offset
	saddr,saddr	(saddr)−(saddr)	3	3-7	9	11		x	x	x	0 1 1 1 1 1 1 1
											Saddr-offset
											Saddr-offset
	A,mem	A−(mem)	2-4	8-13	11-15	13-17	13-17	x	x	x	0 0 0 mod
											0 mem 1 1 1 1
											Low Offset
											High Offset
	A,&mem	A−(&mem)	3-5	10-15	14-18	16-20	16-20	x	x	x	0 0 0 0 0 0 0 1
											0 0 0 mod
											0 mem 1 1 1 1
											Low Offset
											High Offset
16-Bit Operation											
ADDW	AX,#word	AX,CY ← AX + word	3	4	9			x	x	x	0 0 1 0 1 1 0 1
											Low Byte
											High Byte
	AX,rp	AX,CY ← AX + rp	2	6	8			x	x	x	1 0 0 0 1 0 0 0
											0 0 0 0 1 P_2 P_1 0
	AX,saddrp	AX,CY ← AX + (saddrp)	2	7/11	9	13		x	x	x	0 0 0 1 1 1 0 1
											Saddr-offset
	AX,sfrp	AX,CY ← AX + sfrp	3	13		16		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 0 1
											Sfr-offset
SUBW	AX,#word	AX,CY ← AX−word	3	4	9			x	x	x	0 0 1 0 1 1 1 0
											Low Byte
											High Byte
	AX,rp	AX,CY ← AX−rp	2	6	8			x	x	x	1 0 0 0 1 0 1 0
											0 0 0 0 1 P_2 P_1 P_0
	AX,saddrp	AX,CY ← AX−(saddrp)	2	7/11	9	13		x	x	x	0 0 0 1 1 1 1 0
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
16-Bit Operation (cont)											
SUBW	AX,sfrp	AX,CY ← AX−sfrp	3	13		16		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 1 0
											Sfr-offset
CMPW	AX,#word	AX−word	3	3	9			x	x	x	0 0 1 0 1 1 1 1
											Low Byte
											High Byte
	AX,rp	AX−rp	2	5	7			x	x	x	1 0 0 0 1 1 1 1
											0 0 0 0 1 P_2 P_1 0
	AX,saddrp	AX−(saddrp)	2	6/10	8	12		x	x	x	0 0 0 1 1 1 1 1
											Saddr-offset
	AX,sfrp	AX−sfrp	3	12		15		x	x	x	0 0 0 0 0 0 0 1
											0 0 0 1 1 1 1 1
											Sfr-offset
Multiplication/Division											
MULU	r	AX ← Axr	2	22	24						0 0 0 0 0 1 0 1
											0 0 0 0 1 R_2 R_1 R_0
DIVUW	r	AX(quotient), r (remainder) ← AX ÷ r	2	71	76						0 0 0 0 0 1 0 1
											0 0 0 1 1 R_2 R_1 R_0
Increment/Decrement											
INC	r	r ← r + 1	1	2	3			x	x		1 1 0 0 0 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) + 1	2	2/6	6	7		x	x		0 0 1 0 0 1 1 0
											Saddr-offset
DEC	r	r ← r − 1	1	2	3			x	x		1 1 0 0 1 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) − 1	2	2/6	6	7		x	x		0 0 1 0 0 1 1 1
											Saddr-offset
INCW	rp	rp ← rp + 1	1	3	3						0 1 0 0 0 1 P_1 P_0
DECW	rp	rp ← rp − 1	1	3	3						0 1 0 0 1 1 P_1 P_0
Shift/Rotate											
ROR	r,n	(CY,r_7 ← r_0, r_{m-1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 0
											0 1 N_2 N_1 N_0 R_2 R_1 R_0
ROL	r,n	(CY,r_0 ← r_7, r_{m+1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 1
											0 1 N_2 N_1 N_0 R_2 R_1 R_0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Shift/Rotate (cont)											
RORC	r,n	(CY ← r_0, r_7 ← CY, r_{m-1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 0 0 0 N_2 N_1 N_0 R_2 R_1 R_0
ROLC	r,n	(CY ← r_7, r_0 ← CY, r_{m+1} ← r_m) xn times, n=0-7	2	3+2n	5+2n					x	0 0 1 1 0 0 0 1 0 0 N_2 N_1 N_0 R_2 R_1 R_0
SHR	r,n	(CY ← r_0, r_7 ← 0, r_{m-1} ← r_m) xn times, n=0-7	2	3+2n	5+2n			x	0	x	0 0 1 1 0 0 0 0 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHL	r,n	(CY ← r_7, r_0 ← 0, r_{m+1} ← r_m) xn times, n=0-7	2	3+2n	5+2n			x	0	x	0 0 1 1 0 0 0 1 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHRW	rp,n	(CY ← rp_0, rp_{15} ← 0, rp_{m-1} ← rp_m) xn times, n=0-7	2	3+3n	5+3n			x	0	x	0 0 1 1 0 0 0 0 1 1 N_2 N_1 N_0 R_2 R_1 R_0
SHLW	rp,n	(CY ← rp_{15}, rp_0 ← 0, rp_{m+1} ← rp_m) xn times, n=0-7	2	3+3n	5+3n			x	0	x	0 0 1 1 0 0 0 1 1 1 N_2 N_1 N_0 R_2 R_1 R_0
ROR4	mem1	A_{3-0} ← $(mem1)_{3-0}$, $(mem1)_{7-4}$ ← A_{3-0}, $(mem1)_{3-0}$ ← $(mem1)_{7-4}$	2	24	26	34	34				0 0 0 0 0 1 0 1 1 0 0 0 1 1 R_1 0
	&mem1	A_{3-0} ← $(\&mem1)_{3-0}$, $(\&mem1)_{7-4}$ ← A_{3-0}, $(\&mem1)_{3-0}$ ← $(\&mem1)_{7-4}$	3	26	29	37	37				0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 0 1 1 R_1 0
ROL4	mem1	A_{3-0} ← $(mem1)_{7-4}$, $(mem1)_{3-0}$ ← A_{3-0}, $(mem1)_{7-4}$ ← $(mem1)_{3-0}$	2	25	27	35	35				0 0 0 0 0 1 0 1 1 0 0 1 1 1 R_1 0
	&mem1	A_{3-0} ← $(\&mem1)_{7-4}$, $(\&mem1)_{3-0}$ ← A_{3-0}, $(\&mem1)_{7-4}$ ← $(\&mem1)_{3-0}$	3	27	30	38	38				0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 0 1 1 1 R_1 0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
BCD Adjustment											
ADJBA		Decimal adjust accumulator after addition	1	3		3		x	x	x	0 0 0 0 1 1 1 0
ADJBS		Decimal adjust accumulator after addition	–1	3		3		x	x	x	0 0 0 0 1 1 1 1
Bit Manipulation											
MOV1	CY,saddr.bit	CY ← (saddr bit)	3	5/7	9	9	11			x	0 0 0 0 1 0 0 0
											0 0 0 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,sfr.bit	CY ← sfr.bit	3	7		9				x	0 0 0 0 1 0 0 0
											0 0 0 0 1 B_2 B_1 B_0
											Sfr-offset
	CY,A.bit	CY ← A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 0 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 0 0 0 B_2 B_1 B_0
											Sfr-offset
	CY,PSW.bit	CY ← PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 0 0 0 0 B_2 B_1 B_0
	saddr.bit,CY	(saddr bit) ← CY	3	8/12	12	14	14				0 0 0 0 1 0 0 0
											0 0 0 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit,CY	sfr.bit ← CY	3	12		14					0 0 0 0 1 0 0 0
											0 0 0 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit,CY	A.bit ← CY	2	8	10						0 0 0 0 0 0 1 1
											0 0 0 1 1 B_2 B_1 B_0
	X.bit,CY	X.bit ← CY	2	8	10						0 0 0 0 0 0 1 1
											0 0 0 1 0 B_2 B_1 B_0
											Sfr-offset
	PSW.bit,CY	PSW.bit ← CY	2	7		9		x	x		0 0 0 0 0 0 1 0
											0 0 0 1 0 B_2 B_1 B_0
AND1	CY,saddr.bit	CY ← CY ∧ (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 0 1 0 0 B_2 B_1 B_0
											Saddr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Bit Manipulation (cont)											
AND1	CY,/saddr.bit	CY ← CY ∧ $\overline{\text{(saddr.bit)}}$	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 0 1 1 0 B_2 B_1 B_0
											Sfr-offset
	CY,sfr.bit	CY ← CY ∧ sfr.bit	3	7		11				x	0 0 0 0 1 0 0 0
											0 0 1 0 1 B_2 B_1 B_0
											Sfr-offset
	CY,/sfr.bit	CY ← CY ∧ $\overline{\text{sfr.bit}}$	3	7		11				x	0 0 0 0 1 0 0 0
											0 0 1 1 1 B_2 B_1 B_0
	CY,A.bit	CY ← CY ∧ A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 1 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY ∧ $\overline{\text{A.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 1 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY ∧ X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 1 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY ∧ $\overline{\text{X.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 0 1 1 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY ∧ PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 0 1 0 0 B_2 B_1 B_0
	CY,/PSW.bit	CY ← CY ∧ $\overline{\text{PSW.bit}}$	2	5		7				x	0 0 0 0 0 0 1 0
											0 0 1 1 0 B_2 B_1 B_0
OR1	CY,saddr.bit	CY ← CY ∨ (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 1 0 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,/saddr.bit	CY ← CY ∨ $\overline{\text{(saddr.bit)}}$	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 1 0 1 0 B_2 B_1 B_0
											Sfr-offset
	CY,sfr.bit	CY ← CY ∨ sfr.bit)	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 0 1 1 B_2 B_1 B_0
											Sfr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**Bytes**	**IROM**	**IRAM**	**SFR**	**EMEM**	**Z**	**AC**	**CY**	**Bytes B1 thru B5**
Bit Manipulation (cont)											
OR1	CY/sfr.bit	CY ← CY V $\overline{\text{sfr.bit}}$	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 0 1 1 B_2 B_1 B_0
	CY,A.bit	CY ← CY V A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY V $\overline{\text{A.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY V X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY V $\overline{\text{X.bit}}$	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 0 1 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY V PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 0 0 0 B_2 B_1 B_0
	CY,/PSW.bit	CY ← CY V $\overline{\text{PSW.bit}}$	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 0 1 0 B_2 B_1 B_0
XOR1	CY,saddr.bit	CY ← CY $\veebar$ (saddr.bit)	3	5/7	9	11				x	0 0 0 0 1 0 0 0
											0 1 1 0 0 B_2 B_1 B_0
											Saddr-offset
	CY,sfr.bit	CY ← CY $\veebar$ sfr.bit	3	7		11				x	0 0 0 0 1 0 0 0
											0 1 1 0 1 B_2 B_1 B_0
											Sfr-ofset
	CY,A.bit	CY ← CY $\veebar$ A.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 1 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY $\veebar$ X.bit	2	5	7					x	0 0 0 0 0 0 1 1
											0 1 1 0 0 B_2 B_1 B_0
	CY,PSW.bit	CY ← CY $\veebar$ PSW.bit	2	5		7				x	0 0 0 0 0 0 1 0
											0 1 1 0 0 B_2 B_1 B_0
SET1	saddr.bit	(saddr.bit) ← 1	2	3/7	6						1 0 1 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← 1	3	10		14					0 0 0 0 1 0 0 0
											1 0 0 0 1 B_2 B_1 B_0
											Sfr-offset

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Bit Manipulation (cont)											
SET1	A.bit	A.bit ← 1	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 0 1 B_2 B_1 B_0
	X.bit	X.bit ← 1	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 0 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← 1	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											1 0 0 0 0 B_2 B_1 B_0
CLR1	saddr.bit	(saddr.bit) ← 0	2	6/10	6						1 0 1 0 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← 0	3	10		14					0 0 0 0 1 0 0 0
											1 0 0 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit	A.bit ← 0	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 1 1 B_2 B_1 B_0
	X.bit	X.bit ← 0	2	6	8						0 0 0 0 0 0 1 1
											1 0 0 1 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← 0	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											1 0 0 1 0 B_2 B_1 B_0
NOT1	saddr.bit	(saddr.bit) ← 0 ($\overline{\text{saddr.bit}}$)	3	6/10	10	14					0 0 0 0 1 0 0 0
											0 1 1 1 0 B_2 B_1 B_0
											Saddr-offset
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3	10		14					0 0 0 0 1 0 0 0
											0 1 1 1 1 B_2 B_1 B_0
											Sfr-offset
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2	6	8						0 0 0 0 0 0 1 1
											0 1 1 1 1 B_2 B_1 B_0
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2	6	8						0 0 0 0 0 0 1 1
											0 1 1 1 0 B_2 B_1 B_0
	PSW.bit	PSW.bit ← $\overline{\text{PSW.bit}}$	2	5		7		x	x	x	0 0 0 0 0 0 1 0
											0 1 1 1 0 B_2 B_1 B_0
SET1	CY	CY ← 1	1	2		3				1	0 1 0 0 0 0 0 1
CLR1	CY	CY ← 0	1	2		3				0	0 1 0 0 0 0 0 0
NOT1	CY	CY ← $\overline{\text{CY}}$	1	2		3				x	0 1 0 0 0 0 1 0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Call/Return											
CALL	!addr16	(SP−1) ← $(PC+3)_H$, (SP−2) ← $(PC+3)_L$, PC ← !addr16, SP ← SP−2	3	10-15	17		21				0 0 1 0 1 0 0 0 Low Addr High Addr
	rp	(SP−1) ← $(PC+2)_H$, (SP−2) ← $(PC+2)_L$, PC_H ← r_{PH}, PC_L ← rp_L, SP ← SP−2	2	12-17	15		19				0 0 0 0 0 1 0 1 0 1 0 1 1 P_2 P_1 0
CALLF	!addr11	(SP−1) ← $(PC+2)_H$, (SP−2) ← $(PC+2)_L$, PC_{15+11} ← 00001, PC_{10-0} ← !addr11, SP ← SP−2	2	10-15	14		18				1 0 0 1 0 ← fa →
CALLT	[addr5]	(SP−1) ← $(PC+1)_H$, (SP−2) ← $(PC+1)_L$, PC_H ← (00000000, addr5 + 1), PC_L ← (00000000, addr5), SP ← SP−2	1	14-20	20		24				1 1 1 ← ta →
BRK		(SP−1) ← PSW, (SP−2) ← $(PC+1)_H$, (SP−3) ← $(PC+1)_L$, PC_H ← (003FH), PC_H ← (003FH), SP ← SP−3, IE ← 0	1	16-26	22		28				0 1 0 1 1 1 1 0
RET		PC_L ← (SP), PC_H ← (SP + 1), SP ← SP + 2	1	10-15	11		15				0 1 0 1 0 1 1 0
RETI		PC_L ← (SP), PC_H ← (SP + 1), PSW ← (SP + 2), SP ← SP + 3, NMIS ← 0	1	12-20	15		21	R	R	R	0 1 0 1 0 1 1 1
RETB		PC_L ← (SP), PC_H ← (SP + 1), PSW ← (SP + 2), SP ← SP + 3	1	12-20	13		19	R	R	R	0 1 0 1 1 1 1 1
Stack Manipulation											
PUSH	PSW	(SP−1) ← PSW, SP ← SP−1	1	4-8	5		7				0 1 0 0 1 0 0 1
	sfr	(SP−1) ← sfr, SP ← SP−1	2	7-9	9		12				0 0 1 0 1 0 0 1 Sfr-offset
	rp	(SP−1) ← rp_H (SP−2) ← rp_L, SP ← SP−2	1	8-13	8		12				0 0 1 1 1 1 P_1 P_0

Instruction Set (cont)

				Clocks				Flags			Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	IROM	IRAM	SFR	EMEM	Z	AC	CY	Bytes B1 thru B5
Stack Manipulation (cont)											
POP	PSW	PSW ← (SP), SP ← SP + 1	1	4-8	6		8	R	R	R	0 1 0 0 1 0 0 0
	sfr	sfr ← (SP), SP ← SP + 1	2	9-11	9		12				0 1 0 0 0 0 1 1
											Sfr-offset
	rp	rp_L ← (SP), rp_H ← (SP + 1), SP ← SP +2	1	10-15	11		15				0 0 1 1 0 1 P_1 P_0
MOVW	SP,#word	SP ← word	4	8		12					0 0 0 0 1 0 1 1
											1 1 1 1 1 1 0 0
											Low Byte
											High Byte
	SP,AX	SP ← AX	2	9		11					0 0 0 1 0 0 1 1
											1 1 1 1 1 1 0 0
	AX,SP	AX ← SP	2	10		12					0 0 0 1 0 0 0 1
											1 1 1 1 1 1 0 0
INCW	SP	SP ← SP + 1	2	5		7					0 0 0 0 0 1 0 1
											1 1 0 0 1 0 0 0
DECW	SP	SP ← SP − 1	2	5		7					0 0 0 0 0 1 0 1
											1 1 0 0 1 0 0 1

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z AC CY	Bytes B1 thru B5
Unconditional Branch								
BR	!addr16	PC ← !addr16	3	5	11			0 0 1 0 1 1 0 0
								Low Addr
								High Addr
	rp	PC_H ← rp_H, PC_L ← rp_L	2	6	10			0 0 0 0 0 1 0 1
								0 1 0 0 1 P_2 P_1 0
	$addr16	PC ← $addr16	2	4	9			0 0 0 1 0 1 0 0
								jdisp
Conditional Branch								
BC	$addr16	PC ← $addr16 if CY = 1	2	2/4	9	6		1 0 0 0 0 0 1 1
BL								jdisp
BNC	$addr16	PC ← $addr16 if CY = 0	2	2/4	9	6		1 0 0 0 0 0 1 0
BNL								jdisp
BZ	$addr16	PC ← $addr16 if Z = 1	2	2/4	9	6		1 0 0 0 0 0 0 1
BE								jdisp
BNZ	$addr16	PC ← $addr16 if Z = 0	2	2/4	9	6		1 0 0 0 0 0 0 0
BNE								jdisp
BT	saddr.bit, $addr16	PC ← $addr16 if (saddr.bit) = 1	3	5-9	12	9		0 1 1 1 0 B_2 B_1 B_0
								Saddr-offset
								jdisp
	sfr.bit, $addr16	PC ← $addr16 if sfr.bit = 1	4	7/9	16	13		0 0 0 0 1 0 0 0
								1 0 1 1 1 B_2 B_1 B_0
								Sfr-offset
								jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 1
								1 0 1 1 1 B_2 B_1 B_0
								jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 1
								1 0 1 1 0 B_2 B_1 B_0
								jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 1	3	5/7	12	9		0 0 0 0 0 0 1 0
								1 0 1 1 0 B_2 B_1 B_0
								jdisp

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z AC CY	Bytes B1 thru B5
Conditional Branch (cont)								
BF	saddr.bit,$addr16	PC ← $addr16 if (saddr.bit) = 0	4	5-9	15	12		0 0 0 0 1 0 0 0 1 0 1 0 1 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← $addr16 if sfr.bit = 0	4	7/9	16	13		0 0 0 0 1 0 0 0 1 0 1 0 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 0	3	5/7	12	9		0 0 0 0 0 0 1 1 1 0 1 0 1 B_2 B_1 B_0 jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 0	3	5/7	12	9		0 0 0 0 0 0 1 1 1 0 1 0 0 B_2 B_1 B_0 jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 0	3	5/7	12	9		0 0 0 0 0 0 1 0 1 0 1 0 0 B_2 B_1 B_0 jdisp
BTCLR	saddr.bit,$addr16	PC ← $addr16 if (saddr.bit) = 1 then reset (saddr.bit)	4	5-13	15	12		0 0 0 0 1 0 0 0 1 1 0 1 1 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← $addr16 if sfr.bit = 1 then reset sfr.bit	4	7/13	18	13		0 0 0 0 1 0 0 0 1 1 0 1 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← $addr16 if A.bit = 1 then reset A.bit	3	5/9	12	9		0 0 0 0 0 0 1 1 1 1 0 1 1 B_2 B_1 B_0 jdisp
	X.bit,$addr16	PC ← $addr16 if X.bit = 1 then reset X.bit	3	5/9	12	9		0 0 0 0 0 0 1 1 1 1 0 1 0 B_2 B_1 B_0 jdisp
	PSW.bit,$addr16	PC ← $addr16 if PSW.bit = 1 then reset PSW.bit	3	5/8	12	9	x x x	0 0 0 0 0 0 1 0 1 1 0 1 0 B_2 B_1 B_0 jdisp

6

Instruction Set (cont)

				Clocks			Flags	Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	Bytes	Int ROM	Branch	No Branch	Z AC CY	Bytes B1 thru B5
Conditional Branch (cont)								
DBNZ	rl,$addr16	rl ← rl − 1, then PC ← $addr16 if rl ≠ 0	2	3/5	9	6		0 0 1 1 0 0 1 R_0
								jdisp
	saddr,$addr16	(saddr) ← (saddr) − 1, then PC ← $addr16 if (saddr) ≠ 0	3	4-10	12	9		0 0 1 1 1 0 1 1
								Saddr-offset
								jdisp
CPU Control								
MOV	STBC,#byte	STBC ← byte	4	10	15			0 0 0 0 1 0 0 1
								1 1 0 0 0 0 0 0
								Data
								Data
SEL	RBn	RBS1–0 ← n, n = 0–3	2	2	6			0 0 0 0 0 1 0 1
								1 0 1 0 1 0 N_1 N_0
NOP		No Operation	1	2	3			0 0 0 0 0 0 0 0
EI		IE ← 1 (Enable Interrupt)	1	2	3			0 1 0 0 1 0 1 1
DI		IE ← 0 (Disable Interrupt)	1	2	3			0 1 0 0 1 0 1 0

Section 7
μPD78K3 Series:
16-Bit, Advanced Microcomputers

NEC

NEC Electronics Inc.

μPD7831xA/78P31xA 16-/8-Bit, Single-Chip CMOS Microcomputers, Real-Time Control Oriented

Description

The μPD7831xA family of microcomputers is designed for use in process control. They perform all the usual process control functions and are particularly well-suited for driving stepping motors and dc motors in servo loops. The processors include on-chip memory, timers, input/output registers, and a powerful interrupt handling facility. The μPD78310A/312A is constructed of high-speed CMOS circuitry and operates from a single +5-volt power supply.

The input frequency (maximum 12 MHz) is derived from an external crystal or an external oscillator. The internal processor clock is two-phase, and thus machine states are executed at a rate of 6 MHz. The shortest instructions require three states, making the minimum time 500 ns. The CPU contains a three-byte instruction prefetch queue, which allows a subsequent instruction to be fetched during execution of an instruction that does not reference memory.

Program memory is 8K bytes of mask-programmable ROM (μPD78312A only), and data memory is 256 bytes of static RAM. The μPD78310A is the ROMless version. μPD78P312A is a prototyping chip for μPD78312A. It has an on-chip 8K EPROM instead of a mask ROM.

Features

□ Complete single-chip microcomputer
  — 16-bit ALU
  — 8K ROM (μPD78312A only)
  — 256 bytes RAM
  — 1-bit and 8-bit logic

□ Instruction prefetch queue

□ 16-bit unsigned multiply and divide

□ String instructions

□ Memory expansion
  — 8085A bus-compatible
  — Total 64K address space

□ Large I/O capacity: up to 32 I/O port lines

□ Extensive timer/counter system
  — Two 16-bit up/down counters
  — Quadrature counting
  — Two 16-bit timers
  — Free-running counter with two 16-bit capture registers
  — Pulse-width modulated outputs
  — Timebase counter

□ Four-channel 8-bit A/D converter

□ Two 4-bit real-time output ports

□ Two nonmaskable interrupts

□ Eight hardware priority interrupt levels

□ Macroservice facility for interrupts gives the effect of eight DMA channels

□ Bidirectional serial port
  — Either UART or interface mode
  — Dedicated baud rate generator

□ Watchdog timer

□ Refresh output for pseudostatic RAM

□ Programmable HALT and STOP modes

□ One-byte call instruction

□ On-chip clock generator

□ CMOS silicon gate technology

□ +5-volt power supply

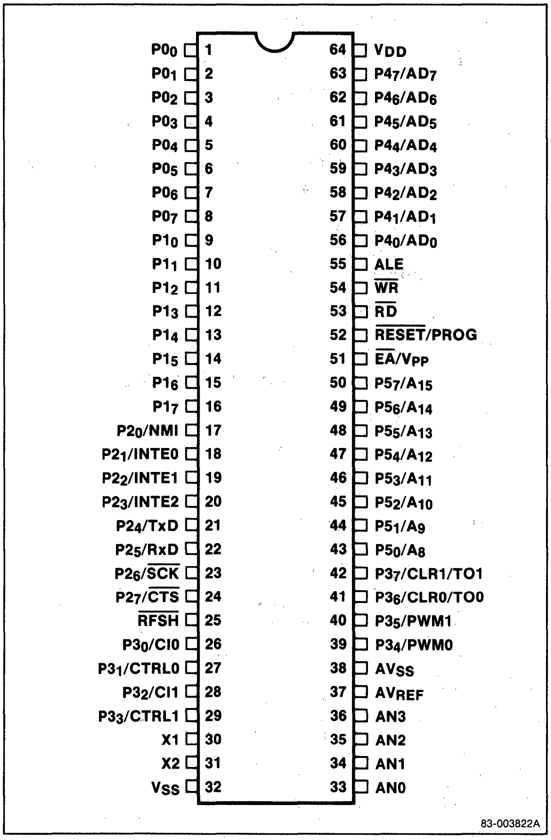
Ordering Information

Part Number	Package	ROM
μPD78310ACW	64-pin plastic shrink DIP	ROMless
μPD78310AGF-3BE	64-pin plastic QFP	
μPD78310AGQ-36	64-pin plastic QUIP	
μPD78310AL	68-pin plastic PLCC	
μPD78312ACW-xxx	64-pin plastic shrink DIP	Mask ROM
μPD78312AGF-xxx-3BE	64-pin plastic QFP	
μPD78312AGQ-xxx-36	64-pin plastic QUIP	
μPD78312AL-xxx	68-pin plastic PLCC	
μPD78P312ACW	64-pin plastic shrink DIP	OTP EPROM
μPD78P312AGF-3BE	64-pin plastic QFP	
μPD78P312AGQ-36	64-pin plastic QUIP	
μPD78P312AL	68-pin plastic PLCC	
μPD78P312ADW	64-pin ceramic shrink DIP with window (350 mil)	EPROM
μPD78P312AR	64-pin ceramic QUIP with window	

Notes: xxx is the ROM code number.

Pin Configurations

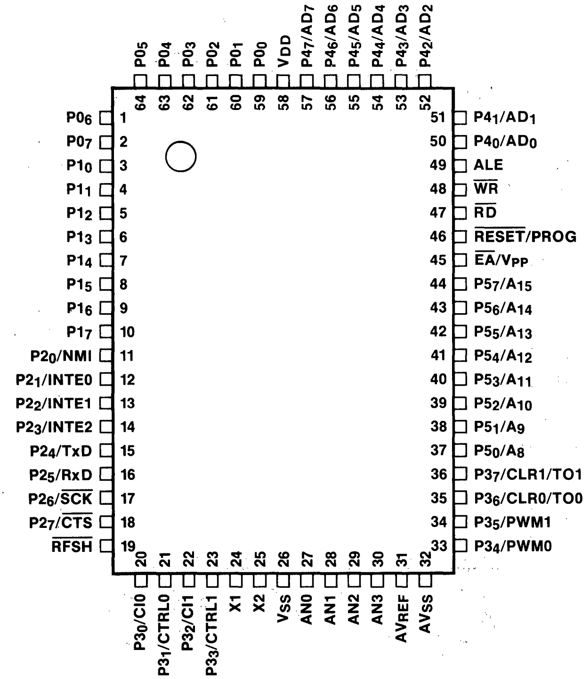
64-Pin Shrink DIP and QUIP, Plastic and Ceramic



83-003822A

Pin Configurations (cont)

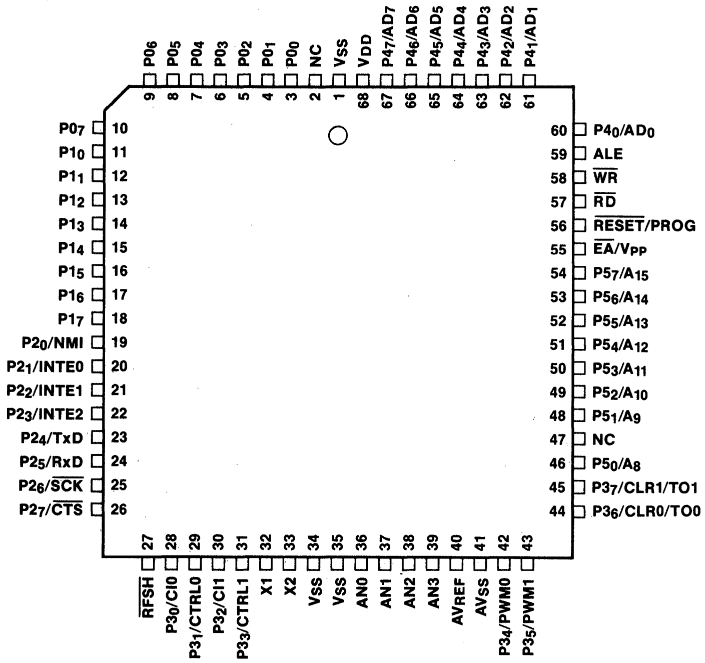
64-Pin Plastic QFP



83-003823B

Pin Configurations (cont)

68-Pin PLCC (Plastic Leaded Chip Carrier)



83-004442B

Pin Identification

Symbol	Function
AN0-AN3	A/D converter inputs
ALE	Address latch enable output
$\overline{EA}/V_{PP}$	External access control input; programming voltage
$P0_7$-$P0_0$	I/O port 0
$P1_7$-$P1_0$	I/O port 1
$P2_0$/NMI	Nonmaskable interrupt input
$P2_1$-$P2_3$/ INTE0-INTE2	Maskable interrupt inputs
$P2_4$/TxD	I/O port 2; serial transmit output
$P2_5$/RxD	I/O port 2; serial receive input
$P2_6/\overline{SCK}$	I/O port 2; serial clock output
$P2_7/\overline{CTS}$	I/O port 2; clear to send input
$P3_0$/CI0	Up/down counter 0 input
$P3_1$/CTRL0	Up/down counter 0 control input
$P3_2$/CI1	Up/down counter 1 input
$P3_3$/CTRL1	Up/down counter 1 control input
$P3_4$/PWM0	I/O port 3; pulse width modulated output 0
$P3_5$/PWM1	I/O port 3; pulse width modulated output 1
$P3_6$/CLR0/TO0	I/O port 3; counter 0 clear input; timer 0 output
$P3_7$/CLR1/TO1	I/O port 3; counter 1 clear input; timer 1 output
$P4_7$-$P4_0$/AD_7-AD_0	I/O port 4; external address; data bus
$P5_7$-$P5_0$/A_{15}-A_8	I/O port 5; high address byte output
$\overline{RD}$	Read strobe output
$\overline{RESET}$/PROG	External reset input; PROM programming mode
$\overline{RFSH}$	Refresh output
$\overline{WR}$	Write strobe output
X1	External crystal or external clock input
X2	External crystal
AV_{REF}	A/D reference voltage
AV_{SS}	Analog ground
V_{DD}	Power supply
V_{SS}	Power return

PIN FUNCTIONS

AN0-AN3 (A/D Converter Inputs)

AN0-AN3 are the four program selectable input channels for the A/D converter.

ALE (Address Latch Enable)

ALE is the address latch enable. It is to be used by external circuitry to latch the low-order 8 address bits during the first part of a read or write cycle.

$\overline{EA}/V_{PP}$

On μPD78312A, a low on $\overline{EA}$ enables use of external memory in place of on-chip ROM. The $\overline{EA}$ pin must be low on μPD78310A. On the μPD78P312A, this pin is used for programming voltage. In normal operation, it must be connected to V_{DD}.

$P0_7$-$P0_0$ (Port 0)

Port 0 consists of 8 bits, individually programmable for input/output or two 4-bit real-time (timer controlled) output ports.

$P1_7$-$P1_0$ (Port 1)

Port 1 consists of 8 bits, individually programmable for input/output.

$P2_0$/NMI (Port 2; Nonmaskable Interrupt)

Port $P2_0$ is dedicated to NMI, the nonmaskable external interrupt request.

$P2_1$-$P2_3$/INTE0-INTE2 (Port 2; Maskable Interrupts)

Ports $P2_1$-$P2_3$ are dedicated to INTE0, INTE1, and INTE2, the maskable external interrupt requests.

$P2_4$/TxD (Port 2; Serial Transmit)

$P2_4$ is an I/O port bit or the transmitted serial data output.

$P2_5$/RxD (Port 2; Serial Receive)

$P2_5$ is an I/O port bit or the received serial data input.

$P2_6/\overline{SCK}$ (Port 2; Serial Clock)

$P2_6$ is an I/O port bit or the serial shift clock output.

$P2_7/\overline{CTS}$ (Port 2; Clear to Send)

$P2_7$ is an I/O port bit or clear-to-send input (external serial transmission control) in the asynchronous communication mode. In the serial I/O interface mode, it becomes the serial receive clock I/O pin.

$P3_0$/CI0 (Port 3; Counter 0)

Port $P3_0$ is dedicated to CI0, the external count input for up/down counter 0.

$P3_1$/CTRL0 (Port 3; Counter 0 Control)

Port $P3_0$ is dedicated to CTRL0, the external control input for up/down counter 0.

P3$_2$/CI1 (Port 3; Counter 1)

Port P3$_2$ is dedicated to CI1, the external count input for up/down counter 1.

P3$_3$/CTRL1 (Port 3; Counter 1 Control)

Port P3$_3$ is dedicated to CTRL1, the external control input for up/down counter 1.

P3$_4$/PWM0 (Port 3; Pulse Width 0)

P3$_4$ is an I/O port bit or the pulse-width modulated output 0.

P3$_5$/PWM1 (Port 3; Pulse Width 1)

P3$_5$ is an I/O port bit or the pulse-width modulated output 1.

P3$_6$/CLR0/TO0 (Port 3; Counter 0 Clear; Timer 0)

P3$_6$ is an I/O port bit, or the clear input for up/down counter 0, or the timer 0 flip-flop output.

P3$_7$/CLR1/TO1 (Port 3; Counter 1 Clear; Timer 1)

P3$_7$ is an I/O port bit, or the clear input for up/down counter 1, or the timer 1 flip-flop output.

P4$_0$-P4$_7$/AD$_0$-AD$_7$ (Port 4; External Address/Data Bus)

Port 4 consists of 8 bits, programmable as a unit for input or output, or as the multiplexed address/data bus if external memory or external interface circuitry is used. The port is controlled by the memory mapping register. If the $\overline{\text{EA}}$ pin is low, port 4 is always an address/data bus.

P5$_0$-P5$_7$/A$_8$-A$_{15}$ (Port 5; High-Address Byte)

Port 5 consists of 8 bits, individually programmable for input or output, or the high-order address bits for external memory. Under control of the memory mask register, bits P5$_3$-P5$_0$ are used for 4K memory expansion, bits P5$_5$-P5$_0$ for 16K memory expansion, or bits P5$_7$-P5$_0$ for 56K memory expansion. If the $\overline{\text{EA}}$ pin is low, port 5 is always the high-order address bus.

$\overline{\text{RD}}$ (Read Strobe)

$\overline{\text{RD}}$ is the read strobe output. It is to be used by external memory (or data registers) to place data on the I/O bus during a read operation.

$\overline{\text{RESET}}$/PROG

This pin is used for the external reset input. A low level sets all registers to their specified reset values. During programming of the µPD78P312A, this pin is used to place the device into PROM programming mode.

$\overline{\text{RFSH}}$ (Refresh)

$\overline{\text{RFSH}}$ is the refresh pulse output to be used for external pseudostatic DRAM.

$\overline{\text{WR}}$ (Write Strobe)

$\overline{\text{WR}}$ is the write strobe output. It is to be used by external memory (or data registers) to latch data from the I/O bus during a write operation.

X1, X2 (External Crystal or Clock Input)

X1 and X2 are the external oscillator inputs or the connections for an external crystal. If an external clock is used, it is connected to X1 and its inverse is connected to X2. The system clock frequency is half the input frequency.

AV$_{REF}$ (A/D Reference Voltage)

AV$_{REF}$ is the reference voltage input for the A/D converter.

AV$_{SS}$ (Analog Ground)

AV$_{SS}$ is the analog ground pin.

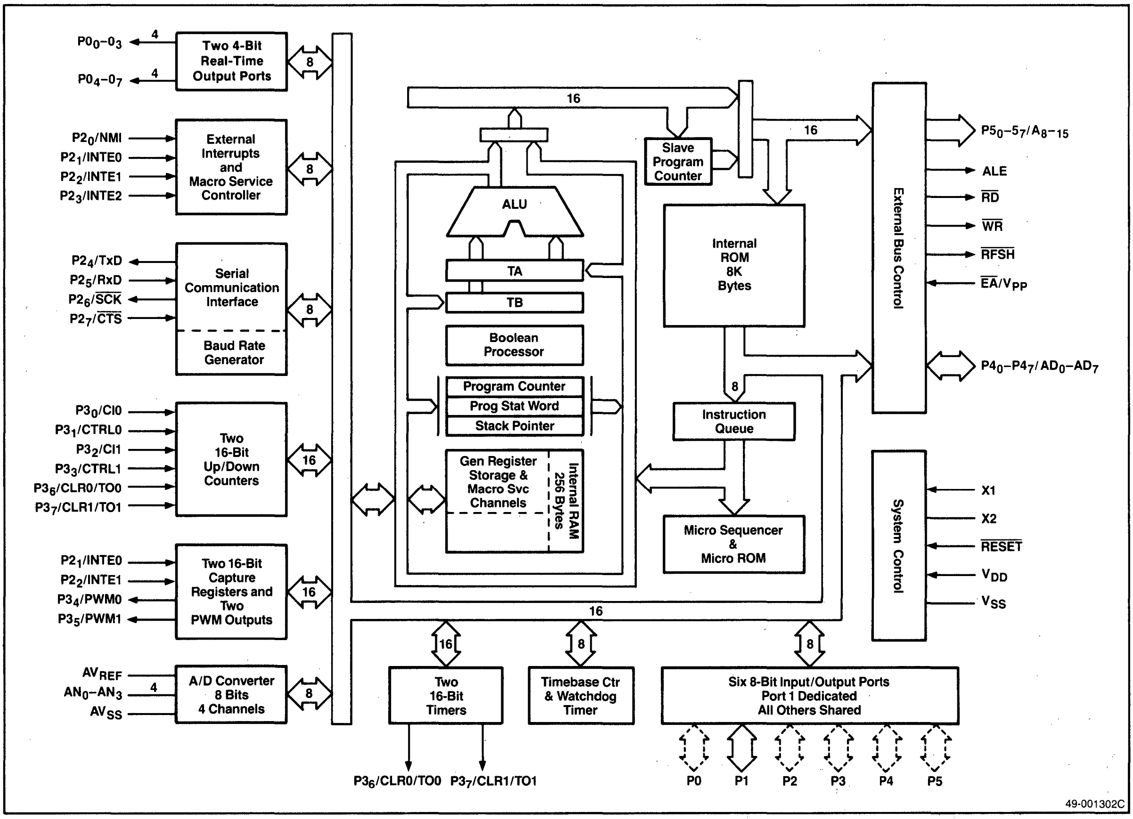
V$_{DD}$ (Power Supply)

V$_{DD}$ is the positive power supply input.

V$_{SS}$ (Power Return)

V$_{SS}$ is the power supply return, normally ground.

Block Diagram



FUNCTIONAL DESCRIPTION

On-chip features designed to facilitate process control include two 16-bit timers, quadrature counting, two 16-bit up/down counters, two pulse-width modulated outputs, a free-running counter with two capture registers, two 4-bit real-time (timer-controlled) output ports, an 8-bit A/D converter with four input channels, a timebase counter to generate widely spaced interrupts, and a watchdog timer to guard against infinite program loops.

In addition, a serial I/O port can be used in either an interface mode or an asynchronous communication mode. HALT and STOP modes are provided to conserve power at times when CPU action is not required.

All I/O, timer, and control registers are defined as special function registers and assigned addresses in the top 256 bytes of memory. The special function registers may be operated on directly by many of the arithmetic, logic, and move instructions of the CPU. Table 1 describes the registers.

Addressing

The μPD783101xA features 1-byte addressing of the special function registers and 1-byte addressing of the internal RAM. There are nine modes of addressing main memory, including autoincrement, autodecrement, indexing, and double indexing. There are 8- and 16-bit immediate operands.

Figure 1. Memory Map

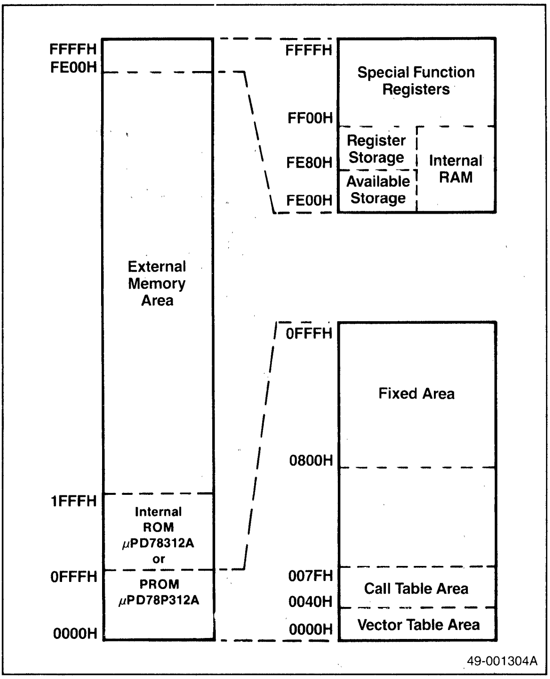
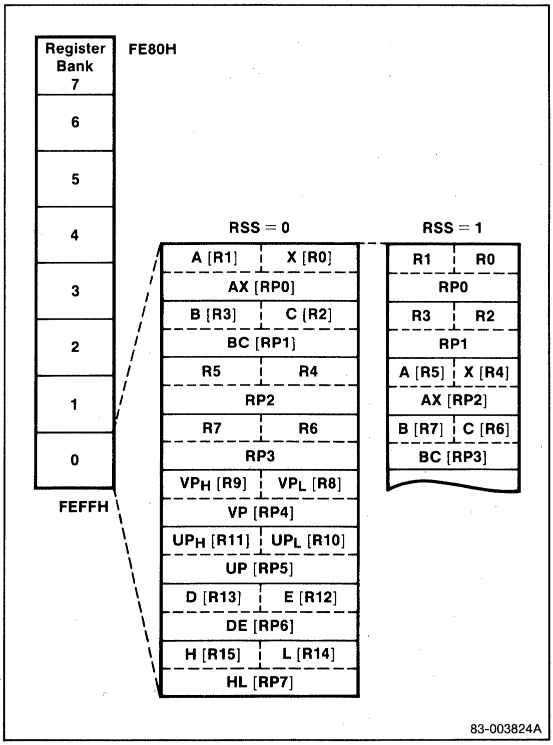


Figure 2. Register Designation and Storage



External Memory

External memory (figure 1) is supported by I/O port 4, an 8-bit multiplexed address/data bus. The memory mapping register controls the size of external memory as well as the number of additional wait states. High-order address bits are taken from I/O port 5 as required. No bits are required for 256 bytes of external memory; bits $P5_3$-$P5_0$ are used for 4K bytes, $P5_5$-$P5_0$ for 16K bytes, and $P5_7$-$P5_0$ for 56K bytes. Any remaining port 5 bits are available for I/O.

Refresh

The μPD7831xA has a refresh signal for use with the pseudostatic RAM. The refresh cycle can be set to one of four intervals ranging from 2.67 to 21.3 μs. The refresh is timed to follow a read or write operation so that there is no interference.

General Registers

The CPU has sixteen 8-bit registers (figure 2) that can also be used in pairs to function as 16-bit registers. A complete set of 16 general registers is mapped into each of 8 program-selectable register banks stored in RAM. Three bits in the PSW specify which of the register banks is active at any given time. Each register bank has two program-selectable accumulators.

The general registers of the μPD7831xA have both absolute and functinal names. AX is the functional name for the accumulator. Setting the RSS bit in the PSW to 1 transfers the AX and BC registers from their normal RP0 and RP1 positions to RP2 and RP3 as shown in figure 2. This adds considerable programming flexibility.

Program Status Word

Following is the program status word format.

15							8
0	RB_2	RB_1	RB_0	0	0	IE	0

7							0
S	Z	RSS	AC	UF	P/V	SUB	CY

RB_2-RB_0	Active register bank number
IE	Interrupt enable
S	Sign (1 if last result was negative)
Z	Zero (1 if last result was zero)
RSS	Register set select
AC	Auxiliary carry (carry out of 3rd bit)
UF	User flag
P/V	Parity or arithmetic overflow
SUB	Subtract (1 if last operation was subtract)
CY	Carry

Input/Output

All ports may be used for either latched output or high-impedance input. All ports except port 4 are bit-programmable for input or output. Port 0 is used for real-time or normal I/O. Port 1 is used for normal I/O. The low nibble of ports 2 and 3 is always used for control and the high nibble for control or normal I/O. Port 4 is used for the external address/data bus or byte-programmable I/O. Port 5 is used for the high bits of the external address or for normal I/O.

Real-Time Output Port

The real-time output port shares pins with I/O port 0. The high and low nibbles are treated separately or together. Data is transferred from a buffer to the port latches on either a timer or software command.

Serial Port

The serial port can operate in UART or interface mode with the baud rate and byte format under program control. The serial port also includes a dedicated baud rate generator.

Pulse-Width Modulated Outputs

The two independent pulse-width modulated outputs are controlled by two 16-bit modulus registers and counters. There are four programmable repetition rates ranging from 91.6 Hz to 23.4 MHz. Figure 3 shows one of these outputs.

Timers

The μPD7831xA has two 16-bit timers. The inputs to these timers may be the internal clock divided by 6 or by 128. Each timer has an associated modulus register to store the timer count. The timer counts down to zero, sets a flag, reloads from the modulus register, and then counts down again. The timer flags can be used under program control to generate interrupt requests and/or a square-wave output. TM0 also functions optionally as two one-shot timers.

Figure 4 is a diagram of the interval timers.

There is a free-running counter that counts the internal clock divided by 4 or by 16. The counter has two 16-bit capture registers. Capture is triggered by an external interrupt request or by the up/down counter clock.

The timebase counter generates a signal at one of four intervals ranging from 170 μs to 175 ms. The signal can be used to generate an interrupt request and/or an up/down counter capture.

Figure 3. Pulse-Width Modulated Output

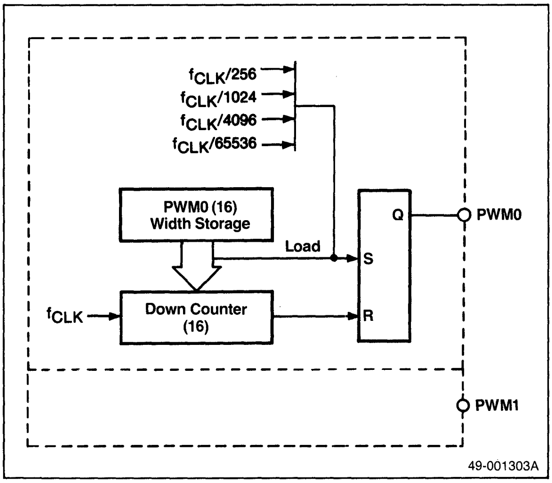
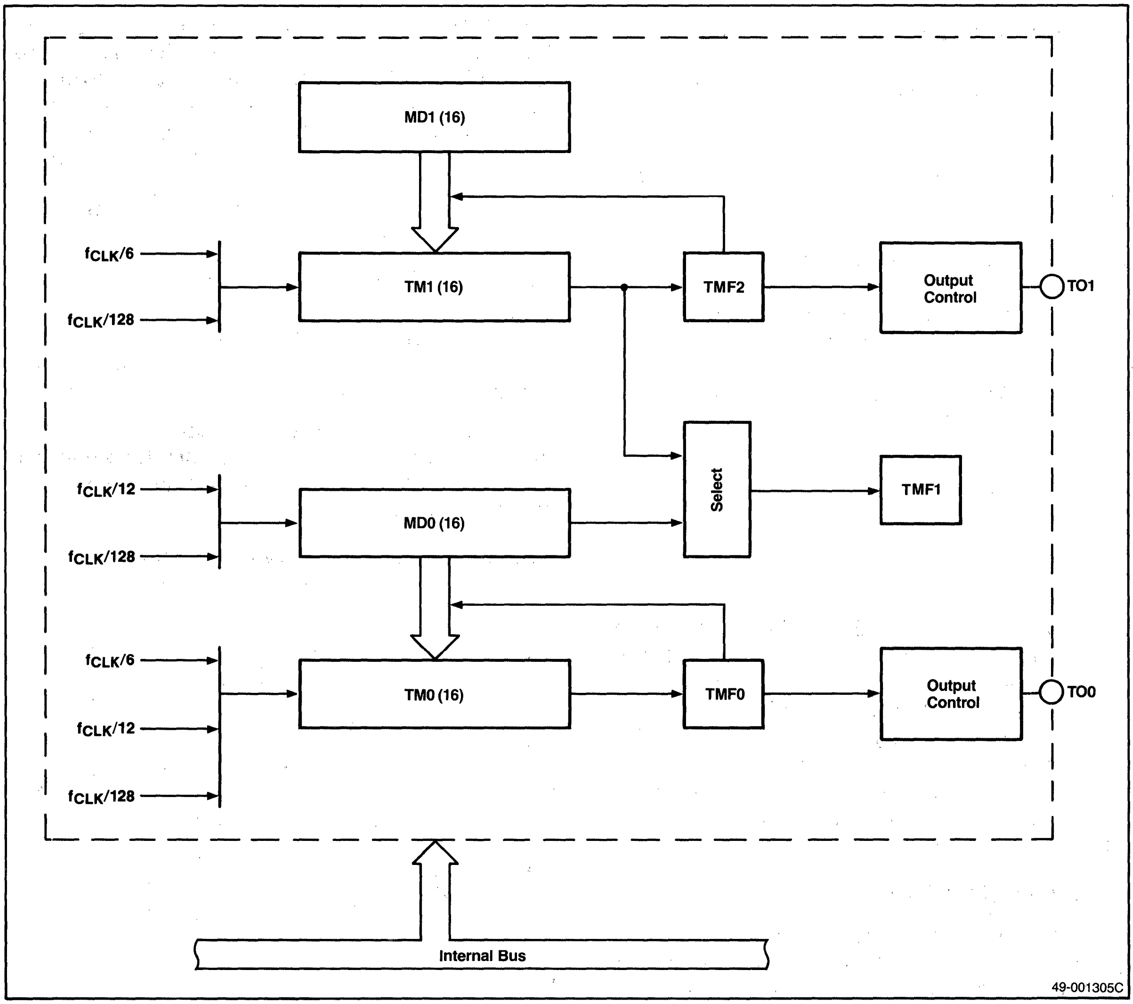


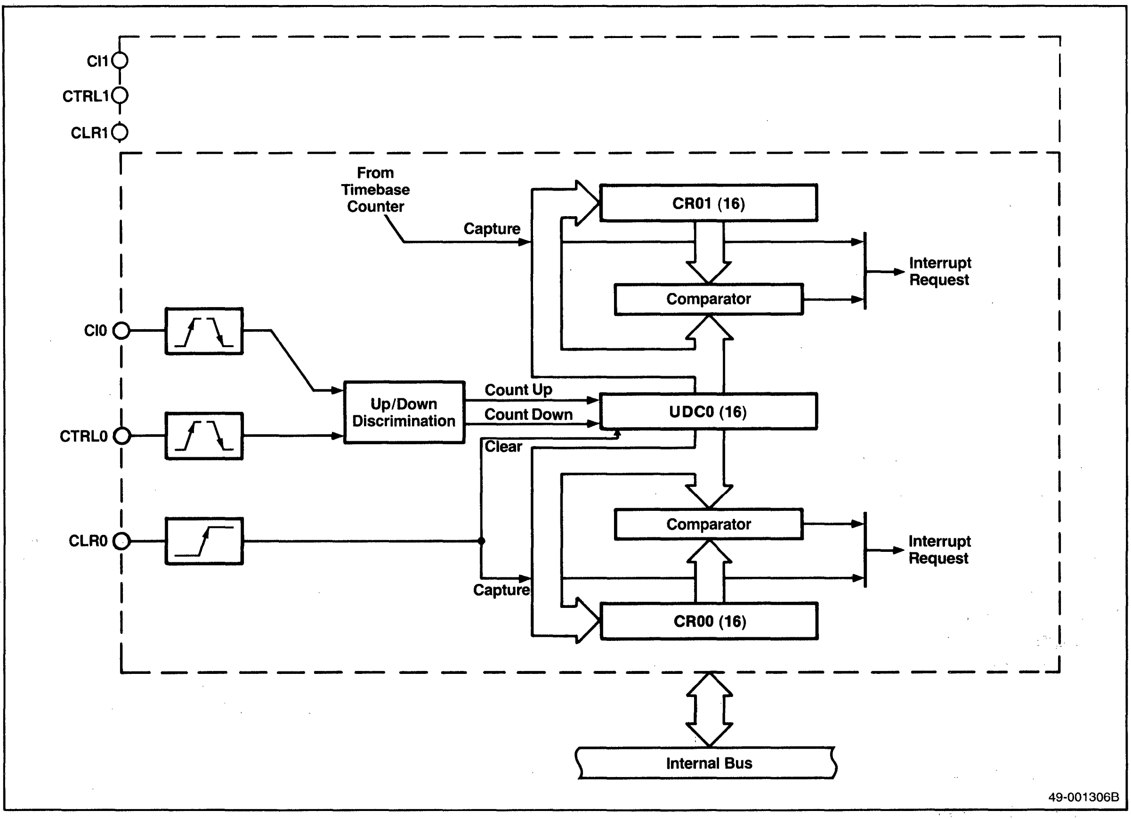
Figure 4. Timer Block Diagram



Up/Down Counters

The μPD7831xA has two 16-bit up/down counters, each of which has two capture/compare registers. There are three modes of operation: compare and interrupt, capture on external command, and capture on timebase counter command. There are five sources of counts: the internal clock divided by 3, the external clock, external independent up and down inputs, external clock with direction control, and external clock with automatic up/down discrimination. Figure 5 shows an up/down counter.

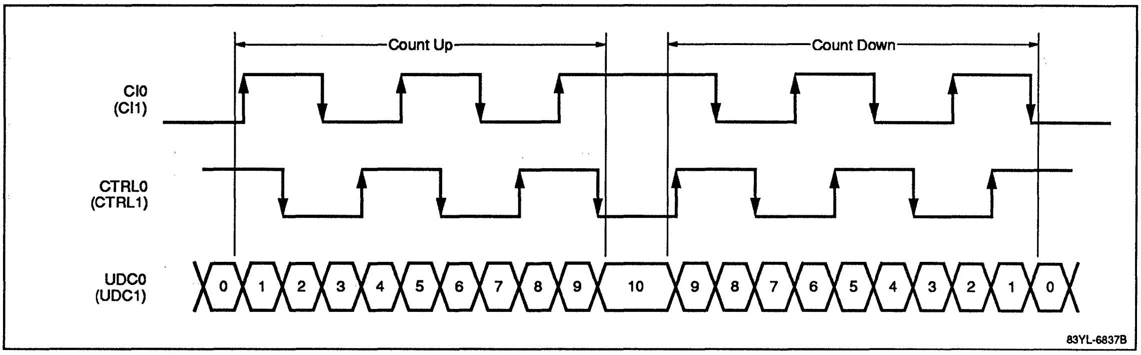
Figure 5. Up/Down Counter Block Diagram



Quadrature Counting

The two up/down counters, UDC0 and UDC1, have an optional quadrature counting mode, which is activated by specifying mode 4 in the counter unit input mode register, CUIM. It is designed to count the output of a two-phase pulsed optical shaft angle encoder. The input for phase A is the CI0 (or CI1) pin, and the input for phase B is the CTRL0 (or CTRL1) pin. The counter UDC0 (or UDC1) is incremented or decremented at both positive and negative transitions of both input signals. Whether it is incremented or decremented is dependent upon the relative phase of the two signals as illustrated in figure 6.

Figure 6. Counter Operation (Mode 4)



Standby Modes

HALT and STOP modes conserve power when CPU action is not required. In HALT mode, the CPU stops and the clock continues to run. Maskable interrupts can restart the CPU.

In STOP mode, the CPU and clock are both stopped. A $\overline{\text{RESET}}$ pulse or the nonmaskable external interrupt is required to restart them. There is also the option of slowing the system clock by a factor of four. The standby control register controls the standby modes and is a protected location written to only by a special instruction.

Watchdog Timer

The watchdog timer protects against inadvertent program loops. A nonmaskable interrupt occurs if the timer is not reset before a timeout occurs. There are four program-selectable intervals ranging from 5.5 to 349.3 ms. The watchdog timer can be disabled by software. The watchdog timer mode register controls the watchdog timer and is a protected location written to only by a special instruction.

A/D Converter

The A/D converter has four input channels and can operate in either scan or select mode. The A/D converter performs 8-bit successive approximation conversions, has a 30-μs conversion time, and is triggered either internally or externally. The A/D converter includes an on-chip sample and hold amplifier.

Interrupts

There are two nonmaskable interrupt sources: the external nonmaskable interrupt and the watchdog timer. Their relative priorities are software selectable.

There are eight hardware priority interrupt levels, level 0 having the highest priority and level 7 the lowest. The 15 maskable interrupt sources (table 2) are divided into five groups, and each group can, under program control, be assigned to any one of the priority levels.

Interrupts may be serviced by routines entered either by vectoring or by context switching. Context switching automatically saves all the general registers, the program status word, and the program counter. Figure 7 illustrates the mechanism of context switching.

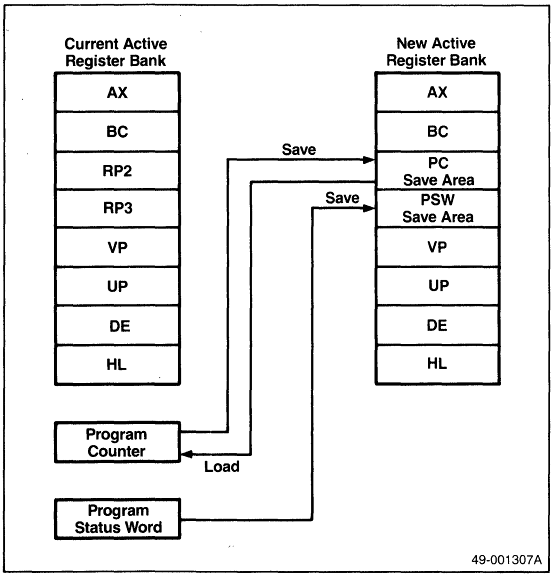
Finally, an optional macroservice function transfers data between any one special function register and memory without program intervention.

Macroservice

The macroservice controller can be programmed to perform word or byte transfers. It can transfer data from a special function register to memory or from memory to a special function register. Transfer events are triggered by interrupt requests and take place without software intervention.

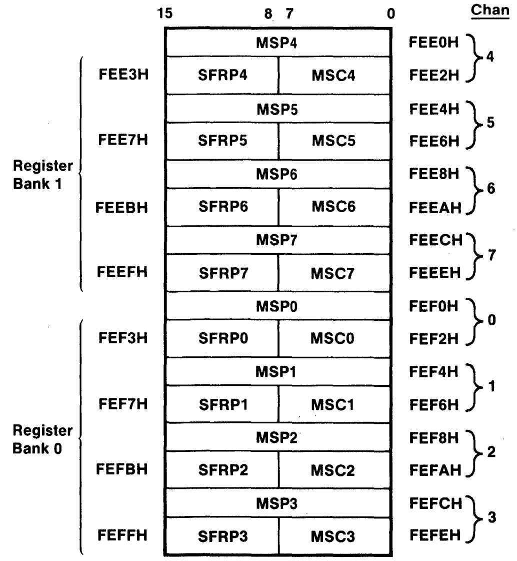
There are eight macroservice channels; channel control information is stored in RAM. This information (figure 8) consists of a 16-bit memory address (optionally incremented at each transfer), and 8-bit special function register designator, and an 8-bit transfer counter (decremented at each transfer). When the count equals 0, a context switch or vectored interrupt occurs.

Figure 7. Hardware Context Switching



49-001307A

Figure 8. Macroservice Pointer Addresses



Note:

[1] The macroservice pointers share storage with register banks 0 and 1.

[2] MSP = Memory address pointer
SFRP = Special function register pointer
MSC = Transfer counter

83-003825A

Table 1. Special Function Registers

Address	Function	Mnemonic		Read/Write	16-Bit Transfer	Reset State
FF00H	I/O port 0	P0		R/W	No	Undefined
FF01H	I/O port 1	P1		R/W	No	Undefined
FF02H	I/O port 2	P2		R/W (Note 1)	No	Undefined
FF03H	I/O port 3	P3		R/W (Note 1)	No	Undefined
FF04H	I/O port 4	P4		R/W	No	Undefined
FF05H	I/O port 5	P5		R/W	No	Undefined
FF08H FF09H	Capture/compare register 00	CR00L CR00H	CR00	R/W	Yes	Undefined
FF0AH FF08H	Capture/compare register 01	CR01L CR01H	CR01	R/W	Yes	Undefined
FF0CH FF0DH	Capture/compare register 10	CR10L CR10H	CR10	R/W	Yes	Undefined
FF0EH FF0FH	Capture/compare register 11	CR11L CR11H	CR11	R/W	Yes	Undefined
FF10H F11H	Capture register 0 (from FRC)	CPT0L CPT0H	CPT0	R/W	Yes	Undefined
FF12H FF13H	Capture register 1 (from FRC)	CPT1L CPT1H	CPT1	R/W	Yes	Undefined

7

Table 1. Special Function Registers (cont)

Address	Function	Mnemonic		Read/Write	16-Bit Transfer	Reset State
FF14H FF15H	PWM register 0 (duration)	PWM0L PWM0H	PWM0	R/W	Yes	Undefined
FF16H FF17H	PWM register 1 (duration)	PWM1L PWM1H	PWM1	R/W	Yes	Undefined
FF1CH FF1DH	Presettable up/down counter 0	UDC0L UDC0H	UDC0	R/W	Yes	Undefined
FF1EH FF1FH	Presettable up/down counter 1	UDC1L UDC1H	UDC1	R/W	Yes	Undefined
FF20H	Port 0 mode register	PM0		R/W	No	FFH
FF21H	Port 1 mode register	PM1		R/W	No	FFH
FF22H	Port 2 mode register	PM2		R/W (Note 1)	No	FFH
FF23H	Port 3 mode register	PM3		R/W (Note 1)	No	FFH
FF25H	Port 5 mode register	PM5		R/W	No	FFH
FF32H	Port 2 mode control register	PMC2		R/W	No	0FH
FF33H	Port 3 mode control register	PMC3		R/W	No	0FH
FF38H	Real-time output port control register	RTPC		R/W	No	08H
FF3AH FF3BH	Port 0 buffer register (Note 2)	P0L P0H		R/W	No	Undefined
FF40H	Memory expansion mode register	MM		R/W	No	30H
FF41H	Refresh mode register	RFM		R/W	No	10H
FF42H	Watchdog timer mode register	WDM		R/W	No	00H
FF44H	Standby control register	STBC		R/W	No	2nH (Note 3)
FF46H	Timebase mode register	TBM		R/W	No	00H
FF48H	External interrupt mode register	INTM		R/W	No	00H
FF4AH	In-service priority register	ISPR		R	No	00H
FF4EH	CPU control word	CCW		R/W	No	00H
FF50H	Serial communication mode register	SCM		R/W	No	00H
FF52H	Serial communication control register	SCC		R/W	No	00H
FF53H	Baud rate generator	BRG		R/W	No	00H
FF56H	Serial communication receive buffer	RXB		R	No	Undefined
FF57H	Serial communication transmit buffer	TXB		W	No	Undefined
FF60H	Free-running counter control register	FRCC		R/W	No	00H

Table 1. Special Function Registers (cont)

Address	Function	Mnemonic	Read/Write	16-Bit Transfer	Reset State
FF64H	Capture mode register	CPTM	R/W	No	00H
FF66H	PWM mode register	PWMM	R/W	No	00H
FF68H	A/D converter mode register	ADM	R/W	No	00H
FF6AH	A/D converter result register	ADCR	R	No	Undefined
FF70H	Count unit input mode register	CUIM	R/W	No	00H
FF72H	Up/down counter control register 0	UDCC0	R/W	No	00H
FF74H	Capture/compare control register	CRC	R/W	No	00H
FF7AH	Up/down counter control register 1	UDCC1	R/W	No	00H
FF80H	Timer 0 control register	TMC0	R/W	No	00H
FF82H	Timer 1 control register	TMC1	R/W	No	00H
FF88H FF89H	Timer 0	TM0L TM0 TM0H	R/W	Yes	Undefined
FF8AH FF8BH	Modulus/timer register 0	MD0L MD0 MD0H	R/W	Yes	Undefined
FF8CH FF8DH	Timer 1	TM1L TM1 TM1H	R/W	Yes	Undefined
FF8EH FF8FH	Modulus register 1	MD1L MD1 TM1H	R/W	Yes	Undefined
FFB0H to FFBFH	External area (Note 4)				
FFC0H	CRF00 interrupt control Up/down counter 0	CRIC00	R/W	No	47H
FFC1H	CRF00 macroservice control Up/down counter 0	CRMS00	R/W	No	Undefined
FFC2H	CRF01 interrupt control Up/down counter 0	CRIC01	R/W	No	47H
FFC4H	CRF10 Interrupt control Up/down counter 1	CRIC10	R/W	No	47H
FFC5H	CRF10 macroservice control Up/down counter 1	CRMS10	R/W	No	Undefined
FFC6H	CRF11 interrupt control Up/down counter 1	CRIC11	R/W	No	47H
FFC8H	EXIF0 interrupt control External interrupt INTE0	EXIC0	R/W	No	47H
FFC9H	EXIF0 macroservice control External interrupt INTE0	EXMS0	R/W	No	Undefined
FFCAH	EXIF1 interrupt control External interrupt INTE1	EXIC1	R/W	No	47H
FFCBH	EXIF1 macroservice control External interrupt INTE1	EXMS1	R/W	No	Undefined
FFCCH	EXIF2 interrupt control External interrupt INTE2	EXIC2	R/W	No	47H

Table 1. Special Function Registers (cont)

Address	Function	Mnemonic	Read/Write	16-Bit Transfer	Reset State
FFCDH	EXIF2 macroservice control External interrupt INTE2	EXMS2	R/W	No	Undefined
FFCEH	TMF0 interrupt control Timer flag	TMIC0	R/W	No	47H
FFCFH	TMF0 macroservice control Timer flag	TMMS0	R/W	No	Undefined
FFD0H	TMF1 interrupt control Timer flag	TMIC1	R/W	No	47H
FFD1H	TMF1 macroservice control Timer flag	TMMS1	R/W	No	Undefined
FFD2H	TMF2 interrupt control Timer flag	TMIC2	R/W	No	47H
FFD3H	TMF2 macroservice control Timer flag	TMMS2	R/W	No	Undefined
FFDAH	Receive error interrupt control Serial port	SEIC	R/W	No	47H
FFDCH	Receive interrupt control Serial port	SRIC	R/W	No	47H
FFDDH	Receive macroservice control Serial port	SRMS	R/W	No	Undefined
FFDEH	Transmit interrupt control Serial port	STIC	R/W	No	47H
FFDFH	Transmit macroservice control Serial port	STMS	R/W	No	Undefined
FFE0H	A/D converter interrupt control	ADIC	R/W	No	47H
FFE1H	A/D converter macroservice control	ADMS	R/W	No	Undefined
FFE2H	Timebase counter interrupt control	TBIC	R/W	No	47H

Notes:

(1) Bits 0-3 of port 2 and of port 3 are read-only.

(2) P0H and P0L are 4-bit buffer registers used to store data to be loaded into the high and low nibbles of the real-time output (P0). The high order 4 bits of P0H and the low order 4 bits of P0L are used.

(3) Bit 3 of the STBC is not affected by $\overline{\text{RESET}}$ (n = 0 or 8).

(4) External registers interfaced with these addresses can be accessed by special function register addressing.

Table 2. Interrupt Sources and Vector Addresses

	Default Priority	Mnemonic	Interrupt Source	Macroservice	Vector
Software	—	BRK	Break instruction	No	003EH
Nonmaskable Interrupts		NMI	External nonmaskable interrupt	No	0002H
	—	WDT	Watchdog timer	No	000AH
Maskable interrupts	0	CRF00	Up/down counter 0	Yes	001AH
	1	CRF01	Up/down counter 0	No	001CH
	2	CRF10	Up/down counter 1	Yes	001EH
	3	CRF11	Up/down counter 1	No	0020H
	4	EXIF0	External interrupt 0	Yes	0004H
	5	EXIF1	External interrupt 1	Yes	0006H
	6	EXIF2	External interrupt 2	Yes	0008H
	7	TMF0	Timer flag 0	Yes	000EH
	8	TMF1	Timer flag 1	Yes	0010H
	9	TMF2	Timer flag 2	Yes	0012H
	10	SEF	Serial port error	No	0022H
	11	SRF	Serial port receive buffer	Yes	0024H
	12	STF	Serial port transmit buffer	Yes	0026H
	13	ADF	A/D converter done flag	Yes	0028H
	14	TBF	Timebase counter flag	No	000CH
Reset	—	$\overline{\text{RESET}}$	External reset line	—	0000H

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

T_A +25°C

Power supply voltage V_{DD}	−0.5 to +7.0 V
Reference voltage, AV_{REF}	−0.5 V to V_{DD} +0.3 V
Power supply return, AV_{SS}	−0.5 to +0.5 V
Input voltage, V_{I1} (except $\overline{\text{RESET}}$ of μPD78P312A)	−0.5 to $+V_{DD}$ + 0.5
Input voltage, V_{I2} ($\overline{\text{RESET}}$ of μPD78P312A only)	−0.5 to +13.5 V
Output voltage, V_O	−0.5 to V_{DD} +0.5 V
Output current, low; I_{OL} (single pin)	4 mA
Output current, low; I_{OL}; total, all output pins (μPD78312/310A)	100 mA
Output current, low; I_{OL}; total, all output pins (μPD78P312A)	60 mA
Output current, high; I_{OH} (single pin)	−1 mA
Output current, high; I_{OH}; total, all output pins (μPD78312/310A)	−25 mA
Output current, high; I_{OH}; total, all output pins (μPD78P312A)	−15 mA
Operating temperature, T_{OPT}	−10 to +70 °C
Storage temperature, T_{STG}	−65 to +150 °C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Operating Frequency

Oscillator Frequency f_{XX}	T_A	V_{DD}
4 MHz ≤ f_{XX} ≤ 12 MHz	−10 to +70°C	+5.0 V 10%

Capacitance

T_A = +25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input capacitance	C_I	10	pF	f = 1 MHz; unmeasured pins returned to 0 V.
Output capacitance	C_O	20	pF	
I/O capacitance	C_{IO}	20	pF	

DC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±5%; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input low voltage	V_{IL1}	0		0.8	V	Except $\overline{EA}$ on μPD78310A/312A
	V_{IL2}	0		0.5	V	$\overline{EA}$ on (μPD78310A/312A only)
Input high voltage	V_{IH1}	2.2		V_{DD}	V	Except $P2_0$/NMI, X1, X2, $\overline{RESET}$
	V_{IH2}	3.8		V_{DD}	V	$P2_0$/NMI X1, X2, $\overline{RESET}$
Output low voltage	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output high voltage	V_{OH}	V_{DD} −1			V	I_{OH} = −1 mA
Input current	L_I			±10	μA	$P2_0$/NMI, $\overline{RESET}$ V_I = 0.45 V to V_{DD}
Input leakage current	I_{LI}			±10	μA	
Input/output leakage current	I_{LO}			±10	μA	
AV_{REF} current	AI_{REF}		1.5	5	mA	f_{CLK} = 6 MHz
V_{DD} supply current	I_{DD1}		30	60	mA	Operating mode; f_{CLK} = 6 MHz
	I_{DD2}		5	15	mA	Halt mode; f_{CLK} = 6 MHz
Data retention voltage	V_{DDDR}	2.5			V	Stop mode
Stop mode supply current	I_{DDDR}		3	15	μA	Stop mode; V_{DDDR} = 2.5 V
			10	50	μA	Stop mode; V_{DDDR} = 5.0 V ±10%

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Read/Write Operation						
System clock cycle time	t_{CYK}	166		2000	ns	(Note 1)
Address setup time to ALE ↓	t_{SAL}	150			ns	
Address hold time after ALE ↓	t_{HLA}	30			ns	(Note 4)
Address to $\overline{RD}$ ↓ delay time	t_{DAR}	230			ns	
$\overline{RD}$ ↓ to address floating	t_{FRA}			0	ns	
Address to data input	t_{DAID}			410	ns	
ALE ↓ to data input	t_{DLID}			230	ns	
$\overline{RD}$ ↓ to data input	t_{DRID}			180	ns	
ALE ↓ to $\overline{RD}$ ↓ delay time	t_{DLR}	60			ns	
Data hold time after $\overline{RD}$ ↑	t_{HRID}	0			ns	
$\overline{RD}$ ↑ to address active	t_{DRA}	50			ns	
$\overline{RD}$ ↑ to ALE ↑ delay time	t_{DRL}	100			ns	
$\overline{RD}$ width low	t_{WRL}	200			ns	
ALE width high	t_{WLH}	120			ns	
Address to $\overline{WR}$ ↓ delay time	t_{DAW}	300			ns	
ALE ↓ to data output	t_{DLOD}			190	ns	
$\overline{WR}$ ↓ to data output	t_{DWOD}			100	ns	
ALE ↓ to $\overline{WR}$ ↓ delay time (Note 2)	t_{DLW}	30			ns	
		110			ns	During refresh mode
Data setup time to $\overline{WR}$ ↑	t_{SODWR}	150			ns	

AC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Read/Write Operation (cont)						
Data setup time to $\overline{WR}$ ↓ (Note 3)	t_{SODWF}	30			ns	During refresh mode
Data hold time to $\overline{WR}$ ↑	t_{HWOD}	20			ns	(Note 4)
$\overline{WR}$ ↑ to ALE ↑ delay time	t_{DWL}	110			ns	
$\overline{WR}$ width low	t_{WWL}	200			ns	
Serial Port						
Serial clock cycle time	t_{CYSK}	1.33			μs	$\overline{SCK}$ output (Note 5)
		1.33			μs	$\overline{CTS}$ output (Note 6)
		1			μs	$\overline{CTS}$ input (Note 7)
Serial clock low level width	t_{WSKL}	580			ns	$\overline{SCK}$ output (Note 5)
		580			ns	$\overline{CTS}$ output (Note 6)
		420			ns	$\overline{CTS}$ input (Note 7)
Serial clock high level width	t_{WSKH}	580			ns	$\overline{SCK}$ output (Note 5)
		580			ns	$\overline{CTS}$ output (Note 6)
		420			ns	$\overline{CTS}$ input (Note 7)
$\overline{CTS}$ high, low level	t_{WCSH}, t_{WCSL}	3			t_{CYK}	Asynchronous mode
RxD setup time to $\overline{CTS}$ ↑	t_{SRXSK}	80			ns	
RxD hold time after $\overline{CTS}$ ↑	t_{HSKRX}	80			ns	
$\overline{SCK}$ ↓ to TxD delay time	t_{DSKTX}			210	ns	
A/D Converter $T_A = -10°C$ to $+70°C$; $V_{DD} = +5$ V ±10%; $AV_{REF} = 4.0$ V to V_{DD}; $AV_{SS} = V_{SS} = 0$ V						
Resolution		8			Bit	
Full scale error				0.4	%	t_{CYK} = 166 to 500 ns
Quantization error				±1/2	LSB	
Conversion time	t_{CONV}	180			t_{CYK}	t_{CYK} = 166 to 250 ns
		120			t_{CYK}	t_{CYK} = 250 to 500 ns
Sampling time	t_{SAMP}	36			t_{CYK}	t_{CYK} = 166 to 250 ns
		24			t_{CYK}	t_{CYK} = 250 to 500 ns
Analog input voltage	V_{IAN}	0		AV_{REF}	V	
Input impedance	R_{AN}		1000		mΩ	
Analog reference voltage	AV_{REF}	4.0		V_{DD}	V	
AV_{REF} current	AI_{REF}		1.5	5.0	mA	f_{CLK} = 6 MHz
Counter Operation						
CI0, CI1 high, low levels	t_{WCIH}, t_{WCIL}	3			t_{CYK}	
CTRL0, CRTL1 high, low levels	t_{WCTH}, t_{WCTL}	3			t_{CYK}	
CTRL0, CTRL1 setup time to CI ↑	t_{SCTCI}	2			t_{CYK}	Operating mode of count unit is set to mode 3. CI input is set to rising edge active.

AC Characteristics (cont)

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Counter Operation (cont)						
CTRL0, CTRL1 hold time after CI ↑	t_{HCICT}	5			t_{CYK}	
CLRO. CLR1 high, low level width	t_{WCRH}, t_{WCRL}	3			t_{CYK}	
CI0, CI1 setup time to CTRL	t_{S4CTCI}	6			t_{CYK}	Counter mode 4
CTRL0, CTRL1 setup time to CI	t_{H4CTCI}	6			t_{CYK}	Counter mode 4
CI0/CI1, CTRL0/CTRL1 cycle time	t_{CYC4}			250	KHz	Counter mode 4
External Interrupts and Reset						
NMI high, low level width	t_{WNIH}, t_{WNIL}	10			μs	
INTE0 high, low level width	t_{WI0H}, t_{WI0L}	3			t_{CYK}	
INTE1 high, low level width	t_{WI1H}, t_{WI1L}	3			t_{CYK}	
INTE2 high, low level width	t_{WI2H}, t_{WI2L}	3			t_{CYK}	
$\overline{\text{RESET}}$ high, low level width	t_{WRSH}, t_{WRSL}	10			μs	
V_{DD} rise, fall time	t_{RVD}, t_{FVD}	200			μs	

Notes:

(1) The internal clock (f_{CLK}) equals the oscillation clock (f_{XX}) divided by 2 or 8 as determined by bit 5 of the STBC. In this table, f_{XX} = 12 MHz and $f_{CLK} = f_{XX}/2$.

(2) During refresh operation, the $\overline{\text{WR}}$ signal falls to low level 1/2 clock cycle later than if there is no refresh.

(3) When accessing data from pseudostatic DRAMs (e.g. μPD4168) with the falling edge of the $\overline{\text{WR}}$ signal, the data setup time is t_{SODWF} instead of t_{SODWR}.

(4) Hold time is measured with C_L = 100 pF and R_L = 2 kΩ load, and includes the period necessary to guarantee V_{OH} and V_{OL}.

(5) I/O interface mode transmit data at a data rate of 750 kb/s.

(6) I/O interface mode receive data, internal clock, at a data rate of 750 kb/s.

(7) In the I/O interface mode this is the optional external clock for received data at a maximum rate of 1 MB/s.

Oscillator Characteristics

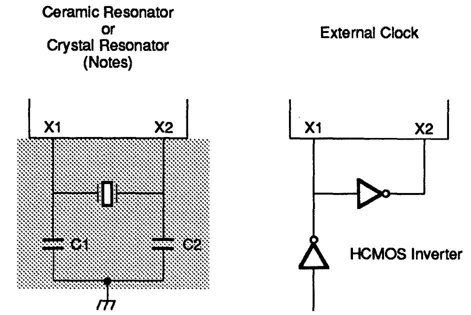
T_A = −10 to 70°C; V_{DD} = +5.0 V ±10%; V_{SS} = AV_{SS} = 0 V;
4 V ≤ AV_{REF} ≤ V_{DD}

Oscillator	Parameter	Symbol	Min	Max	Unit
Ceramic resonator or crystal resonator	Oscillation frequency	f_{XX}	4	12	MHz
External clock	X1 input frequency	f_X	4	12	MHz
	X1 input rise, fall time	t_{XR}, t_{XF}	0	30	ns
	X1 input high-low-level width	t_{WXH}, t_{WXL}	30	130	ns

Recommended Ceramic Resonators (μPD78310/312A)

			External Capacitance (pF)	
Manufacturer	Part No.	Frequency (MHz)	C1	C2
Murata Mfg. Co., Ltd.	CSA12.OMT	12.0	30	30
	CST12.OMT	12.0	Included	Included

Recommended Circuits



Notes:

1. When using a crystal resonator, the following external capacitor is recommended:
   C1 = C2 = 15pF
2. Oscillator circuit must be located as close as possible to the X1 and X2 pins.
3. To prevent noise from affecting operation, avoid locating other signal lines within the shaded area.

83YL-6838A

Timing Dependent on t_{CYK}

Symbol	Formula	Min/Max	Unit
t_{SAL}	1.5T − 100	Min	ns
t_{DAR}	2T − 100		
t_{DAID}	(3.5 + n) T − 170	Max	ns
t_{DLID}	(2 + n) T − 100		
t_{DRID}	(1.5 +n) T − 70		
t_{DLR}	0.5T − 20	Min	ns
t_{DRL}	T − 50		
t_{DRA}	0.5T − 30		
t_{WRL}	(1.5 + n) T − 50		
t_{WLH}	T − 40		
t_{DAW}	2T − 100		
t_{DLOD}	0.5T + 110	Max	ns
t_{DLW}	0.5T − 20 (normal operation)	Min	ns
	T − 50 (during refresh mode)		
t_{SODWR}	(1.5 +n) T − 100		
t_{SODWF}	0.5T − 50		
t_{DWL}	T − 50		
t_{WWL}	(1.5 + n) − 50		

Notes:

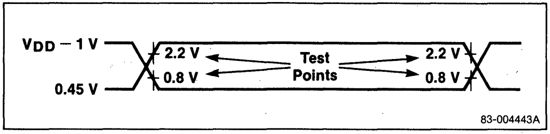
(1) n is the number of additional wait cycles specified by the MM register.

(2) T = t_{CYK} = $1/f_{CLK}$ = $2/f_{XX}$. f_{CLK} is the internal sytem clock frequency.

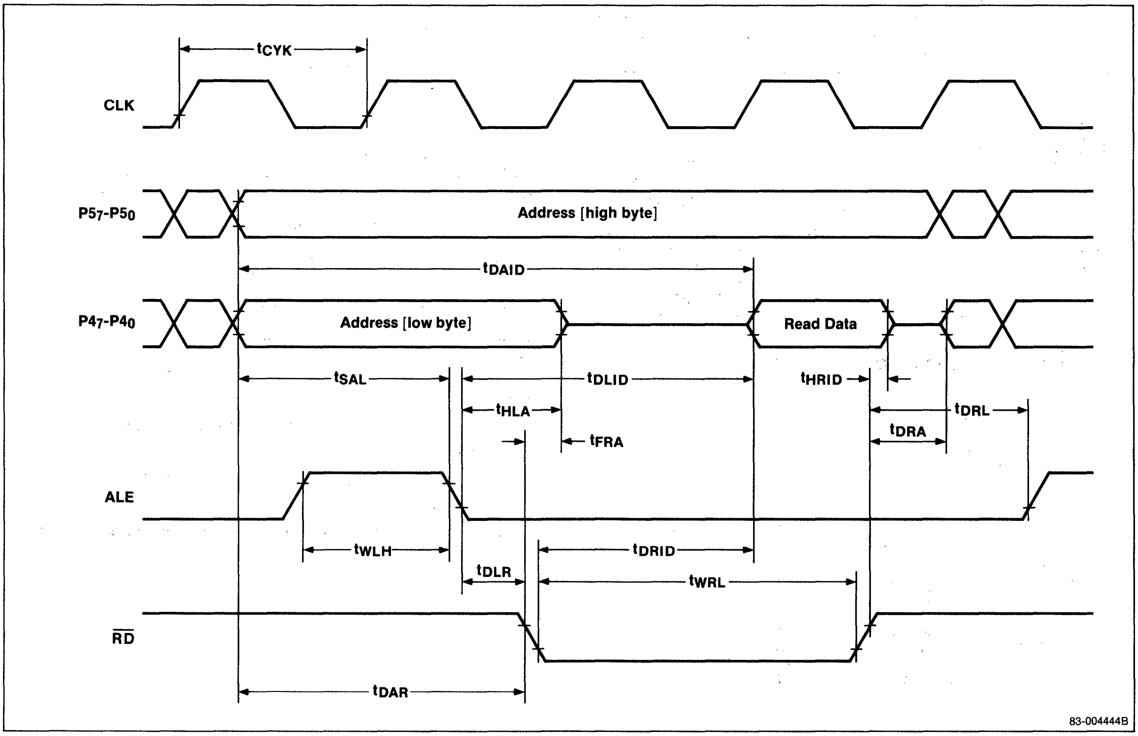
(3) Any parameter not included in this table is not dependent on f_{CLK}.

Timing Waveforms

AC Timing Test Points

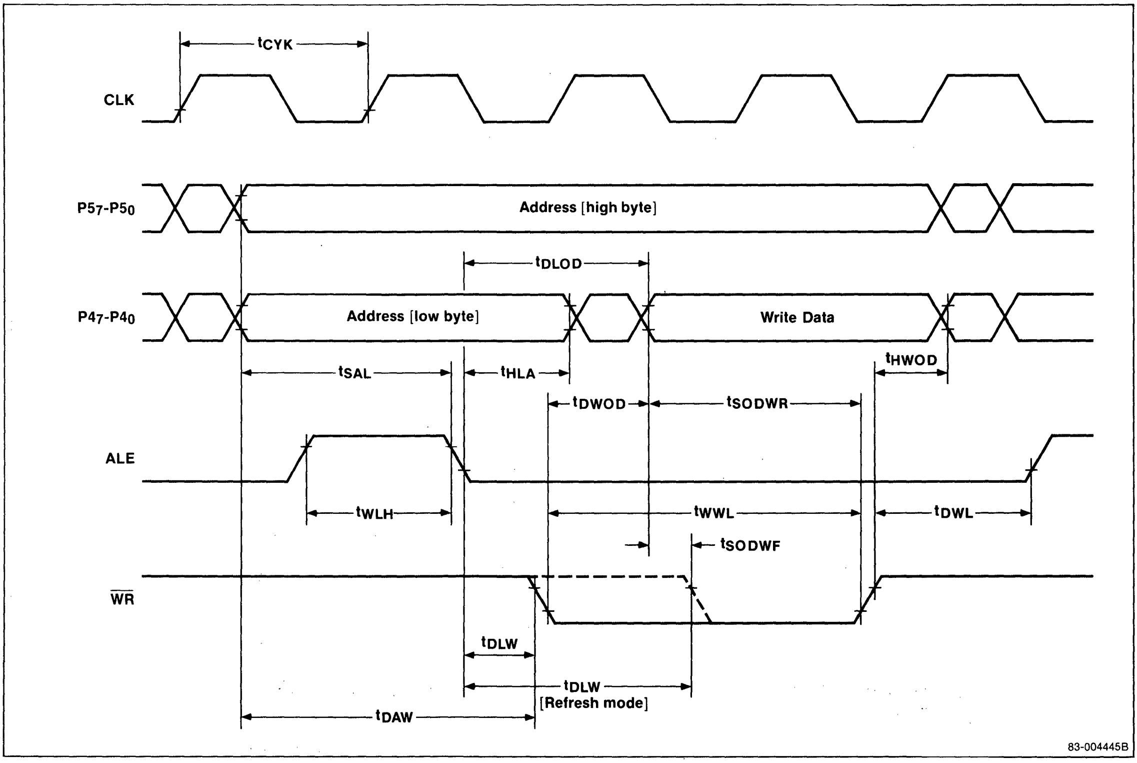


Read Operation



Timing Waveforms (cont)

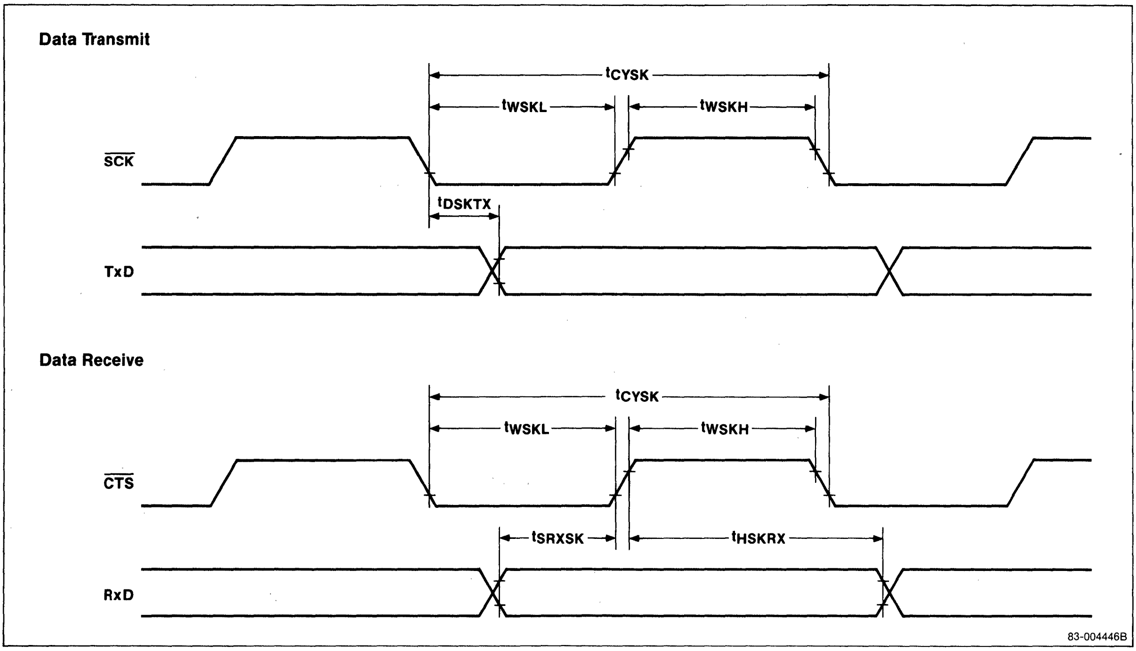
Write Operation



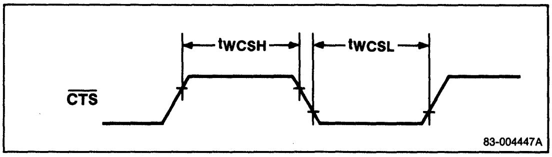
7

Timing Waveforms (cont)

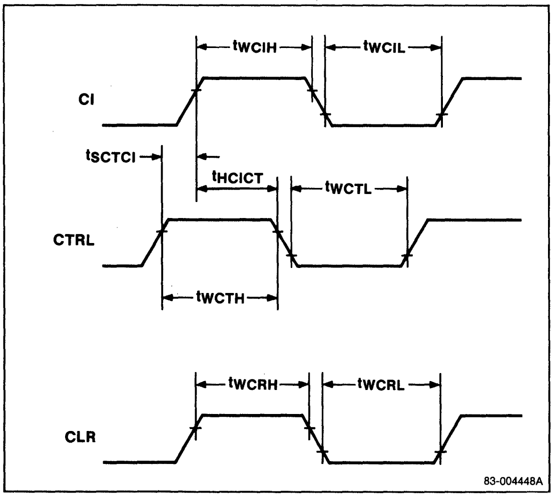
Serial Port, I/O Interface Mode



Serial Port, Asynchronous Mode Send Enable Input Timing

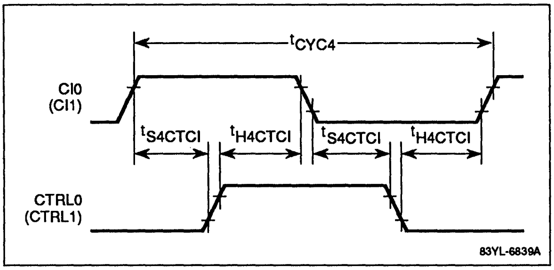


Counter Operation (Mode 3)

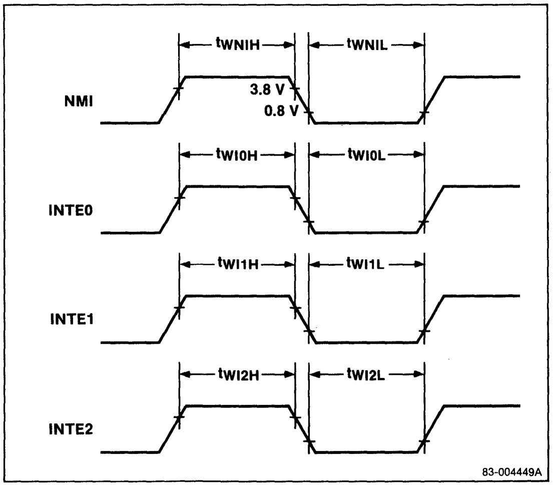


Timing Waveforms (cont)

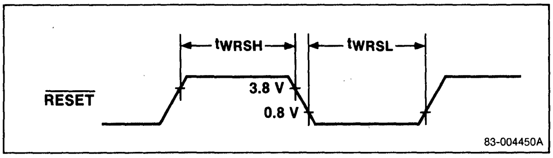
Count Timing Specification (Mode 4)



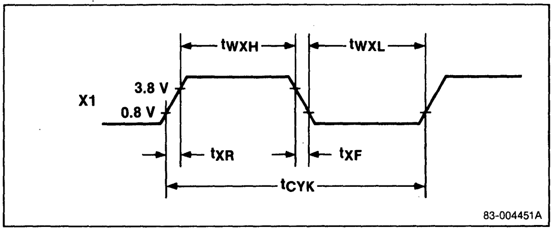
External Interrupts



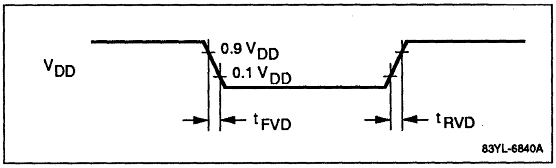
External Reset



External Clock



Data Retention Timing



PROM PROGRAMMING

The PROM in the μPD78P312A is an OTP or UVE EPROM with an 8,192 x 8-bit configuration. The pins listed in the table below are used to program the PROM.

When used in the normal operation mode, 5V ±10% is applied to the V_{DD} and V_{PP} pins. A voltage higher than V_{DD} should not be applied to other pins.

The programming characteristics of the μPD78P312A are identical to those of the μPD27C256A.

Pin	Function
V_{PP}	High voltage input (write/verify mode), high-level input (read mode)
PROG	High voltage input (write/verify mode, read mode)
A_0-A_7	Address input (lower 8 bits)
A_8-A_{12}	Address input (upper 8 bits)
D_0-D_7	Data input (write mode), data output (verify mode)
$\overline{CE}$	Program pulse input
$\overline{OE}$	Output enable input
V_{DD}	Power supply pin

Notes:

(1) Mask the window of the UVE EPROM version to protect the PROM from being erased accidentally.

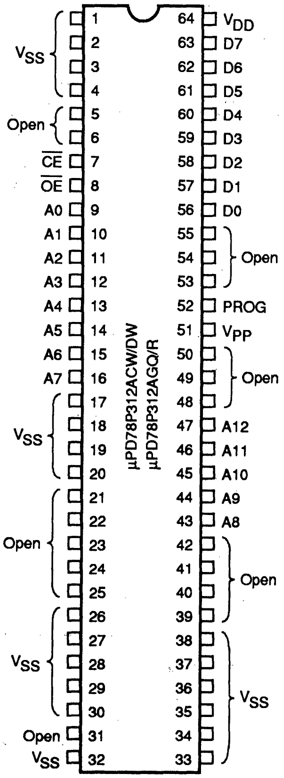
(2) The OTP EPROM version cannot be erased by ultraviolet rays because it does not have a window.

Programming Setup

Programming socket adaptors PA-78P312CW/GF/GQ/L are used to configure the μPD78P312A to fit a standard PROM socket. Set the PROM programmer to program the 27C256A. If the PROM programmer is an older model, check that the programming voltage does not exceed 12.5 volts.

Pin Functions, PROM Programming Mode

64-Pin Shrink DIP and QUIP, Plastic and Ceramic



Notes:

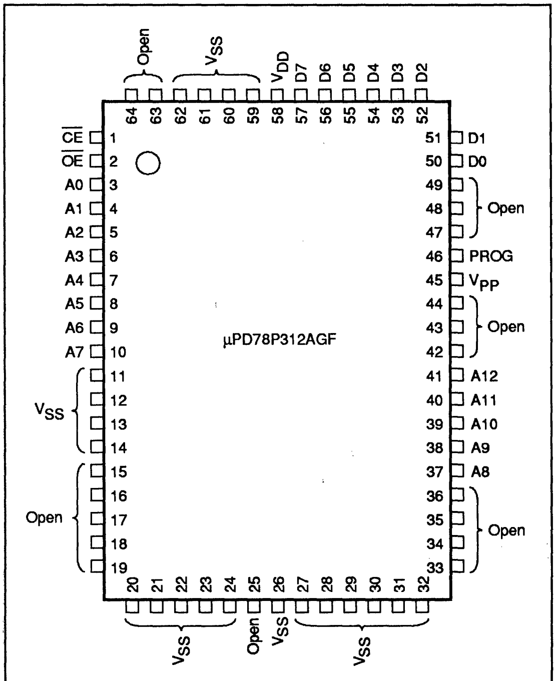
[1] V_{SS}: Ground this pin.

[2] Open: Do not connect this pin.

83YL-6841A

Pin Functions, PROM Programming Mode (cont)

64-Pin Plastic QFP (bent leads)



Notes:

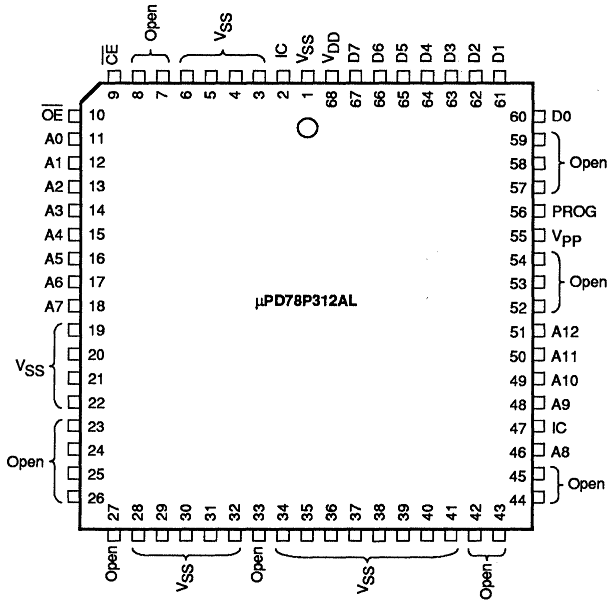
VSS: Ground this pin

Open: Do not connect this pin

83YL-6842A

Pin Functions, PROM Programming Mode (cont)

68-Pin PLCC



Notes:

[1] V_{SS}: Ground this pin.

[2] Open: Do not connect this pin.

83YL-6843B

PROM Programming Mode

When + 6 V is applied to the V_{DD} pin and + 12.5 V is applied to the PROG pin and V_{PP} pin, the μPD78P312A enters the program write/verify mode. Operation in this mode is determined by the setting of $\overline{CE}$ and $\overline{OE}$ pins as indicated in the table below.

Mode	$\overline{CE}$	$\overline{OE}$	V_{PP}	V_{DD}	PROG
Write	L	H	+12.5 V	+6V	+12.5 V
Verify	H	L			
Program inhibit	H	H			
Read (Note 2)	L/H	L	+5 V	+5 V	+12.5 V
Read (Note 3)	L/H	H			

Notes:

(1) When +12.5 V is applied to V_{PP} and +6 V is applied to V_{DD}, both $\overline{CE}$ and $\overline{OE}$ must not be set to the low level (L) simultaneously.

(2) Data is output from the D_0-D_7 pins.

(3) D_0-D_7 are high impedance.

Recommended Conditions for Unused Pins

Table 3 describes how to set unused pins when programming the PROM.

Table 3. Recommended Conditions for Unused Pins

Pin	Recommended Connection
$P0_0$-$P0_3$	Connect to V_{SS}
$P0_4$, $P0_5$	Open
$P2_0$-$P2_3$	Connect to V_{SS}
$P2_5$-$P2_7$, $\overline{RFSH}$	Open
$P3_0$-$P3_3$, X1	Connect to V_{SS}
X2	Open
AN0-AN3, AV_{REF}, AV_{SS}	Connect to V_{SS}
$P3_4$-$P3_7$, $P5_5$-$P5_7$, $\overline{RD}$, $\overline{WR}$, ALE	Open

PROM Write Procedure

Data can be written to the PROM by using the following procedure.

(1) Set the pins not used for programming as indicated in table 3, and supply + 6 V to the V_{DD} pin, and + 12.5 V to the V_{PP} and PROG pins.

(2) Provide the initial address.

(3) Provide write data.

(4) Provide a 1 ms program pulse (active low) to the $\overline{CE}$ pin.

(5) Use the verify mode to test the data. If the data has been written, proceed to (7), if not, repeat steps (3) to (5). If the data cannot be correctly written in 25 attempts, go to step (6).

(6) Classify the PROM as defective and cease write operation.

(7) Provide write data and supply program pulse (for additional writing) for 3 ms times the number of repeats performed between steps (3) to (5).

(8) Increment the address.

(9) Repeat steps (3) to (8) until the last address is reached.

PROM Read Procedure

The contents of the PROM can be read out to the external data bus D_0-D_7 by using the following procedure.

(1) Set the unused pins as indicated in table 3.

(2) Supply + 5 V to the V_{DD} pin and V_{PP} pin, and + 12.5 V to the PROG pin.

(3) Input the address of the data to be read to the A_0 to A_{12} pins.

(4) Put an active low pulse of at least 1 μs on the $\overline{OE}$ pin.

(5) Data is output to the D_0 to D_7 pins.

Erasure

The UVE EPROM can be erased by exposing the window to light having a wavelength shorter than 400 nm, including ultraviolet rays, direct sunlight, and fluorescent light. To prevent unintentional erasure, mask the window.

Typically, data is erased by 254-nm ultraviolet rays. A minimum lighting level of 15 W s/cm^2 (ultraviolet ray intensity x exposure time) is required to completely erase written data. Erasure by an ultraviolet lamp rated at 12,000 μW/cm^2 takes approximately 15 to 20 minutes. Remove any filter on the lamp and place the device within 2.5 cm of the lamp tubes.

DC Programming Characteristics

T_A = 25 ±5°C; V_{IP} = 12.0 ±0.5 V; V_{SS} = 0 V

Parameter	Symbol	Symbol (Note)	Min	Typ	Max	Unit	Condition
High-level input voltage	V_{IH}	V_{IH}	2.2		V_{DDP} + 0.3	V	
Low-level input voltage	V_{IL}	V_{IL}	–0.3		0.8	V	
Input leakage current	V_{LIP}	V_{LI}			10	μA	$0 \leq V_I \leq V_{DDP}$
High-level output voltage	V_{OH}	V_{OH}	V_{DD}–1			V	I_{OH} = –1.0 mA
Low-level output voltage	V_{OL}	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output leakage current	I_{LO}	—			10	μA	$0 \leq V_o \leq V_{DDP}$, $\overline{OE} = V_{IH}$
PROG pin high voltage input current	I_{IP}	—			±10	μA	
V_{DDP} power supply voltage	V_{DDP}	V_{DD}	5.75	6.0	6.25	V	Program memory write mode
			4.5	5.0	5.5	V	Program memory read mode
V_{PP} power supply voltage	V_{PP}	V_{PP}	12.2	12.5	12.8	V	Program memory write mode
				$V_{PP} = V_{DDP}$		V	Program memory read mode
V_{DDP} power supply current	I_{DD}	I_{DD}		10	30	mA	Program memory write mode
				10	30	mA	Program memory read mode $\overline{CE} = V_{IL}$, $V_I = V_{IH}$
V_{PP} power supply current	I_{PP}	I_{PP}		10	30	mA	Program memory write mode $\overline{CE} = V_{IL}$, $\overline{OE} = V_{IH}$
				1	100	μA	Program memory read mode

Notes:

(1) Corresponding symbols for the μPD27C256A

AC Programming Characteristics

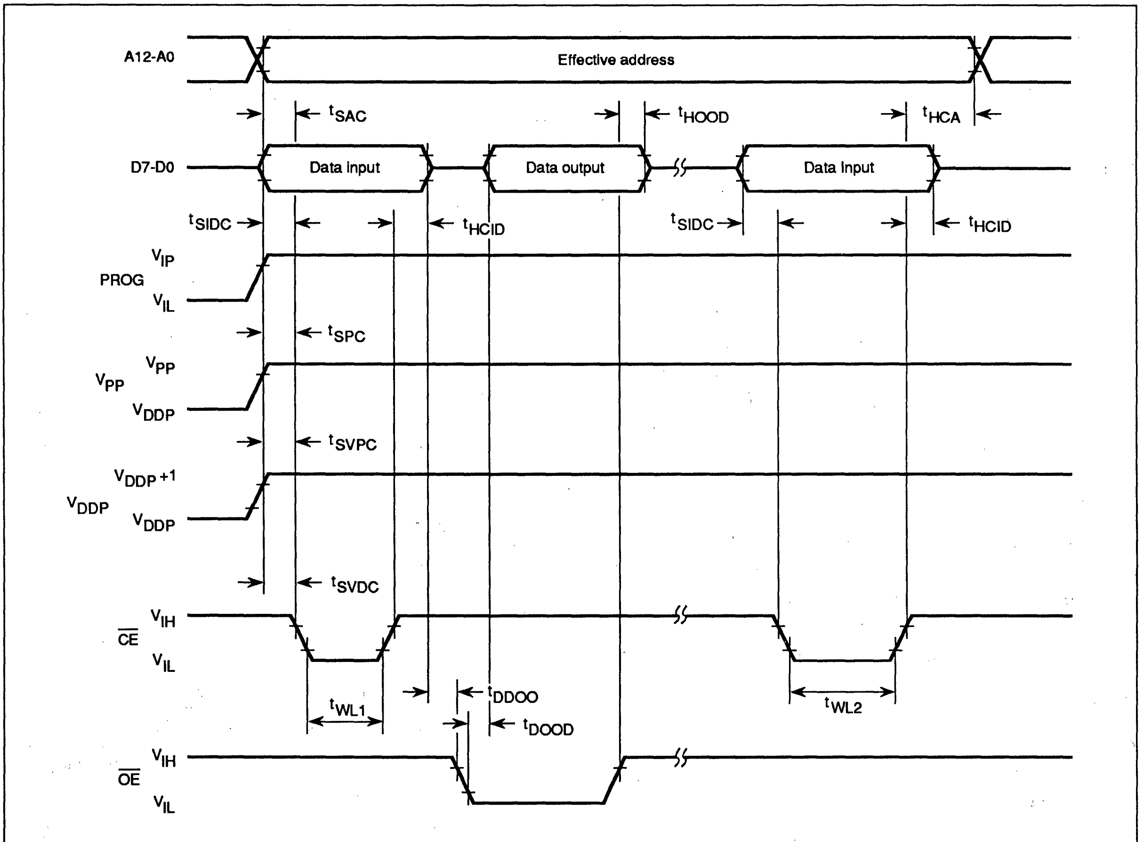
T_A = 25 ±5°C; V_{IP} = 12.0 ±0.5 V; V_{SS} = 0 V

Parameter	Symbol	Symbol (Note)	Min	Typ	Max	Unit	Condition
Address setup time to $\overline{CE}$ ↓	t_{SAC}	t_{AS}	2			μs	
Data to $\overline{OE}$ ↓ delay time	t_{DDOO}	t_{OES}	2			μs	
Input data setup time to $\overline{CE}$ ↓	t_{SIDC}	t_{DS}	2			μs	
Address hold time after $\overline{CE}$ ↑	t_{HCA}	t_{AH}	2			μs	
Input data hold time after $\overline{CE}$ ↑	t_{HCID}	t_{DH}	2			μs	
Output data hold time after $\overline{OE}$ ↑	t_{HOOD}	t_{DF}	0		130	ns	
V_{PP} setup time before $\overline{CE}$ ↓	t_{SVPC}	t_{VPS}	2			μs	
V_{DDP} setup time before $\overline{CE}$ ↓	t_{SVDC}	t_{VDS}	2			μs	
Initial program pulse width	t_{WL1}	t_{PW}	0.95	1.0	1.05	ms	
Additional program pulse width	t_{WL2}	t_{OPW}	2.85		78.75	ms	
PROG high-voltage input setup time before $\overline{CE}$ ↓	t_{SPC}		2			μs	
Address to data output time	t_{DAOD}	t_{ACC}			2	μs	$\overline{OE} = V_{IL}$
$\overline{OE}$ ↓ to data output time	t_{DOOD}	t_{OE}			1	μs	
Data hold time after $\overline{OE}$ ↑	t_{HCOD}	t_{DF}	0		130	ns	
Data hold time after address not valid	t_{HAOD}	t_{OH}	0			ns	$\overline{OE} = V_{IL}$

Notes:

(1) Corresponding symbols for the μPD27C256A

PROM Write Mode Timing



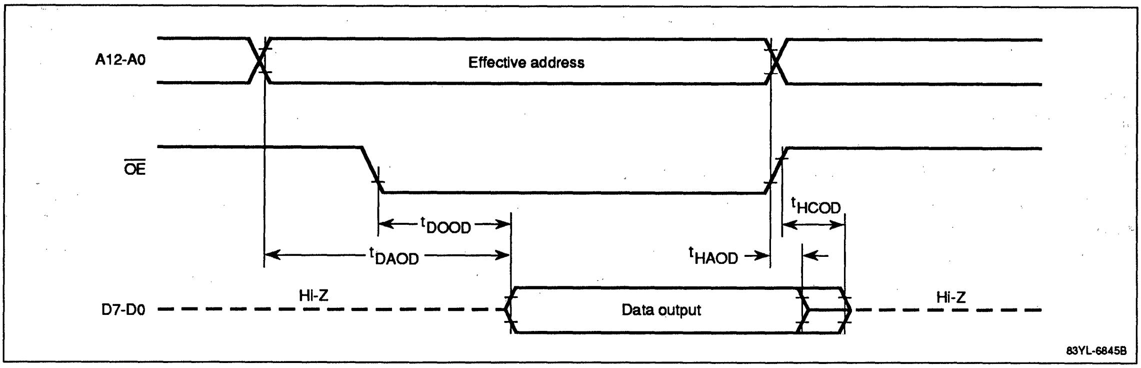
Notes:

[1] V_{DDP} must be applied before V_{PP} is applied and must be removed after V_{PP} is removed.

[2] V_{PP} must not exceed +13 V including overshoot voltage.

83YL-6844B

PROM Read Mode Timing



83YL-6845B

INSTRUCTION SET

The instruction set for the μPD7831xA has 8- and 16-bit arithmetic instructions including: a 16 x 16-bit unsigned multiply with a 32-bit product; a 32 by 16-bit unsigned divide with a 32-bit quotient and a 16-bit remainder. The instruction set also executes an 8-bit and a 16-bit shift and rotate by count, 1-and 8-bit logic, and 1-, 2-, and 3-byte call instructions. String manipulation instructions are also included.

Branch

There are four addressing modes for unconditional branching. Branch instructions exist to test single bits in the program status word, the 16-bit accumulator, the special function registers, and internal RAM. The instruction set also includes multiple register PUSH and POP instructions.

Addressing

On-chip RAM locations FE20H through FEFFH can be addressed by "saddr" addressing, in which the machine code specifies the address by its low-order byte only. This mode is also used to address the first 32 special function registers, addresses FF00H through FF1FH.

Timing

Access to on-chip ROM requires one state per byte, on-chip RAM two states per byte, and external memory four states per byte minimum.

The States column of the instruction set listing indicates the number of states required to execute an instruction after it has been fetched. In "saddr" addressing, the number after the slash is applicable when addressing special function registers FF00H through FF1FH. In conditional branch instructions, the number in parentheses is applicable when the branch is not taken. String instructions are interruptable, and the number in parentheses applies if the instruction has been interrupted during its execution.

The Idle States column indicates the number of states during which the CPU does not use the peripheral bus. They are therefore available for fetching succeeding instructions. If sufficient idle states are available, prefetching will continue until the buffer is full, so as many as three bytes can be pre-fetched in this manner. If the instructions are stored in external memory, a minimum of four states is required for each byte. Idle states from each instruction are used in multiples of four, and any states in excess of multiples of four are lost.

Symbols

Symbols designations, and codes used in the instruction set are explained in the following tables.

In addition to the general register designations (such as $P_2P_1P_0$, $Q_2Q_1Q_0$ and $R_2R_1R_0$), the following designations appear in the Operation Code column.

$B_2B_1B_0$	Bit number (bit = 0 through 7) in single-bit instructions
$N_2N_1N_0$	Number of bits (n = 0 through 7) in shift and rotate instructions
$N_2N_1N_0$	Register bank number (n = 0 through 7) in BRKCS and SEL instructions

Symbols

Symbol	Meaning
r	R0-R15
r1	R0-R7
r2	C, B
rp	RP0-RP7*
rp1	RP0-RP7*
rp2	DE, HL, VP, UP
sfr	Special function register, 8 bits
sfrp	Special function register, 16 bits
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7. Bits set to 1 indicate register pairs to be pushed/popped to/from the stack. RP5 pushed/popped by PUSH/POP: SP is stack pointer. PSW pushed/popped by PUSHU/POPU: RP5 is stack pointer
mem	Register indirect: [DE], [HL], [DE+], [HL+], [DE–], [HL–], [VP], [UP] Base index mode: [DE + A], [HL +A], [DE + B], [HL + B], [VP + DE], [VP + HL] Base Mode: [DE + byte], [HL + byte], [VP + byte], [UP + byte], [SP + byte] Index mode: Word [A], word [B], word [DE], word [HL]
saddr	FE20H-FF1FH: immediate byte addresses one byte in RAM, or label
saddrp	FE20H-FF1FH: immediate byte (bit 0 = 0) addresses one word in RAM or label
#word	16 bits of immediate data or label
#byte	8 bits of immediate data or label
jdisp	8-bit two's complement displacement (immediate data)
f_0-f_{10}	Eleven bits of immediate data corresponding to addr11
t_0-t_4	Five bits of immediate data corresponding to addr5

*rp and rp1 refer to the same register pairs, but generate different machine code.

Symbols

Symbol	Meaning
.bit	3 bits of immediate data (bit position in byte), or label
n	3 bits of immediate data
!addr16	16-bit absolute address specified by an immediate address or label
$addr16	Relative branch address {(PC) + jdisp} or label
addr16	16-bit address
!addr11	11-bit immediate address or label
addr11	0800H to 0FFFH; 0800H + 11-bit immediate address
addr5	Pointer into call table, 0040H-007EH: or 8040H-807EH, 5 bit immediate data or label
A	A register (8-bit accumulator)
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R15	Register 0-15
AX	Register pair AX (16-bit accumulator)
BC	Register pair BC
VP	Register pair VP
UP	Register pair UP (user stack pointer)
DE	Register pair DE
HL	Register pair HL
RP0-RP7	Register pair 0-7
PC	Program counter
SP	Stack pointer
PSW	Program status word
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
P/V	Parity/overflow flag
S	Sign flag
SUB	Subtract flag
TPF	Table position flag
RBS	Register bank select flag
RSS	Register set select flag
IE	Interrupt enable flag
EOS	End of software interrupt flag
STBC	Standby control register
WDM	Watchdog timer mode register

Symbol	Meaning
/	Logical complement
( )	Contents of the location whose address is within ( ); (+) and (–) indicate that the address is incremented or decremented after it is used.
(( ))	Contents of the memory location defined by the contents of the location defined by the quantity within the (( )).
XXH	Hexadecimal number
XH, XL	High-order 8 bits and low-order 8 bits of X

Flag Indicators

Symbol	Meaning
(blank)	No change
0	Cleared to 0
1	Set to 1
X	Set or cleared according to result
P	Parity of result
V	Arithmetic overflow
U	Undefined
R	Restored from saved PSW

Execution Times of Memory Reference Instructions: Number of Processor States

		Memory Reference Mode			
Instruction		Register Indirect	Base Index	Base	Index
MOV	A, mem	5	6	6	6
	mem,A				
XCH	A, mem	7	8	8	8
	mem,A				
ADD, ADDC, SUB, SUBC, AND, OR, XOR	A, mem	6	7	7	7
	mem,A	7	8	8	8
CMP	A, mem	6	7	7	7
	mem,A				

Memory Addressing Modes

mem	mod	1 0110	1 0111	0 0110	0 1010
		Register Indirect	Base Index	Base	Index
0 0 0		[DE+]*	[DE+ A]	[DE+ byte]	word [DE]
0 0 1		[HL+]*	[HL+ A]	[SP+ byte]	word [A]
0 1 0		[DE–]*	[DE+ B]	[HL+ byte]	word [HL]
0 1 1		[HL–]*	[HL+ B]	[UP+ byte]	word [B]
1 0 0		[DE]*	[VP+ DE]	[VP+ byte]	—
1 0 1		[HL]*	[VP+ HL]	—	—
1 1 0		[VP]	—	—	—
1 1 1		[UP]	—	—	—

*1-byte instructions: defined by special opcode and mem only.

General Register Designation r, r1

R_3	R_2	R_1	R_0	Reg		
0	0	0	0	R0		
0	0	0	1	R1		
0	0	1	0	R2		
0	0	1	1	R3	r1	
0	1	0	0	R4		
0	1	0	1	R5		
0	1	1	0	R6		
0	1	1	1	R7		r
1	0	0	0	R8		
1	0	0	1	R9		
1	0	1	0	R10		
1	0	1	1	R11		
1	1	0	0	R12		
1	1	0	1	R13		
1	1	1	0	R14		
1	1	1	1	R15		

r2

C	Reg
0	C
1	B

rp

P_2	P_1	P_0	Reg Pair
0	0	0	RP0
0	0	1	RP1
0	1	0	RP2
0	1	1	RP3
1	0	0	RP4
1	0	1	RP5
1	1	0	RP6
1	1	1	RP7

rp1

Q_2	Q_1	Q_0	Reg Pair
0	0	0	RP0
0	0	1	RP4
0	1	0	RP1
0	1	1	RP5
1	0	0	RP2
1	0	1	RP6
1	1	0	RP3
1	1	1	RP7

rp2

S_1	S_0	Reg Pair
0	0	VP
0	1	UP
1	0	DE
1	1	HL

Instructions

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Data Transfer												
MOV	r1,#byte	r1 ← byte	3	3	2							1 0 1 1 1 R_2 R_1 R_0 Data
	saddr,#byte	(saddr) ← byte	3/4	0	3							0 0 1 1 1 0 1 0 Saddr-offset Data
	sfr,#byte	sfr ← byte	4	0	3							0 0 1 0 1 0 1 1 Sfr-offset Data
	r,r1	r ← r1	3	3	2							0 0 1 0 0 1 0 0 R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,r1	A ← r1	3	3	1							1 1 0 1 0 R_2 R_1 R_0
	A,saddr	A ← (saddr)	3/4	1	2							0 0 1 0 0 0 0 0 Saddr-offset
	saddr,A	(saddr) ← A	3/4	0	2							0 0 1 0 0 0 1 0 Saddr-offset
	saddr,saddr	(saddr) ← (saddr)	4/6	0	3							0 0 1 1 1 0 0 0 Saddr-offset Saddr-offset
	A,sfr	A ← sfr	4	1	2							0 0 0 1 0 0 0 0 Sfr-offset
	sfr,A	sfr ← A	4	0	2							0 0 0 1 0 0 1 0 Sfr-offset
	A,mem*	A ← (mem)	5	3	1							0 1 0 1 1 mem
	A,mem	A ← (mem)	5-6	3-4	2-4							0 0 0 mod 0 mem 0 0 0 0 Low offset High offset
	mem,A*	(mem) ← A	5	2	1							0 1 0 1 0 mem
	mem,A	(mem) ← A	5-6	2	2-4							0 0 0 mod 1 mem 0 0 0 0 Low offset High offset

*When mem is [DE], [HL], [DE+], [DE–], [HL+], or [HL–]

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**States**	**States**	**Bytes**	**S**	**Z**	**AC**	**P/V**	**SUB**	**CY**	**Bytes B1 thru B5**
Data Transfer (cont)												
MOV (cont)	A,[saddrp]	A ← ((saddrp))	5/6	1	2							0 0 0 1 1 0 0 0
												Saddr-offset
	[saddrp],A	((saddrp)) ← A	4/5	0	2							0 0 0 1 1 0 0 1
												Saddr-offset
	A,!addr16	A ← (addr16)	5	3	4							0 0 0 0 1 0 0 1
												1 1 1 1 0 0 0 0
												Low addr
												High addr
	!addr16,A	(addr16) ← A	4	2	4							0 0 0 0 1 0 0 1
												1 1 1 1 0 0 0 1
												Low addr
												High addr
	PSWL,#byte	PSWL ← byte	4	0	3	X	X	X	X	X	X	0 0 1 0 1 0 1 1
												1 1 1 1 1 1 1 0
												Data
	PSWH,#byte	PSWH ← byte	4	0	3							0 0 1 0 1 0 1 1
												1 1 1 1 1 1 1 1
												Data
	PSWL,A	PSWL ← A	4	0	2	X	X	X	X	X	X	0 0 0 1 0 0 1 0
												1 1 1 1 1 1 1 0
	PSWH,A	PSWH ← A	4	0	2							0 0 0 1 0 0 1 0
												1 1 1 1 1 1 1 1
	A,PSWL	A ← PSWL	4	1	2							0 0 0 1 0 0 0 0
												1 1 1 1 1 1 1 0
	A,PSWH	A ← PSWH	4	1	2							0 0 0 1 0 0 0 0
												1 1 1 1 1 1 1 1
XCH	A,r1	A ←→ r1	4	4	1							1 1 0 1 1 R_2 R_1 R_0
	r,r1	r ←→ r1	4	4	2							0 0 1 0 0 1 0 1
												R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,mem	A ←→ (mem)	7-8	3-4	2-4							0 0 0 mod
												0 mem 0 1 0 0
												Low offset
												High offset
	A,saddr	A ←→ (saddr)	4/6	0	2							0 0 1 0 0 0 0 1
												Saddr-offset
	A,sfr	A ←→ sfr	8	3	3							0 0 0 0 0 0 0 1
												0 0 1 0 0 0 0 1
												Sfr-offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Data Transfer (cont)												
XCH (cont)	A,[saddrp]	A ⟷ ((saddrp))	6/7	0	2							0 0 1 0 0 0 1 1 Saddr-offset
	saddr,saddr	(saddr) ⟷ (saddr)	8/12	0	3							0 0 1 1 1 0 0 1 Saddr-offset Saddr-offset
MOVW	rp1,#word	rp1 ← word	3	3	3							0 1 1 0 0 Q_2 Q_1 Q_0 Low byte High byte
	saddrp,#word	(saddrp) ← word	3/4	0	4							0 0 0 0 1 1 0 0 Saddr-offset Low byte High byte
	sfrp,#word	sfrp ← word	4	0	4							0 0 0 0 1 0 1 1 Sfr-offset Low byte High byte
	rp,rp1	rp ← rp1	3	3	2							0 0 1 0 0 1 0 0 P_2 P_1 P_0 0 1 Q_2 Q_1 Q_0
	AX,saddrp	AX ← (saddrp)	3/4	1	2							0 0 0 1 1 1 0 0 Saddr-offset
	saddrp,AX	(saddrp) ← AX	3/4	0	2							0 0 0 1 1 0 1 0 Saddr-offset
	saddrp,saddrp	(saddrp) ← (saddrp)	4/6	0	3							0 0 1 1 1 1 0 0 Saddr-offset Saddr-offset
	AX,sfrp	AX ← sfrp	4	1	2							0 0 0 1 0 0 0 1 Sfr-offset
	sfrp,AX	sfrp ← AX	4	0	2							0 0 0 1 0 0 1 1 Sfr-offset
	rpl,!addr16	rpl ← (addr16)	10	6	4							0 0 0 0 1 0 0 1 1 0 0 0 0 Q_2 Q_1 Q_0 Low Addr High Addr
	!addr16,rpl	(addr16) ← rpl	8	4	4							0 0 0 0 1 0 0 1 1 0 0 1 0 Q_2 Q_1 Q_0 Low Addr High Addr

Instructions (cont)

Mnemonic	Operand	Operation	States	Idle States	Bytes	Flags S	Z	AC	P/V	SUB	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
Data Transfer (cont)												
XCHW	AX,saddrp	AX ⟷ (saddrp)	4/6	0	2							0 0 0 1 1 0 1 1
												Saddr-offset
	AX,sfrp	AX ⟷ sfrp	9	3	3							0 0 0 0 0 0 0 1
												0 0 0 1 1 0 1 1
												Sfr-offset
	saddrp, saddrp	(saddrp) ⟷ (saddrp)	8/12	0	3							0 0 1 0 1 0 1 0
												Saddr-offset
												Saddr-offset
	rp,rp1	rp ⟷ rp1	5	5	2							0 0 1 0 0 1 0 1
												P_2 P_1 P_0 0 1 Q_2 Q_1 Q_0
8-Bit Operation												
ADD	A,#byte	A, CY ← A + byte	3	3	2	X	X	X	V	0	X	1 0 1 0 1 0 0 0
												Data
	saddr,#byte	(saddr), CY ← (saddr) + byte	5/7	0	3	X	X	X	V	0	X	0 1 1 0 1 0 0 0
												Saddr-offset
												Data
	sfr,#byte	sfr, CY ← sfr + byte	10	3	4	X	X	X	V	0	X	0 0 0 0 0 0 0 1
												0 1 1 0 1 0 0 0
												Sfr-offset
												Data
	r,r1	r, CY ← r + r1	3	3	2	X	X	X	V	0	X	1 0 0 0 1 0 0 0
												R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A, CY ← A + (saddr)	3/4	1	2	X	X	X	V	0	X	1 0 0 1 1 0 0 0
												Saddr-offset
	A,sfr	A, CY ← A + sfr	7	4	3	X	X	X	V	0	X	0 0 0 0 0 0 0 1
												1 0 0 1 1 0 0 0
												Sfr-offset
	saddr,saddr	(saddr), CY ← (saddr) + (saddr)	6/9	0	3	X	X	X	V	0	X	0 1 1 1 1 0 0 0
												Saddr-offset
												Saddr-offset
	A,mem	A, CY ← A + (mem)	6-7	4-5	2-4	X	X	X	V	0	X	0 0 0 mod
												0 mem 1 0 0 0
												Low offset
												High offset
	mem,A	(mem), CY ← (mem) + A	7-8	2-3	2-4	X	X	X	V	0	X	0 0 0 mod
												1 mem 1 0 0 0
												Low offset
												High offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
8-Bit Operation (cont)												
ADDC	A,#byte	A, CY ← A + byte + CY	3	3	2	X	X	X	V	0	X	1 0 1 0 1 0 0 1 Data
	saddr,#byte	(saddr), CY ← (saddr) + byte + CY	5/7	0	3	X	X	X	V	0	X	0 1 1 0 1 0 0 1 Saddr-offset Data
	sfr,#byte	sfr, CY ← sfr + byte + CY	10	3	4	X	X	X	V	0	X	0 0 0 0 0 0 0 1 0 1 1 0 1 0 0 1 Sfr-offset Data
	r,r1	r, CY ← r + r1 + CY	3	3	2	X	X	X	V	0	X	1 0 0 0 1 0 0 1 R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A, CY ← A + (saddr) + CY	3/4	1	2	X	X	X	V	0	X	1 0 0 1 1 0 0 1 Saddr-offset
	A,sfr	A, CY ← A + sfr + CY	7	4	3	X	X	X	V	0	X	0 0 0 0 0 0 0 1 1 0 0 1 1 0 0 1 Sfr-offset
	saddr,saddr	(saddr), CY ← (saddr) + (saddr) + CY	6/9	0	3	X	X	X	V	0	X	0 1 1 1 1 0 0 1 Saddr-offset Saddr-offset
	A,mem	A, CY ← A + (mem) + CY	6-7	4-5	2-4	X	X	X	V	0	X	0 0 0 mod 0 mem 1 0 0 1 Low offset High offset
	mem,A	(mem), CY ← (mem) + A + CY	7-8	2-3	2-4	X	X	X	V	0	X	0 0 0 mod 1 mem 1 0 0 1 Low offset High offset
SUB	A,#byte	A, CY ← A − byte	3	3	2	X	X	X	V	1	X	1 0 1 0 1 0 1 0 Data
	saddr,#byte	(saddr), CY ← (saddr) − byte	5/7	0	3	X	X	X	V	1	X	0 1 1 0 1 0 1 0 Saddr-offset Data
	sfr,#byte	sfr, CY ← sfr − byte	10	3	4	X	X	X	V	1	X	0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 0 Sfr-offset Data
	r,r1	r, CY ← r − r1	3	3	2	X	X	X	V	1	X	1 0 0 0 1 0 1 0 R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A, CY ← A − (saddr)	3/4	1	2	X	X	X	V	1	X	1 0 0 1 1 0 1 0 Saddr-offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
8-Bit Operation (cont)												
SUB (cont)	A,sfr	A, CY ← A – sfr	7	4	3	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												1 0 0 1 1 0 1 0
												Sfr-offset
	saddr,saddr	(saddr), CY ← (saddr) – (saddr)	6/9	0	3	X	X	X	V	1	X	0 1 1 1 1 0 1 0
												Saddr-offset
												Saddr-offset
	A,mem	A, CY ← A – (mem)	6-7	4-5	2-4	X	X	X	V	1	X	0 0 0 mod
												0 mem 1 0 1 0
												Low offset
												High offset
	mem, A	(mem), CY ← (mem) – A	7-8	2-3	2-4	X	X	X	V	1	X	0 0 0 mod
												1 mem 1 0 1 0
												Low offset
												High offset
SUBC	A,#byte	A, CY ← A – byte – CY	3	3	2	X	X	X	V	1	X	1 0 1 0 1 0 1 1
												Data
	saddr,#byte	(saddr), CY ← (saddr) – byte – CY	5/7	0	3	X	X	X	V	1	X	0 1 1 0 1 0 1 1
												Saddr-offset
												Data
	sfr,#byte	sfr, CY ← sfr – byte – CY	10	3	4	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												0 1 1 0 1 0 1 1
												Sfr-offset
												Data
	r,r1	r, CY ← r – r1 – CY	3	3	2	X	X	X	V	1	X	1 0 0 0 1 0 1 1
												R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A, CY ← A – (saddr) – CY	3/4	1	2	X	X	X	V	1	X	1 0 0 1 1 0 1 1
												Saddr-offset
	A,sfr	A, CY ← A – sfr – CY	7	4	3	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												1 0 0 1 1 0 1 1
												Sfr-offset
	saddr,saddr	(saddr), CY ← (saddr) – (saddr) – CY	6/9	0	3	X	X	X	V	1	X	0 1 1 1 1 0 1 1
												Saddr-offset
												Saddr-offset
	A,mem	A, CY ← A – (mem) – CY	6-7	4-5	2-4	X	X	X	V	1	X	0 0 0 mod
												0 mem 1 0 1 1
												Low offset
												High offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	**Operand**	**Operation**	**States**	**States**	**Bytes**	**S**	**Z**	**AC**	**P/V**	**SUB**	**CY**	**Bytes B1 thru B5**
8-Bit Operation (cont)												
SUBC (cont)	mem, A	(mem), CY ← (mem) − A − CY	7-8	2-3	2-4	X	X	X	V	1	X	0 0 0 mod
												1 mem 1 0 1 1
												Low offset
												High offset
AND	A,#byte	A ← A ∧ byte	3	3	2	X	X		P	0		1 0 1 0 1 1 0 0
												Data
	saddr,#byte	(saddr) ← (saddr) ∧ byte	5/7	0	3	X	X		P	0		0 1 1 0 1 1 0 0
												Saddr-offset
												Data
	sfr,#byte	sfr ← sfr ∧ byte	10	3	4	X	X		P	0		0 0 0 0 0 0 0 1
												0 1 1 0 1 1 0 0
												Sfr-offset
												Data
	r,r1	r ← r ∧ r1	3	3	2	X	X		P	0		1 0 0 0 1 1 0 0
												R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A ← A ∧ (saddr)	3/4	1	2	X	X		P	0		1 0 0 1 1 1 0 0
												Saddr-offset
	A,sfr	A ← A ∧ sfr	7	4	3	X	X		P	0		0 0 0 0 0 0 0 1
												1 0 0 1 1 1 0 0
												Sfr-offset
	saddr,saddr	(saddr) ← (saddr) ∧ (saddr)	6/9	0	3	X	X		P	0		0 1 1 1 1 1 0 0
												Saddr-offset
												Saddr-offset
	A,mem	A ← A ∧ (mem)	6-7	4-5	2-4	X	X		P	0		0 0 0 mod
												0 mem 1 1 0 0
												Low offset
												High offset
	mem,A	(mem) ← (mem) ∧ A	7-8	2-3	2-4	X	X		P	0		0 0 0 mod
												1 mem 1 1 0 0
												Low offset
												High offset
OR	A,#byte	A ← A V byte	3	3	2	X	X		P	0		1 0 1 0 1 1 1 0
												Data
	saddr,#byte	(saddr) ← (saddr) V byte	5/7	0	3	X	X		P	0		0 1 1 0 1 1 1 0
												Saddr-offset
												Data

Instructions (cont)

8-Bit Operation (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
OR (cont)	sfr,#byte	sfr ← sfr V byte	10	3	4	X	X		P	0		0 0 0 0 0 0 0 1
												0 1 1 0 1 1 1 0
												Sfr-offset
												Data
	r,r1	r ← r V r1	3	3	2	X	X		P	0		1 0 0 0 1 1 1 0
												R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A ← A V (saddr)	3/4	1	2	X	X		P	0		1 0 0 1 1 1 1 0
												Saddr-offset
	A,sfr	A ← A V sfr	7	4	3	X	X		P	0		0 0 0 0 0 0 0 1
												1 0 0 1 1 1 1 0
												Sfr-offset
	saddr,saddr	(saddr) ← (saddr) V (saddr)	6/9	0	3	X	X		P	0		0 1 1 1 1 1 1 0
												Saddr-offset
												Saddr-offset
	A,mem	A ← A V (mem)	6-7	4-5	2-4	X	X		P	0		0 0 0 mod
												0 mem 1 1 1 0
												Low offset
												High offset
	mem,A	(mem) ← (mem) V A	7-8	2-3	2-4	X	X		P	0		0 0 0 mod
												1 mem 1 1 1 0
												Low offset
												High offset
XOR	A,#byte	A ← ⊻ byte	3	3	2	X	X		P	0		1 0 1 0 1 1 0 1
												Data
	saddr,#byte	(saddr) ← (saddr) ⊻ byte	5/7	0	3	X	X		P	0		0 1 1 0 1 1 0 1
												Saddr-offset
												Data
	sfr,#byte	sfr ← sfr ⊻ byte	10	3	4	X	X		P	0		0 0 0 0 0 0 0 1
												0 1 1 0 1 1 0 1
												Sfr-offset
												Data
	r,r1	r ← r ⊻ r1	3	3	2	X	X		P	0		1 0 0 0 1 1 0 1
												R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A ← A ⊻ (saddr)	3/4	1	2	X	X		P	0		1 0 0 1 1 1 0 1
												Saddr-offset
	A,sfr	A ← A ⊻ sfr	7	4	3	X	X		P	0		0 0 0 0 0 0 0 1
												1 0 0 1 1 1 0 1
												Sfr-offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
8-Bit Operation (cont)												
XOR (cont)	saddr,saddr	(saddr) ← (saddr) ∀ (saddr)	6/9	0	3	X	X		P	0		0 1 1 1 1 1 0 1 Saddr-offset Saddr-offset
	A,mem	A ← A ∀ (mem)	6-7	4-5	2-4	X	X		P	0		0 0 0 mod 0 mem 1 1 0 1 Low offset High offset
	mem,A	(mem) ← (mem) ∀ A	7-8	2-3	2-4	X	X		P	0		0 0 0 mod 1 mem 1 1 0 1 Low offset High offset
CMP	A,#byte	A − byte	3	3	2	X	X	X	V	1	X	1 0 1 0 1 1 1 1 Data
	saddr,#byte	(saddr) − byte	5/7	1	3	X	X	X	V	1	X	0 1 1 0 1 1 1 1 Saddr-offset Data
	sfr,#byte	sfr − byte	10	4	4	X	X	X	V	1	X	0 0 0 0 0 0 0 1 0 1 1 0 1 1 1 1 Sfr-offset Data
	r,r1	r − r1	3	3	2	X	X	X	V	1	X	1 0 0 0 1 1 1 1 R_3 R_2 R_1 R_0 0 R_2 R_1 R_0
	A,saddr	A − (saddr)	3/4	1	2	X	X	X	V	1	X	1 0 0 1 1 1 1 1 Saddr-offset
	A,sfr	A − sfr	7	4	3	X	X	X	V	1	X	0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 1 Sfr-offset
	saddr,saddr	(saddr) − (saddr)	6/8	1	3	X	X	X	V	1	X	0 1 1 1 1 1 1 1 Saddr-offset Saddr-offset
	A,mem	A − (mem)	6-7	4-5	2-4	X	X	X	V	1	X	0 0 0 mod 0 mem 1 1 1 1 Low offset High offset
	mem,A	(mem) − A	6-7	3-4	2-4	X	X	X	V	1	X	0 0 0 mod 1 mem 1 1 1 1 Low offset High offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
16-Bit Operation												
ADDW	AX,#word	AX, CY ← AX + word	4	4	3	X	X	X	V	0	X	0 0 1 0 1 1 0 1
												Low byte
												High byte
	saddrp,#word	(saddrp), CY ← (saddrp) + word	5/7	0	4	X	X	X	V	0	X	0 0 0 0 1 1 0 1
												Saddr-offset
												Low byte
												High byte
	sfrp,#word	sfrp, CY ← sfrp + word	10	3	5	X	X	X	V	0	X	0 0 0 0 0 0 0 1
												0 0 0 0 1 1 0 1
												Sfr-offset
												Low byte
												High byte
	rp,rp1	rp, CY ← rp + rp1	4	4	2	X	X	X	V	0	X	1 0 0 0 1 0 0 0
												P_2 P_1 P_0 0 1 Q_2 Q_1 Q_0
	AX,saddrp	AX, CY ← AX + (saddrp)	4/5	2	2	X	X	X	V	0	X	0 0 0 1 1 1 0 1
												Saddr-offset
	AX,sfrp	AX, CY ← AX + sfrp	8	5	3	X	X	X	V	0	X	0 0 0 0 0 0 0 1
												0 0 0 1 1 1 0 1
												Sft-offset
	saddrp,saddrp	(saddrp), CY ← (saddrp) + (saddrp)	6/9	0	3	X	X	X	V	0	X	0 0 1 1 1 1 0 1
												Saddr-offset
												Saddr-offset
SUBW	AX,#word	AX, CY ← AX – word	4	3	3	X	X	X	V	1	X	0 0 1 0 1 1 1 0
												Low byte
												High byte
	saddrp,#word	(saddrp), CY ← (saddrp) – word	5/7	0	4	X	X	X	V	1	X	0 0 0 0 1 1 1 0
												Saddr-offset
												Low byte
												High byte

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
16-Bit Operation (cont)												
SUBW (cont)	sfrp,#word	sfrp, CY ← sfrp – word	10	3	5	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												0 0 0 0 1 1 1 0
												Sfr-offset
												Low byte
												High byte
	rp,rp1	rp, CY ← rp – rp1	4	4	2	X	X	X	V	1	X	1 0 0 0 1 0 1 0
												P_2 P_1 P_0 0 1 Q_2 Q_1 Q_0
	AX,saddrp	AX, CY ← AX – (saddrp)	4/5	2	2	X	X	X	V	1	X	0 0 0 1 1 1 1 0
												Saddr-offset
	AX,sfrp	AX, CY ← AX – sfrp	8	5	3	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												0 0 0 1 1 1 1 0
												Sfr-offset
	saddrp,saddrp	(saddrp), CY ← (saddrp) – (saddrp)	6/9	0	3	X	X	X	V	1	X	0 0 1 1 1 1 1 0
												Saddr-offset
												Saddr-offset
CMPW	AX,#word	AX – word	4	3	3	X	X	X	V	1	X	0 0 1 0 1 1 1 1
												Low byte
												High byte
	saddrp,#word	(saddrp) – word	4/5	1	4	X	X	X	V	1	X	0 0 0 0 1 1 1 1
												Saddr-offset
												Low byte
												High byte
	sfrp,#word	sfrp – word	8	4	5	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												0 0 0 0 1 1 1 1
												Sfr-offset
												Low byte
												High byte
	rp,rp1	rp – rp1	4	4	2	X	X	X	V	1	X	1 0 0 0 1 1 1 1
												P_2 P_1 P_0 0 1 Q_2 Q_1 Q_0
	AX,saddrp	AX – (saddrp)	4/5	1	2	X	X	X	V	1	X	0 0 0 1 1 1 1 1
												Saddr-offset
	AX,sfrp	AX – sfrp	8	4	3	X	X	X	V	1	X	0 0 0 0 0 0 0 1
												0 0 0 1 1 1 1 1
												Sfr-offset
	saddrp,saddrp	(saddrp) – (saddrp)	5/7	1	3	X	X	X	V	1	X	0 0 1 1 1 1 1 1
												Saddr-offset
												Saddr-offset

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Multiplication/Division												
MULU	r1	AX ← A x r1	18	18	2							0 0 0 0 0 1 0 1
												0 0 0 0 1 R_2 R_1 R_0
DIVUW	r1	AX (Quotient), r1 (Remainder) ← AX ÷ r1	26	26	2							0 0 0 0 0 1 0 1
												0 0 0 1 1 R_2 R_1 R_0
MULUW	rp1	AX (High-order 16 bits), rp1 (Low-order 16 bits) ← AX x rp1	27	27	2							0 0 0 0 0 1 0 1
												0 0 1 0 1 Q_2 Q_1 Q_0
DIVUX	rp1	AXDE (Quotient), rp1 (Remainder) ← AXDE ÷ rp1	50	50	2							0 0 0 0 0 1 0 1
												1 1 1 0 1 Q_2 Q_1 Q_0
Increment/Decrement												
INC	r1	r1 ← r1 + 1	3	3	1	X	X	X	V	0		1 1 0 0 0 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) + 1	4/6	0	2	X	X	X	V	0		0 0 1 0 0 1 1 0
												Saddr-offset
DEC	r1	r1 ← r1 − 1	3	3	1	X	X	X	V	1		1 1 0 0 1 R_2 R_1 R_0
	saddr	(saddr) ← (saddr) − 1	4/6	0	2	X	X	X	V	1		0 0 1 0 0 1 1 1
												Saddr-offset
INCW	rp2	rp2 ← rp2 + 1	3	3	1							0 1 0 0 0 1 S_1 S_0
	saddrp	(saddrp) ← (saddrp) + 1	6/8	2	3							0 0 0 0 0 1 1 1
												1 1 1 0 1 0 0 0
												Saddr-offset
DECW	rp2	rp2 ← rp2 − 1	3	3	1							0 1 0 0 1 1 S_1 S_0
	saddrp	(saddrp) ← (saddrp) − 1	6/8	2	3							0 0 0 0 0 1 1 1
												1 1 1 0 1 0 0 1
												Saddr-offset
Shift and Rotate												
ROR	r1,n	(CY,$r1_7$ ← $r1_0$, $r1_{m-1}$ ← $r1_m$) x n	4+3n	4+3n	2				P	0	X	0 0 1 1 0 0 0 0
												0 1 N_2 N_1 N_0 R_2 R_1 R_0
ROL	r1,n	(CY, $r1_0$ ← $r1_7$, $r1_{m+1}$ ← $r1_m$) x n	4+3n	4+3n	2				P	0	X	0 0 1 1 0 0 0 1
												0 1 N_2 N_1 N_0 R_2 R_1 R_0
RORC	r1,n	(CY ← $r1_0$, $r1_7$ ← CY, $r1_{m-1}$ ← $r1_m$) x n	4+3n	4+3n	2				P	0	X	0 0 1 1 0 0 0 0
												0 0 N_2 N_1 N_0 R_2 R_1 R_0
ROLC	r1,n	(CY ← $r1_7$, $r1_0$ ← CY, $r1_{m+1}$ ← $r1_m$) x n	4+3n	4+3n	2				P	0	X	0 0 1 1 0 0 0 1
												0 0 N_2 N_1 N_0 R_2 R_1 R_0

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Shift and Rotate (cont)												
SHR	r1,n	(CY ← $r1_0$, $r1_7$ ← 0, $r1_{m-1}$ ← $r1_m$) x n	4+3n	4+3n	2	X	X	0	P	0	X	0 0 1 1 0 0 0 0 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHL	r1,n	(CY ← $r1_7$, $r1_0$ ← 0, $r1_{m+1}$ ← $r1_m$) x n	4+3n	4+3n	2	X	X	0	P	0	X	0 0 1 1 0 0 0 1 1 0 N_2 N_1 N_0 R_2 R_1 R_0
SHRW	rp1,n	(CY ← $rp1_0$, $rp1_{15}$ ← 0, $rp1_{m-1}$ ← $rp1_m$) x n	4+3n	4+3n	2	X	X	0	P	0	X	0 0 1 1 0 0 0 0 1 1 N_2 N_1 N_0 Q_2 Q_1 Q_0
SHLW	rp1,n	(CY ← $rp1_{15}$, $rp1_0$ ← 0, $rp1_{m+1}$ ← $rp1_m$) x n	4+3n	4+3n	2	X	X	0	P	0	X	0 0 1 1 0 0 0 1 1 1 N_2 N_1 N_0 Q_2 Q_1 Q_0
ROR4	[rp1]	A_{3-0} ← $(rp1)_{3-0}$, $(rp1)_{7-4}$ ← A_{3-0}, $(rp1)_{3-0}$ ← $(rp1)_{7-4}$	7	3	2							0 0 0 0 0 1 0 1 1 0 0 0 1 Q_2 Q_1 Q_0
ROL4	[rp1]	A_{3-0} ← $(rp1)_{7-4}$, $(rp1)_{3-0}$ ← A_{3-0}, $(rp1)_{7-4}$ ← $(rp1)_{3-0}$	7	3	2							0 0 0 0 0 1 0 1 1 0 0 1 1 Q_2 Q_1 Q_0
BCD Adjustment												
ADJ4		Decimal adjust accumulator	3	3	1	X	X	X	P		X	0 0 0 0 0 1 0 0
Bit Manipulation												
MOV1	CY,saddr.bit	CY ← (saddr.bit)	6/7	4	3						X	0 0 0 0 1 0 0 0 0 0 0 0 0 B_2 B_1 B_0 Saddr-offset
	CY,sfr.bit	CY ← sfr.bit	7	4	3						X	0 0 0 0 1 0 0 0 0 0 0 0 1 B_2 B_1 B_0 Sfr-offset
	CY,A.bit	CY ← A.bit	6	6	2						X	0 0 0 0 0 0 1 1 0 0 0 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← X.bit	6	6	2						X	0 0 0 0 0 0 1 1 0 0 0 0 0 B_2 B_1 B_0
	CY,PSWL.bit	CY ← PSWH.bit	6	6	2						X	0 0 0 0 0 0 1 0 0 0 0 0 1 B_2 B_1 B_0
	CY,PSWL.bit	CY ← PSWL.bit	6	6	2						X	0 0 0 0 0 0 1 0 0 0 0 0 0 B_2 B_1 B_0
	saddr.bit,CY	(saddr.bit) ← CY	7/8	3	3							0 0 0 0 1 0 0 0 0 0 0 1 0 B_2 B_1 B_0 Saddr-offset
	sfr.bit,CY	sfr.bit ← CY	8	3	3							0 0 0 0 1 0 0 0 0 0 0 1 1 B_2 B_1 B_0 Sfr-offset
	A.bit,CY	A.bit ← CY	8	8	2							0 0 0 0 0 0 1 1 0 0 0 1 1 B_2 B_1 B_0

7

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Bit Manipulation (cont)												
MOV1 (cont)	X.bit,CY	X.bit ← CY	8	8	2							0 0 0 0 0 0 1 1
												0 0 0 1 0 B_2 B_1 B_0
	PSWH.bit,CY	PSW_H.bit ← CY	9	9	2							0 0 0 0 0 0 1 0
												0 0 0 1 1 B_2 B_1 B_0
	PSWL.bit,CY	PSW_L.bit ← CY	9	9	2	X	X	X	X	X		0 0 0 0 0 0 1 0
												0 0 0 1 0 B_2 B_1 B_0
AND1	CY,saddr.bit	CY ← CY ∧ (saddr.bit)	6/7	4	3						X	0 0 0 0 1 0 0 0
												0 0 1 0 0 B_2 B_1 B_0
												Saddr-offset
	CY,/saddr.bit	CY ← CY ∧ ($\overline{\text{saddr.bit}}$)	6/7	4	3						X	0 0 0 0 1 0 0 0
												0 0 1 1 0 B_2 B_1 B_0
												Saddr-offset
	CY,sfr.bit	CY ← CY ∧ sfr.bit	7	4	3						X	0 0 0 0 1 0 0 0
												0 0 1 0 1 B_2 B_1 B_0
												Sfr-offset
	CY,/sfr.bit	CY ← CY ∧ $\overline{\text{sfr.bit}}$	7	4	3						X	0 0 0 0 1 0 0 0
												0 0 1 1 1 B_2 B_1 B_0
												Sfr-offset
	CY,A.bit	CY ← CY ∧ A.bit	6	6	2						X	0 0 0 0 0 0 1 1
												0 0 1 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY ∧ $\overline{\text{A.bit}}$	6	6	2						X	0 0 0 0 0 0 1 1
												0 0 1 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY ∧ X.bit	6	6	2						X	0 0 0 0 0 0 1 1
												0 0 1 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY ∧ $\overline{\text{X.bit}}$	6	6	2						X	0 0 0 0 0 0 1 1
												0 0 1 1 0 B_2 B_1 B_0
	CY,PSWH.bit	CY ← CY ∧ PSWH.bit	6	6	2						X	0 0 0 0 0 0 1 0
												0 0 1 0 1 B_2 B_1 B_0
	CY,/PSWH.bit	CY ← CY ∧ $\overline{\text{PSWH.bit}}$	6	6	2						X	0 0 0 0 0 0 1 0
												0 0 1 1 1 B_2 B_1 B_0
	CY,PSWL.bit	CY ← CY ∧ PSWL.bit	6	6	2						X	0 0 0 0 0 0 1 0
												0 0 1 0 0 B_2 B_1 B_0
	CY,/PSWL.bit	CY ← CY ∧ $\overline{\text{PSWL.bit}}$	6	6	2						X	0 0 0 0 0 0 1 0
												0 0 1 1 0 B_2 B_1 B_0
OR1	CY,saddr.bit	CY ← CY V (saddr.bit)	6/7	4	3						X	0 0 0 0 1 0 0 0
												0 1 0 0 0 B_2 B_1 B_0
												Saddr-offset
	CY,/saddr.bit	CY ← CY V ($\overline{\text{saddr.bit}}$)	6/7	4	3						X	0 0 0 0 1 0 0 0
												0 1 0 1 0 B_2 B_1 B_0
												Saddr-offset

Instructions (cont)

Mnemonic	Operand	Operation	States	Idle States	Bytes	Flags S	Z	AC	P/V	SUB	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
Bit Manipulation (cont)												
OR1 (cont)	CY,sfr.bit	CY ← CY V sfr.bit	7	4	3						X	0 0 0 0 1 0 0 0 0 1 0 0 1 B_2 B_1 B_0 Sfr-offset
	CY,/sfr.bit	CY ← CY V $\overline{\text{sfr.bit}}$	7	4	3						X	0 0 0 0 1 0 0 0 0 1 0 1 1 B_2 B_1 B_0 Sfr-offset
	CY,A.bit	CY ← CY V A.bit	6	6	2						X	0 0 0 0 0 0 1 1 0 1 0 0 1 B_2 B_1 B_0
	CY,/A.bit	CY ← CY V $\overline{\text{A.bit}}$	6	6	2						X	0 0 0 0 0 0 1 1 0 1 0 1 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY V X.bit	6	6	2						X	0 0 0 0 0 0 1 1 0 1 0 0 0 B_2 B_1 B_0
	CY,/X.bit	CY ← CY V $\overline{\text{X.bit}}$	6	6	2						X	0 0 0 0 0 0 1 1 0 1 0 1 0 B_2 B_1 B_0
	CY,PSWH.bit	CY ← CY V PSWH.bit	6	6	2						X	0 0 0 0 0 0 1 0 0 1 0 0 1 B_2 B_1 B_0
	CY,/PSWH.bit	CY ← CY V $\overline{\text{PSWH.bit}}$	6	6	2						X	0 0 0 0 0 0 1 0 0 1 0 1 1 B_2 B_1 B_0
	CY,PSWL.bit	CY ← CY V PSWL.bit	6	6	2						X	0 0 0 0 0 0 1 0 0 1 0 0 0 B_2 B_1 B_0
	CY,/PSWL.bit	CY ← CY V $\overline{\text{PSWL.bit}}$	6	6	2						X	0 0 0 0 0 0 1 0 0 1 0 1 0 B_2 B_1 B_0
XOR1	CY,saddr.bit	CY ← CY $\veebar$ (saddr.bit)	6/7	4	3						X	0 0 0 0 1 0 0 0 0 1 1 0 0 B_2 B_1 B_0 Saddr-offset
	CY,sfr.bit	CY ← CY $\veebar$ sfr.bit	7	4	3						X	0 0 0 0 1 0 0 0 0 1 1 0 1 B_2 B_1 B_0 Sfr-offset
	CY,A.bit	CY ← CY $\veebar$ A.bit	6	6	2						X	0 0 0 0 0 0 1 1 0 1 1 0 1 B_2 B_1 B_0
	CY,X.bit	CY ← CY $\veebar$ X.bit	6	6	2						X	0 0 0 0 0 0 1 1 0 1 1 0 0 B_2 B_1 B_0
	CY,PSWH.bit	CY ← CY $\veebar$ PSWH.bit	6	6	2						X	0 0 0 0 0 0 1 0 0 1 1 0 1 B_2 B_1 B_0
	CY,PSWL.bit	CY ← CY $\veebar$ PSWL.bit	6	6	2						X	0 0 0 0 0 0 1 0 0 1 1 0 0 B_2 B_1 B_0

Instructions (cont)

Bit Manipulation (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
SET1	saddr.bit	(saddr.bit) ← 1	5/7	1	2							1 0 1 1 0 B_2 B_1 B_0
												Saddr-offset
	sfr.bit	sfr.bit ← 1	8	2	3							0 0 0 0 1 0 0 0
												1 0 0 0 1 B_2 B_1 B_0
												Sfr-offset
	A.bit	A.bit ← 1	7	7	2							0 0 0 0 0 0 1 1
												1 0 0 0 1 B_2 B_1 B_0
	X.bit	X.bit ← 1	7	7	2							0 0 0 0 0 0 1 1
												1 0 0 0 0 B_2 B_1 B_0
	PSWH.bit	PSWH.bit ← 1	8	8	2							0 0 0 0 0 0 1 0
												1 0 0 0 1 B_2 B_1 B_0
	PSWL.bit	PSWL.bit ← 1	8	8	2	X	X	X	X	X	X	0 0 0 0 0 0 1 0
												1 0 0 0 0 B_2 B_1 B_0
CLR1	saddr.bit	(saddr.bit) ← 0	5/7	1	2							1 0 1 0 0 B_2 B_1 B_0
												Saddr-offset
	sfr.bit	sfr.bit ← 0	8	2	3							0 0 0 0 1 0 0 0
												1 0 0 1 1 B_2 B_1 B_0
												Sfr-offset
	A.bit	A.bit ← 0	7	7	2							0 0 0 0 0 0 1 1
												1 0 0 1 1 B_2 B_1 B_0
	X.bit	X.bit ← 0	7	7	2							0 0 0 0 0 0 1 1
												1 0 0 1 0 B_2 B_1 B_0
	PSWH.bit	PSWH.bit ← 0	8	8	2							0 0 0 0 0 0 1 0
												1 0 0 1 1 B_2 B_1 B_0
	PSWL.bit	PSWL.bit ← 0	8	8	2	X	X	X	X	X	X	0 0 0 0 0 0 1 0
												1 0 0 1 0 B_2 B_1 B_0
NOT1	saddr.bit	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$	6/8	2	3							0 0 0 0 1 0 0 0
												0 1 1 1 0 B_2 B_1 B_0
												Saddr-offset
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	8	2	3							0 0 0 0 1 0 0 0
												0 1 1 1 1 B_2 B_1 B_0
												Sfr-offset
	A.bit	A.bit ← $\overline{\text{A.bit}}$	7	7	2							0 0 0 0 0 0 1 1
												0 1 1 1 1 B_2 B_1 B_0
	X.bit	X.bit ← $\overline{\text{X.bit}}$	7	7	2							0 0 0 0 0 0 1 1
												0 1 1 1 0 B_2 B_1 B_0
	PSWH.bit	PSWH.bit ← $\overline{\text{PSWH.bit}}$	8	8	2							0 0 0 0 0 0 1 0
												0 1 1 1 1 B_2 B_1 B_0
	PSWL.bit	PSWL.bit ← $\overline{\text{PSWL.bit}}$	8	8	2	X	X	X	X	X	X	0 0 0 0 0 0 1 0
												0 1 1 1 0 B_2 B_1 B_0

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Bit Manipulation (cont)												
SET1	CY	CY ← 1	3	3	1						1	0 1 0 0 0 0 0 1
CLR1	CY	CY ← 0	3	3	1						0	0 1 0 0 0 0 0 0
NOT1	CY	CY ← $\overline{CY}$	3	3	1						X	0 1 0 0 0 0 1 0
Call/Return												
CALL	!addr16	$(SP-1) \leftarrow (PC+3)_H$, $(SP-2) \leftarrow (PC+3)_L$, PC ← addr16, SP ← SP − 2	8	0	3							0 0 1 0 1 0 0 0 Low addr High addr
CALLF	!addr11	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, PC ← addr11, SP ← SP − 2	8	0	2							1 0 0 1 0 f_{10} f_9 f_8 f_7 f_6 f_5 f_4 f_3 f_2 f_1 f_0
CALLT	[addr5]	$(SP-1) \leftarrow (PC+1)_H$, $(SP-2) \leftarrow (PC+1)_L$, $PC_H \leftarrow$ (TPF x 8000H + addr5 + 1), $PC_L \leftarrow$ (TPF x 8000H + addr5), SP ← SP − 2	13	0	1							1 1 1 t_4 t_3 t_2 t_1 t_0
CALL	rp1	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_H \leftarrow rp1_H$, $PC_L \leftarrow rp1_L$, SP ← SP − 2	9	0	2							0 0 0 0 0 1 0 1 0 1 0 1 1 Q_2 Q_1 Q_0
	[rp1]	$(SP-1) \leftarrow (PC+2)_H$, $(SP-2) \leftarrow (PC+2)_L$, $PC_H \leftarrow (rp1)_H$, $PC_L \leftarrow (rp1)_L$, SP ← SP − 2	11	0	2							0 0 0 0 0 1 0 1 0 1 1 1 1 Q_2 Q_1 Q_0
BRK		$(SP-1) \leftarrow PSW_H$, $(SP-2) \leftarrow PSW_L$, $(SP-3) \leftarrow (PC+1)_H$, $(SP-4) \leftarrow (PC+1)_L$, $PC_L \leftarrow$ (003EH), $PC_H \leftarrow$ (003FH), SP ← SP − 4 IE ← 0	20	0	1							0 1 0 1 1 1 1 0
RET		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), SP ← SP + 2	8	0	1							0 1 0 1 0 1 1 0
RETI		$PC_L \leftarrow$ (SP), $PC_H \leftarrow$ (SP + 1), $PSW_L \leftarrow$ (SP + 2), $PSW_H \leftarrow$ (SP + 3), SP ← SP + 4, EOS ← 0	14	0	1	R	R	R	R	R	R	0 1 0 1 0 1 1 1
Stack Manipulation												
PUSH	post	$((SP-1) \leftarrow rpp_H$,* $(SP-2) \leftarrow rpp_L$, SP ← SP − 2) x n	41+4n	41	2							0 0 1 1 0 1 0 1 Post byte
	PSW	$(SP-1) \leftarrow PSW_H$, $(SP-2) \leftarrow PSW_L$, SP ← SP − 2	5	1	1							0 1 0 0 1 0 0 1

*rpp refers to register pairs specified in post byte. n is the number of register pairs specified in post byte.

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Stack Manipulation (cont)												
PUSHU	post	((UP – 1) ← rpp_H,* (UP – 2) ← rpp_L, UP ← UP – 2) x n	42+4n	42	2							0 0 1 1 0 1 1 1 Post byte
POP	post	(rpp_L ← (SP),* rpp_H ← (SP + 1), SP ← SP + 2) x n	41+5n	41+n	2							0 0 1 1 0 1 0 0 Post byte
	PSW	PSW_L ← (SP), PSW_H ← (SP + 1), SP ← SP + 2	6	2	1	R	R	R	R	R	R	0 1 0 0 1 0 0 0
POPU	post	(rpp_L ← (UP),* rpp_H ← (UP + 1), UP ← UP + 2) x n	42+5n	42+n	2							0 0 1 1 0 1 1 0 Post byte
MOVW	SP,#word	SP ← word	4	0	4							0 0 0 0 1 0 1 1 1 1 1 1 1 1 0 0 Low byte High byte
	SP,AX	SP ← AX	4	0	2							0 0 0 1 0 0 1 1 1 1 1 1 1 1 0 0
	AX,SP	AX ← SP	4	1	2							0 0 0 1 0 0 0 1 1 1 1 1 1 1 0 0
INCW	SP	SP ← SP + 1	5	5	2							0 0 0 0 0 1 0 1 1 1 0 0 1 0 0 0
DECW	SP	SP ← SP – 1	5	5	2							0 0 0 0 0 1 0 1 1 1 0 0 1 0 0 1
Unconditional Branch												
BR	!addr16	PC ← addr16	4	0	3							0 0 1 0 1 1 0 0 Low addr High addr
	rp1	PC_H ← $rp1_H$, PC_L ← $rp1_L$	5	0	2							0 0 0 0 0 1 0 1 0 1 0 0 1 Q_2 Q_1 Q_0
	[rp1]	PC_H ← $(rp1)_H$, PC_L ← $(rp1)_L$	8	0	2							0 0 0 0 0 1 0 1 0 1 1 0 1 Q_2 Q_1 Q_0
	$addr16	PC ← addr16	7	0	2							0 0 0 1 0 1 0 0 jdisp
Conditional Branch												
BC or BL**	$addr16	PC ← addr16 if CY = 1	7(3)	0(3)	2							1 0 0 0 0 0 1 1 jdisp
BNC or BNL**	$addr16	PC ← addr16 if CY = 0	7(3)	0(3)	2							1 0 0 0 0 0 1 0 jdisp
BZ or BE**	$addr16	PC ← addr16 if Z = 1	7(3)	0(3)	2							1 0 0 0 0 0 0 1 jdisp

*rpp refers to register pairs specified in post byte. n is the number of register pairs specified in post byte.

**Either of the two mnemonics may be used.

Instructions (cont)

				Idle		Flags						Operation Code (Bits 7-0)
Mnemonic	Operand	Operation	States	States	Bytes	S	Z	AC	P/V	SUB	CY	Bytes B1 thru B5
Conditional Branch (cont)												
BNZ or BNE**	$addr16	PC ← addr16 if Z = 0	7(3)	0(3)	2							1 0 0 0 0 0 0 0 jdisp
BV or BPE**	$addr16	PC ← addr16 if P/V = 1	7(3)	0(3)	2							1 0 0 0 0 1 0 1 jdisp
BNV or BPO**	$addr16	PC ← addr16 if P/V = 0	7(3)	0(3)	2							1 0 0 0 0 1 0 0 jdisp
BN	$addr16	PC ← addr16 if S = 1	7(3)	0(3)	2							1 0 0 0 0 1 1 1 jdisp
BP	$addr16	PC ← addr16 if S = 0	7(3)	0(3)	2							1 0 0 0 0 1 1 0 jdisp
BGT	$addr16	PC ← addr16 if (P/V⊻S) V Z = 0	9(5)	0(5)	3							0 0 0 0 0 1 1 1 1 1 1 1 1 0 1 1 jdisp
BGE	$addr16	PC ← addr16 if P/V⊻ S = 0	9(5)	0(5)	3							0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 1 jdisp
BLT	$addr16	PC ← addr16 if P/V⊻ S = 1	9(5)	0(5)	3							0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 jdisp
BLE	$addr16	PC ← addr16 if (P/V⊻ S) V Z = 1	9(5)	0(5)	3							0 0 0 0 0 1 1 1 1 1 1 1 1 0 1 0 jdisp
BH	$addr16	PC ← addr16 if Z V CY = 0	9(5)	0(5)	3							0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 1 jdisp
BNH	$addr16	PC ← addr16 if Z V CY = 1	9(5)	0(5)	3							0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 jdisp
BT	saddr.bit,$addr16	PC ← addr16 if (saddr.bit) = 1	9(6)/ 10(7)	0(4)	3							0 1 1 1 0 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← addr16 if sfr.bit = 1	11(8)	0(5)	4							0 0 0 0 1 0 0 0 1 0 1 1 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← addr16 if A.bit = 1	10(7)	0(7)	3							0 0 0 0 0 0 1 1 1 0 1 1 1 B_2 B_1 B_0 jdisp

**Either of the two mnemonics may be used.

Instructions (cont)

Mnemonic	Operand	Operation	States	Idle States	Bytes	Flags						Operation Code (Bits 7-0) Bytes B1 thru B5
						S	Z	AC	P/V	SUB	CY	
Conditional Branch (cont)												
BT (cont)	X.bit,$addr16	PC ← addr16 if X.bit = 1	10(7)	0(7)	3							0 0 0 0 0 0 1 1 1 0 1 1 0 B_2 B_1 B_0 jdisp
	PSWH.bit,$addr16	PC ← addr16 if PSWH.bit = 1	10(7)	0(7)	3							0 0 0 0 0 0 1 0 1 0 1 1 1 B_2 B_1 B_0 jdisp
	PSWL.bit,$addr16	PC ← addr16 if PSWL.bit = 1	10(7)	0(7)	3							0 0 0 0 0 0 1 0 1 0 1 1 0 B_2 B_1 B_0 jdisp
BF	saddr.bit,$addr16	PC ← addr16 if (saddr.bit) = 0	10(7)/ 11(8)	0(5)	4							0 0 0 0 1 0 0 0 1 0 1 0 0 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← addr16 if sfr.bit = 0	11(8)	0(5)	4							0 0 0 0 1 0 0 0 1 0 1 0 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← addr16 if A.bit = 0	10(7)	0(7)	3							0 0 0 0 0 0 1 1 1 0 1 0 1 B_2 B_1 B_0 jdisp
	X.bit,$addr16	PC ← addr16 if X.bit = 0	10(7)	0(7)	3							0 0 0 0 0 0 1 1 1 0 1 0 0 B_2 B_1 B_0 jdisp
	PSWH.bit,$addr16	PC ← addr16 if PSWH.bit = 0	10(7)	0(7)	3							0 0 0 0 0 0 1 0 1 0 1 0 1 B_2 B_1 B_0 jdisp
	PSWL.bit,$addr16	PC ← addr16 if PSWL.bit = 0	10(7)	0(7)	3							0 0 0 0 0 0 1 0 1 0 1 0 0 B_2 B_1 B_0 jdisp
BTCLR	saddr.bit,$addr16	PC ← addr16 if (saddr.bit) = 1; then reset (saddr.bit)	12(7)/ 14(8)	0(5)	4							0 0 0 0 1 0 0 0 1 1 0 1 0 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← addr16 if sfr.bit = 1; then reset sfr.bit	14(8)	0(5)	4							0 0 0 0 1 0 0 0 1 1 0 1 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← addr16 if A.bit = 1; then reset A.bit	11(7)	0(7)	3							0 0 0 0 0 0 1 1 1 1 0 1 1 B_2 B_1 B_0 jdisp

Instructions (cont)

Mnemonic	Operand	Operation	States	Idle States	Bytes	Flags S	Z	AC	P/V	SUB	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
Conditional Branch (cont)												
BTCLR (cont)	X.bit,$addr16	PC ← addr16 if X.bit = 1; then reset X.bit	11(7)	0(7)	3							0 0 0 0 0 0 1 1 1 1 0 1 0 B_2 B_1 B_0 jdisp
	PSWH.bit,$addr16	PC ← addr16 if PSWH.bit = 1; then reset PSWH.bit	12(7)	0(7)	3							0 0 0 0 0 0 1 0 1 1 0 1 1 B_2 B_1 B_0 jdisp
	PSWL.bit,$addr16	PC ← addr16 if PSWL.bit = 1; then reset PSWL.bit	12(7)	0(7)	3	X	X	X	X	X	X	0 0 0 0 0 0 1 0 1 1 0 1 0 B_2 B_1 B_0 jdisp
BFSET	saddr.bit,$addr16	PC ← addr16 if (saddr.bit) = 0; then set (saddr.bit)	12(7)/ 14(8)	0(5)	4							0 0 0 0 1 0 0 0 1 1 0 0 0 B_2 B_1 B_0 Saddr-offset jdisp
	sfr.bit,$addr16	PC ← addr16 if sfr.bit = 0; then set sfr.bit	14(8)	0(5)	4							0 0 0 0 1 0 0 0 1 1 0 0 1 B_2 B_1 B_0 Sfr-offset jdisp
	A.bit,$addr16	PC ← addr16 if A.bit = 0; then set A.bit	11(7)	0(7)	3							0 0 0 0 0 0 1 1 1 1 0 0 1 B_2 B_1 B_0 jdisp
	X.bit,$addr16	PC ← addr16 if X.bit = 0; then set X.bit	11(7)	0(7)	3							0 0 0 0 0 0 1 1 1 1 0 0 0 B_2 B_1 B_0 jdisp
	PSWH.bit,$addr16	PC ← addr16 if PSWH.bit = 0; then set PSWH.bit	12(7)	0(7)	3							0 0 0 0 0 0 1 0 1 1 0 0 1 B_2 B_1 B_0 jdisp
	PSWL.bit,$addr16	PC ← addr16 if PSWL.bit = 0; then set PSWL.bit	12(7)	0(7)	3	X	X	X	X	X	X	0 0 0 0 0 0 1 0 1 1 0 0 0 B_2 B_1 B_0 jdisp
DBNZ	r2,$addr16	r2 ← r2 − 1; then PC ← addr16 if r2 ≠ 0	8(5)	0(5)	2							0 0 1 1 0 0 1 C_0 jdisp
	saddr,$addr16	(saddr) ← (saddr) − 1; then PC ← addr16 if saddr ≠ 0	9(6)/ 11(8)	0(2)	3							0 0 1 1 1 0 1 1 Saddr-offset jdisp

Instructions (cont)

Mnemonic	Operand	Operation	States	Idle States	Bytes	Flags S	Z	AC	P/V	SUB	CY	Operation Code (Bits 7-0) Bytes B1 thru B5
Context Switch												
BRKCS	RBn	PC$_H$ ⟷ R5, PC$_L$ ⟷ R4, R7 ← PSW$_H$, R6 ← PSW$_L$, RBS2-RBS0 ← n, RSS ← 0, IE ← 0	12	0	2							0 0 0 0 0 1 0 1 1 1 0 1 1 N_2 N_1 N_0
RETCS	!addr16	PC$_H$ ← R5, PC$_L$ ← R4, R5, R4 ← !addr16, PSW$_H$ ← R7, PSW$_L$ ← R6, EOS ← 0	6	0	3	R	R	R	R	R	R	0 0 1 0 1 0 0 1 Low addr High addr
String												
MOVM	[DE+],A	(DE+) ← A, C ← C − 1 End if C = 0	2+7n (4+7n)	2+5n (3+5n)	2							0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0
	[DE−],A	(DE−) ← A, C ← C − 1 End if C = 0	2+7n (4+7n)	2+5n (3+5n)	2							0 0 0 1 0 1 0 1 0 0 0 1 0 0 0 0
MOVBK	[DE+],[HL+]	(DE+) ← (HL+), C ← C − 1 End if C = 0	2+10n (4+10n)	2+6n (3+6n)	2							0 0 0 1 0 1 0 1 0 0 1 0 0 0 0 0
	[DE−],[HL−]	(DE−) ← (HL−), C ← C − 1 End if C = 0	2+10n (4+10n)	2+6n (3+6n)	2							0 0 0 1 0 1 0 1 0 0 1 1 0 0 0 0
XCHM	[DE+],A	(DE+) ⟷ A, C ← C − 1 End if C = 0	2+12n (4+12n)	2+6n (3+6n)	2							0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 1
	[DE−],A	(DE−) ⟷ A, C ← C − 1 End if C = 0	2+12n (4+12n)	2+6n (3+6n)	2							0 0 0 1 0 1 0 1 0 0 0 1 0 0 0 1
XCHBK	[DE+],[HL+]	(DE+) ⟷ (HL+), C ← C − 1 End if C = 0	2+15n (4+15n)	2+7n (3+7n)	2							0 0 0 1 0 1 0 1 0 0 1 0 0 0 0 1
	[DE−],[HL−]	(DE−) ⟷ (HL−), C ← C − 1 End if C = 0	2+15n (4+15n)	2+7n (3+7n)	2							0 0 0 1 0 1 0 1 0 0 1 1 0 0 0 1
CMPME	[DE+],A	(DE+) − A, C ← C − 1 End if C = 0 or Z = 0	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 0 0 1 0 0
	[DE−],A	(DE−) − A, C ← C − 1 End if C = 0 or Z = 0	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0
CMPBKE	[DE+],[HL+]	(DE+) − (HL+), C ← C − 1 End if C = 0 or Z = 0	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 0 0 1 0 0
	[DE−],[HL−]	(DE−) − (HL−), C ← C − 1 End if C = 0 or Z = 0	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 1 0 1 0 0
CMPMNE	[DE+],A	(DE+) − A, C ← C − 1 End if C = 0 or Z = 1	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 0 0 1 0 1
	[DE−],A	(DE−) − A, C ← C − 1 End if C = 0 or Z = 1	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 1 0 1 0 1
CMPBKNE	[DE+],[HL+]	(DE+) − (HL+), C ← C − 1 End if C = 0 or Z = 1	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 0 0 1 0 1
	[DE−],[HL−]	(DE−) − (HL−), C ← C − 1 End if C = 0 or Z = 1	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 1 0 1 0 1

Instructions

Mnemonic	Operand	Operation	States	Idle States	Bytes	Flags						Operation Code (Bits 7-0) Bytes B1 thru B5
						S	Z	AC	P/V	SUB	CY	
String (cont)												
CMPMC	[DE+],A	(DE+) – A, C ← C – 1 End if C = 0 or CY = 0	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 0 0 1 1 1
	[DE–],A	(DE–) – A, C ← C – 1 End if C = 0 or CY = 0	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 1 0 1 1 1
CMPBKC	[DE+],[HL+]	(DE+) – (HL+), C ← C – 1 End if C = 0 or CY = 0	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 0 0 1 1 1
	[DE–],[HL–]	(DE–) – (HL–), C ← C – 1 End if C = 0 or CY = 0	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 1 0 1 1 1
CMPMNC	[DE+],A	(DE+) – A, C ← C – 1 End if C = 0 or CY = 1	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 0 0 1 1 0
	[DE–],A	(DE–) – A, C ← C – 1 End if C = 0 or CY = 1	2+7n (4+7n)	2+5n (3+5n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 0 1 0 1 1 0
CMPBKNC	[DE+],[HL+]	(DE+) – (HL+), C ← C – 1 End if C = 0 or CY = 1	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 0 0 1 1 0
	[DE–],[HL–]	(DE–) – (HL–), C ← C – 1 End if C = 0 or CY = 1	2+10n (4+10n)	2+6n (3+6n)	2	X	X	X	V	1	X	0 0 0 1 0 1 0 1 0 0 1 1 0 1 1 0
CPU Control												
MOV	STBC,#byte	STBC ← byte	6	1	4							0 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 $\overline{\text{Data}}$ Data
	WDM,#byte	WDM ← byte	6	1	4							0 0 0 0 1 0 0 1 0 1 0 0 0 0 1 0 $\overline{\text{Data}}$ Data
SWRS		RSS ← $\overline{\text{RSS}}$	3	3	1							0 1 0 0 0 0 1 1
SEL	RBn	RSS ← 0, RBS2-RBS0 ← n	4	4	2							0 0 0 0 0 1 0 1 1 0 1 0 1 N_2 N_1 N_0
	RBn,ALT	RSS ← 1, RBS2-RBS0 ← n	4	4	2							0 0 0 0 0 1 0 1 1 0 1 1 1 N_2 N_1 N_0
NOP		No operation	3	3	1							0 0 0 0 0 0 0 0
EI		IE ← 1 (Enable interrupt)	3	3	1							0 1 0 0 1 0 1 1
DI		IE ← 0 (Disable interrupt)	3	3	1							0 1 0 0 1 0 1 0

NEC
NEC Electronics Inc.

μPD7832x
Advanced 8/16-Bit, Real Time Control Microcomputer With A/D Converter

Description

The μPD7832x (78320, 78322) is a single-chip microcomputer designed for process control. It features a 16-bit CPU, an 8-bit external data bus, and a powerful set of on-chip peripherals including counters and timers, an A/D converter, two serial ports, and a maximum of 55 input/output lines.

An advanced interrupt handling facility includes a three-level program-controlled hardware priority interrupt controller and three separate methods of handling interrupt requests. It is manufactured of 1.2μ CMOS process, operates from a single 5 V power supply, and has a maximum oscillator frequency of 16 MHz.

The μPD7832x has 16K bytes of on-chip mask-programmed ROM, and the μPD78320 is a ROM-less version. Both chips have 640 bytes of on-chip RAM and are supplied in a 68-pin PLCC or 74-pin plastic QFP package.

The μPD7832x has an interface for a special dedicated memory chip, the μPD71P301. The μPD71P301 includes memory, interface circuitry, and an instruction prefetch pointer. This makes it possible to fetch instructions from external memory at the same high speed at which they can be fetched from on-chip ROM.

The primary applications of the μPD7832x include automotive engine control, antilock braking control, and control of computer disks and tapes. Its speed and powerful on-chip peripherals, however, make it suitable for all of the more demanding types of process control.

Features

□ Complete single-chip microcomputer
— 16-bit ALU
— 16K bytes of ROM (μPD78322 only)
— 640 bytes RAM

□ Powerful instruction set
— 16-bit multiply and divide
— 1-bit and 8-bit logic instructions
— String instructions

□ Minimum instruction time
— 250 ns @ 16-MHz input

□ 3-byte instruction prefetch queue

□ Memory expansion
— 8085 bus compatible
— 64K-byte address space
— High-speed fetch from external memory

□ Large I/O capacity
— Up to 55 I/O port lines

□ Special interface for turbo access manager (TAM) μPD71P301

□ Memory-mapped on-chip peripherals (special function registers)

□ Multipurpose pulse input/output unit
— 16-/18-bit free-running timer
— 16-bit timer/event counter
— Six 16-bit compare registers
— Four 18-bit capture registers
— Two 18-bit capture/compare registers
— Six external interrupt/capture lines
— One external event counter/interrupt line
— Six timer-controlled output lines

□ 10-bit, 8-channel analog to digital converter
— On-chip sample and hold amplifier

□ Two-channel serial communication interface
— Asynchronous serial interface (UART)
— Serial bus interface
— Dedicated baud rate generator

□ Programmable priority interrupt controller (3 levels)

□ Three methods of interrupt service
— Vectored interrupts
— Context switching with hardware save of all general registers
— Nine macroservice functions

□ Watchdog timer with dedicated output

□ STOP and HALT standby functions

□ Single 5-volt power supply

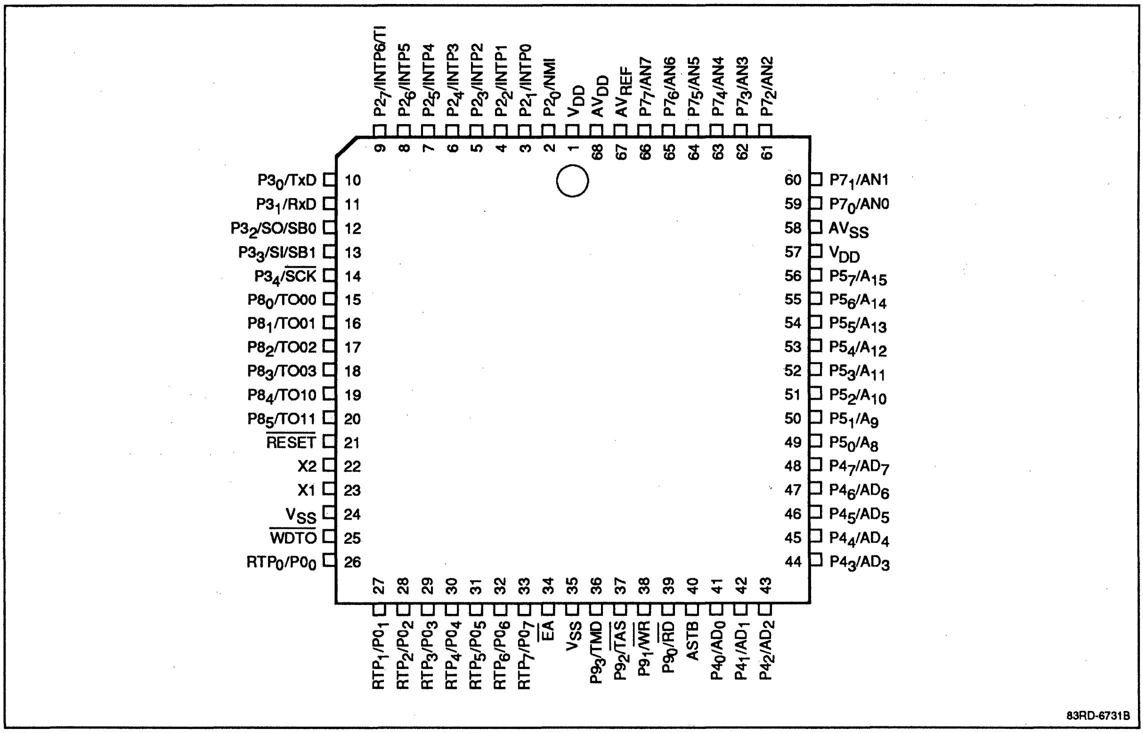
Ordering Information

Part Number	On-Chip ROM	Package Type
μPD78320L	No	68-pin PLCC
μPD78320GJ-5BJ	No	74-pin plastic QFP
μPD78322L-xxx	Yes	68-pin PLCC
μPD78322GJ-xxx-5BJ	Yes	74-pin plastic QFP

xxx is the mask code number

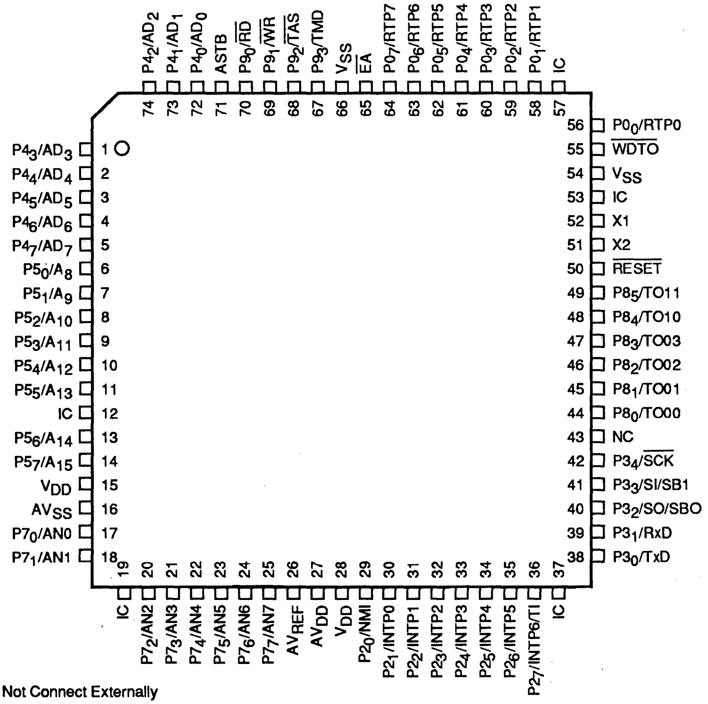
Pin Configurations

68-Pin PLCC



Pin Configuration (cont)

74-Pin Plastic QFP



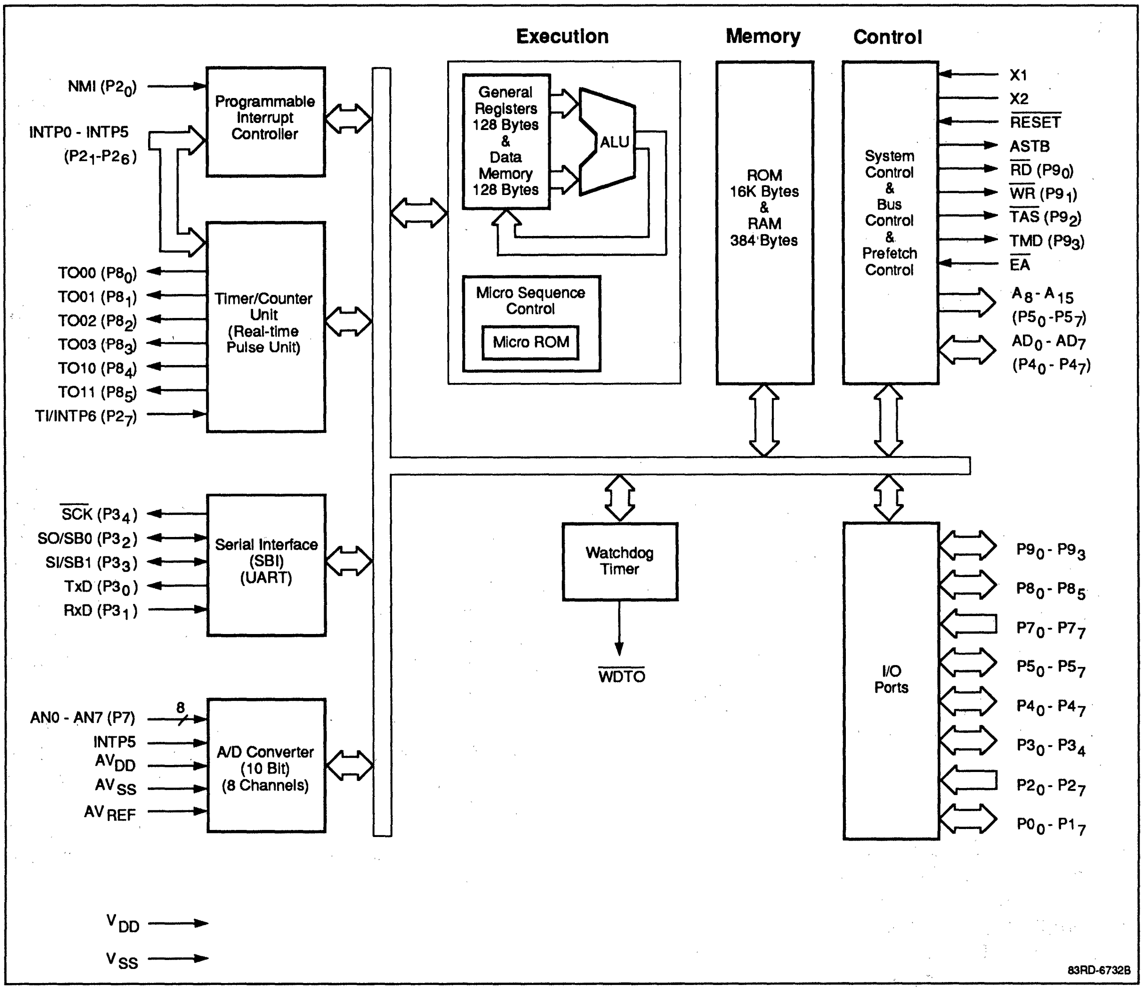
IC • Internally Connected • Do Not Connect Externally

83RD-6730B

Pin Function

Symbol	First Function	Symbol	Second Function
P0$_0$-P0$_7$	Port 0; 8-bit, bit selectable I/O port	RTP$_0$-RTP$_7$	Bit selectable, timer-controlled, real-time output port
P2$_0$	Port 2; 8-bit input port	NMI	External nonmaskable interrupt
P2$_1$		INTP0	Maskable external interrupts; edge-selectable
P2$_2$		INTP1	
P2$_3$		INTP2	
P2$_4$		INTP3	
P2$_5$		INTP4	
P2$_6$		INTP5	
P2$_7$		INTP6/T1	External interrupt or timer input
P3$_0$	Port 3; 5-bit, bit selectable I/O port	TxD	Asynchronous serial transmit
P3$_1$		RxD	Asysnchronous serial receive
P3$_2$		SO/SB0	Synchronous serial line
P3$_3$		SI/SB1	Synchronous serial line
P3$_4$		$\overline{\text{SCK}}$	Serial clock input or output
P4$_0$-P4$_7$	Port 4; 8-bit, byte selectable I/O port	AD$_0$-AD$_7$	Low-order byte of external address/data bus
P5$_0$-P5$_7$	Port 5; 8-bit, bit selectable I/O port	A$_8$-A$_{15}$	High-order byte of external address bus
P$_7$-P7$_7$	Port 7; 8-bit input port	AN0-AN7	Inputs for A/D converter
P8$_0$	Port 8; 6-bit, bit selectable I/O port	TO00	Timer (RPU) output lines
P8$_1$		TO01	
P8$_2$		TO02	
P8$_3$		TO03	
P8$_4$		TO10	
P8$_5$		TO11	
P9$_0$	Port 9; 4-bit, bit-selectable I/O port	$\overline{\text{RD}}$	External read strobe
P9$_1$		$\overline{\text{WR}}$	External write strobe
P9$_2$		$\overline{\text{TAS}}$	TAM strobe
P9$_3$		TMD	TAM control
ASTB	External address latch strobe		
$\overline{\text{EA}}$	External access control; a high level enables access to on-chip ROM; a low level is applied if all program memory is external. Must be tied low for the μPD78320.		
$\overline{\text{RESET}}$	External system reset input		
$\overline{\text{WDTO}}$	Watchdog timer output		
X1, X2	For frequency control of the internal clock oscillator, a crystal is connected to X1 and X2. If the clock is supplied by an external source, the clock signal is connected to X1 and the inverted clock signal is connected to X2.		
AV$_{REF}$	A/D converter reference voltage input		
AV$_{DD}$	A/D converter +5-volt power input		
AV$_{SS}$	A/D converter ground		
V$_{DD}$	+5-volt power input		
V$_{SS}$	Ground		

µPD7832x Block Diagram



FUNCTIONAL DESCRIPTION

Central Processing Unit

The Central Processing Unit (CPU) of the μPD7832x features 16-bit arithmetic including 16-by-16 bit multiply, both signed and unsigned, and 32-by-16 bit divide (producing a 32-bit quotient and 16-bit remainder). String instructions and both 8-bit and 1-bit logic instructions are included.

Instructions range in length from one to five bytes, depending on the instruction and addressing mode. A 1-byte call instruction can access up to 32 addresses specified in the CALLT vector table in lower memory. A 2-byte call instruction can access any routine beginning in a specific CALLF area. A single instruction can test individual bits both in a portion of on-chip RAM and in the special function registers.

A 3-byte instruction prefetch queue makes it possible to fetch instruction bytes on a separate bus during execution cycles. Instructions are fetched from on-chip ROM at a rate of one byte per cycle. An interface is provided for the μPD71P301 memory chip, called the Turbo Access Manager (TAM). TAM makes possible similar fetch rates from external memory.

The CPU clock is generated by dividing the oscillator frequency by two. Therefore, when the oscillator frequency is 16 MHz, the clock is 8 MHz. Some instructions execute in two cycles, and the minimum instruction time is 250 ns.

Addressing

The μPD7832x features 1-byte addressing of both the special function registers and a portion of the on-chip RAM. The 1-byte sfr addressing accesses the entire SFR area, while the 1-byte saddr addressing accesses 32 bytes of the SFR area and 224 bytes of the on-chip RAM. Nine modes for addressing main memory include indexing, double indexing, autoincrement, and autodecrement. Main memory addressing can be used to access the entire 64K address space including the SFR area and RAM. There are also both 8-bit and 16-bit immediate operands.

External Memory

The external memory bus is 8 bits wide, and external memory can be used to fill up the 64K-bit address space. Either ROM or RAM (or both) can be used as required. The low order 8 bits of the address/data bus are multiplexed, and are supplied by I/O port 4. High-order address bits are taken from port 5 as required. Address latch, read, and write strobes are provided. Two special control lines provide access to the TAM. The memory mode register controls the size of the external memory and the number of additional wait states. The high-order address uses 0, 4, 6, or 8 bits from port 5, depending on the amount of external memory required. Any remaining port 5 bits can be used for I/O. Figure 1 shows the memory map of the μPD7832x.

General Registers

Sixteen 8-bit general registers can be used in pairs to function as 16-bit registers. A complete set of 16 registers is mapped into each of eight program selectable register banks stored in RAM. Three bits in the PSW (figure 2) specify which of the register banks is active at any time. Registers have both functional names (A, AX, C, DE, etc.) and absolute names (R1, RP0, R2, RP6, etc.). Each instruction determines whether a register is referred to by functional or absolute name and whether it is 8 or 16 bits.

Two possible relationships may exist between the absolute and functional names of the first four register pairs. The RSS bit in the PSW determines which of these is active at any time. The effect is that the accumulator and counter registers can be saved, and a new set can be specified by toggling the RSS bit. Figure 3 illustrates the general register configuration.

Figure 1. µPD7832x Memory Map

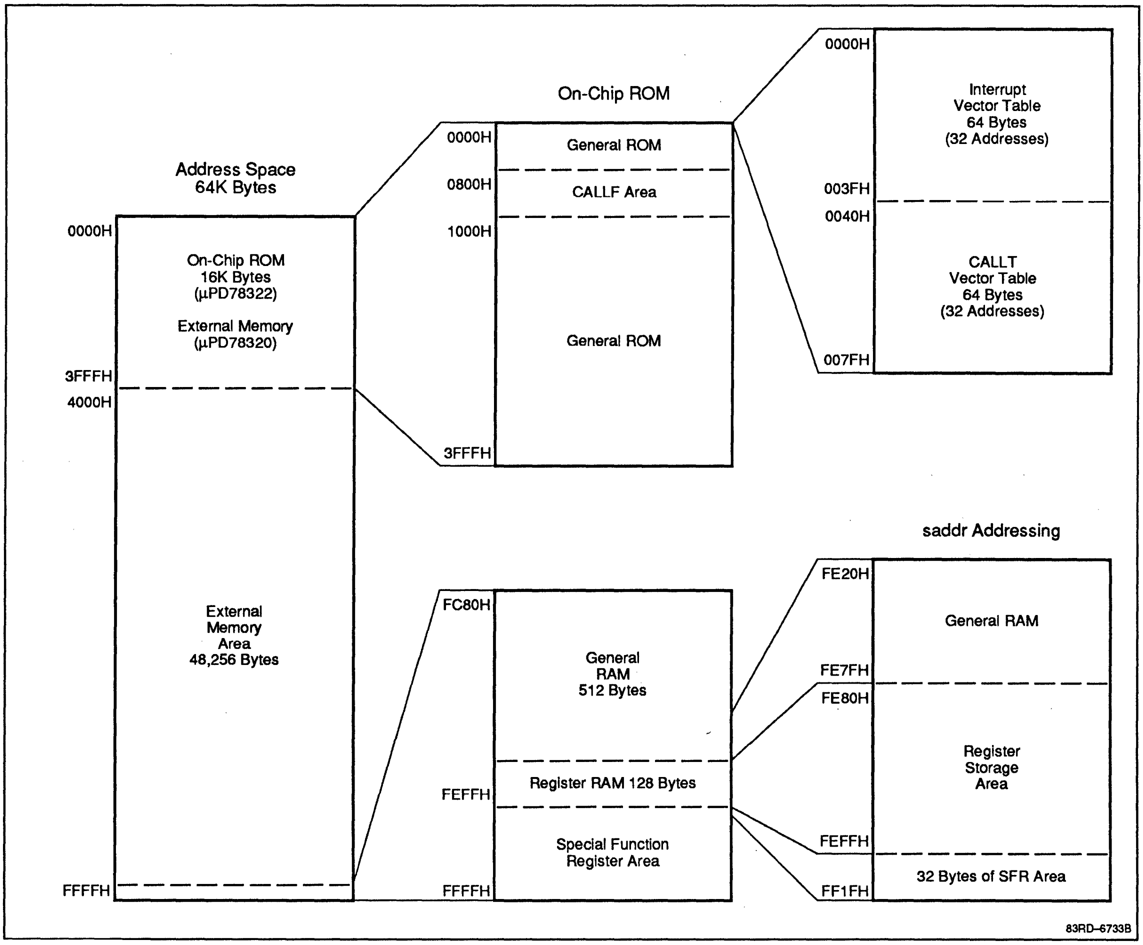


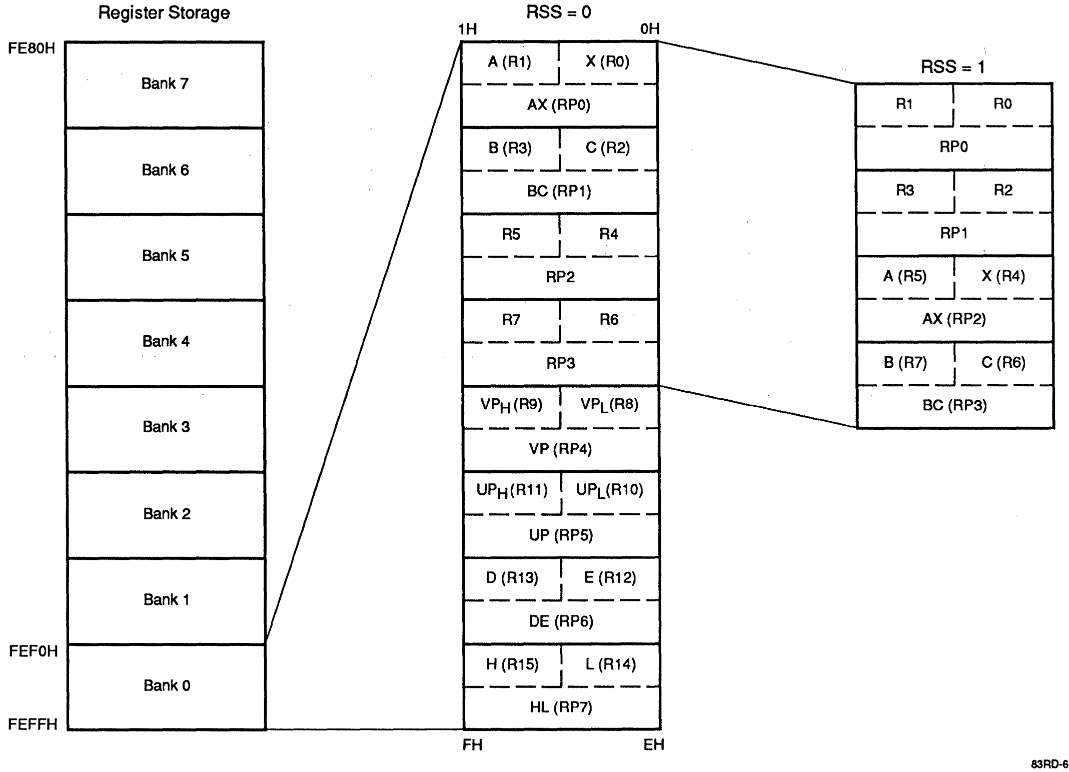
Figure 2. Program Status Word

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UF	RBS2	RBS1	RBS0	0	0	0	0	S	Z	RSS	AC	IE	P/V	LT	CY

UF	User flag
RBS2 - RBS0	Active register bank selector
S	Sign flag
Z	Zero flag
RSS	Register set selection flag
AC	Auxilliary carry flag
IE	Interrupt Enable flag
P/V	Parity or Overflow flag
LT	In-service priority level transition flag
CY	Carry bit

83RD-6735B

Figure 3. Register Configuration and Storage



83RD-6734B

Input/Output

Eight I/O ports range in size from 4 to 8 bits, providing a total of 55 I/O lines. All I/O lines have alternate control functions which can be specified under program control. All except ports 2, 4, and 7 can be specified for input or output on an individual bit basis. Ports 2 and 7 are input (or control input) only. Port 4 is byte selectable for input, output, or control.

Real-Time Output Port

Port 0 can function on a bit-selectable basis as a real-time output port. Real-time port bits can be directly written under program control, or they can be set or cleared under control of timing signals generated by the real-time pulse unit. This provides output timing that is independent of interrupt latency.

External Interrupts

One nonmaskable and 7 maskable external interrupts share pins with port 2. The maskable interrupts can also be used to trigger capture events in the real-time pulse unit. Any masked interrupt automatically becomes an input line. INTP6 is also used as the counter input for timer TM1 when TM1 is used as an external event counter.

Serial Ports

The μPD7832x has two serial ports. The first is a standard asynchronous serial port that shares pins with $P3_0$ (TxD) and $P3_1$ (RxD). It generates three interrupts INTST (transmit complete), INTSR (receive buffer full), and INTSER (receive error).

The second serial port can be used in one of two modes. The first mode is a 3-wire I/O interface mode with send, receive, and clock lines. Data are sent and received most significant bit first, and the clock line can be driven either internally or externally. The second mode is the 2-wire NEC serial bus interface (SBI) mode. SBI features wake-up signals and distinction between commands, addresses, and data, all decoded by hardware.

The synchronous serial port shares I/O pins with port 3 bits 2-4 and generates a single interrupt, INTCSI. A dedicated baud rate generator is included so that all of the commonly used baud rates can be generated when the oscillator frequency is correctly chosen.

Analog to Digital Converter

An 8-channel 10-bit A/D converter provides a relative accuracy of 0.2% full scale. An on-chip sample-and-hold amplifier is included, and the eight input channels share pins with port 7. The A/D converter can be operated in either the scan mode (where either channels 0-3 or 4-7 are repeatedly scanned) or the select mode (where a specific channel is selected and converted repeatedly). The conversion can be started either by software or by an external signal on INTP5.

Real-Time Pulse Unit

The real-time pulse unit (RPU, figure 4) consists of an 18-bit free-running timer, TM0, 16-bit timer/counter, TM1, six 16-bit compare registers, four 18-bit capture registers, two 18-bit registers which can be used for either capture or compare, and six timed output latches. TM0 always counts the system clock (divided by either 4 or 8) and can be reset by external RESET only. TM1 can count either the system clock (divided by either 8 or 16) or external events. TM1 can be reset by either a compare event (a match between a timer and an associated compare register) or by an external signal in INTP0.

Capture events can be triggered by external maskable interrupts INTP0-INTP5, and compare events can be used to generate interrupts, control timed output pins, or both. In addition, two of them, INTCM03 and INTCCX0, can be used to control the real-time output port. The timed output latches share pins with port 8. Four of them can be toggled or set and reset by compare events, and the remaining two can be toggled. These latches, with the macroservice facility, can be used to generate up to four pulse-width modulated outputs.

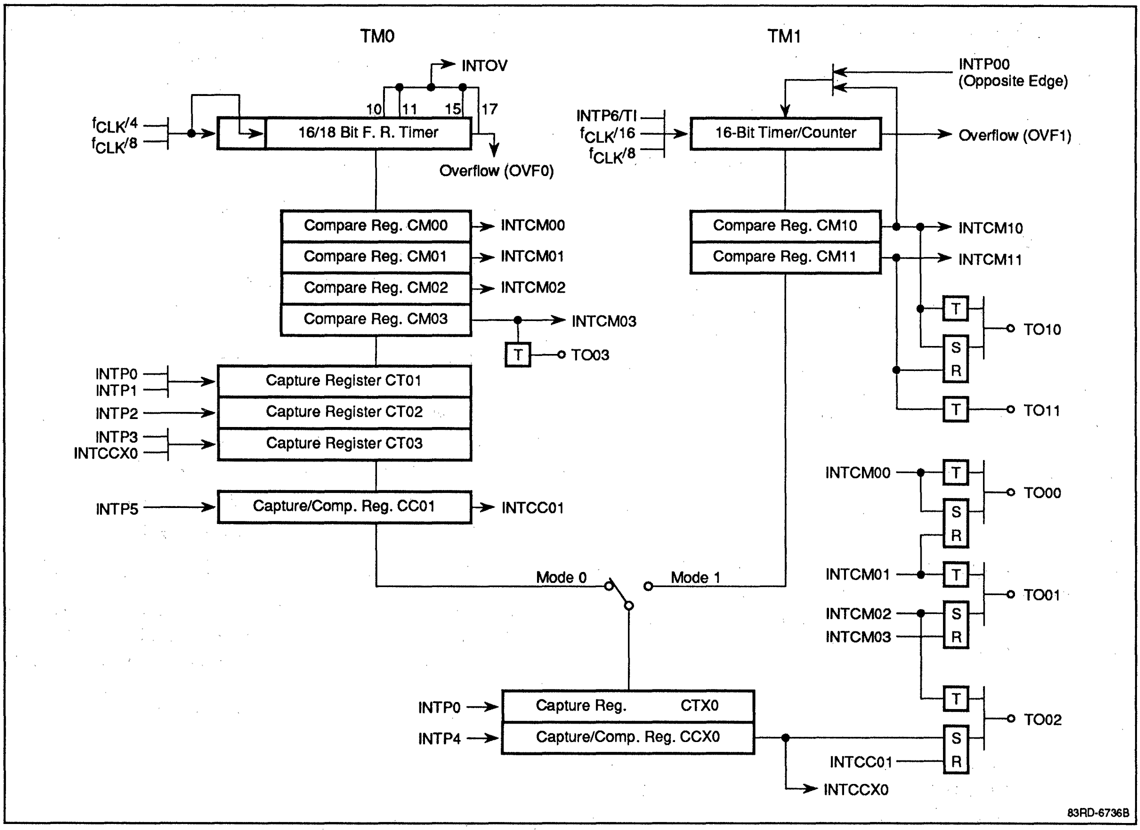
Standby Modes

HALT and STOP modes conserve power when CPU action is not required. In HALT mode, the CPU is stopped and the clock continues to run. Any unmasked interrupt can then restart the CPU. In STOP mode, the CPU and clock are both stopped. Either an external RESET pulse or an external nonmaskable interrupt is required to restart them. The standby control register (STBC) is a protected location and can be written to only by a special instruction.

Watchdog Timer

The watchdog timer protects against inadvertent program loops. A nonmaskable interrupt occurs if the timer is not reset before it overflows. Three program selectable intervals are available: 8.19, 32.7, and 131.0 msec for a system clock frequency of 8 MHz. An output line is provided, which can be connected to the RESET pin or used to control external circuitry. Once started, the timer can be stopped by external RESET only. In addition, the watchdog timer mode register, WDM, is a protected location and can be written to only by a special instruction.

Figure 4. Real-Time Pulse Unit



Interrupt Handling

The μPD7832x has three different methods of handling maskable interrupt requests, standard vectoring, context switching, and macroservice. The programmer can choose the mode that is most advantageous in any given situation. The μPD7832x has 19 maskable hardware interrupt sources: 7 external and 12 internal. In addition, there are two nonmaskable interrupts, two software interrupts, and a RESET. See table 1.

Interrupt Priority

The two nonmaskable interrupts, NMI and INTWDT, take priority over all others. Their priority relative to each other is under program control.

Three hardware controlled priority levels are available for the maskable interrupts. Any one of the three levels can be assigned by software to each of the maskable interrupt lines. Interrupt requests of a priority equal to or higher than the processor's current priority level are accepted. Requests of lower priority are pending until the processor's priority state is lowered by a return instruction from the current service routine. Interrupt requests programmed to be handled by macroservice have priority over all software interrupt service regardless of the assigned priority level. See figure 5.

Software interrupts, the BRK and BRKCS instruction, and operation code trap, are executed regardless of the processor's priority level and do not alter the priority level.

Figure 5. Interrupt Service Sequence

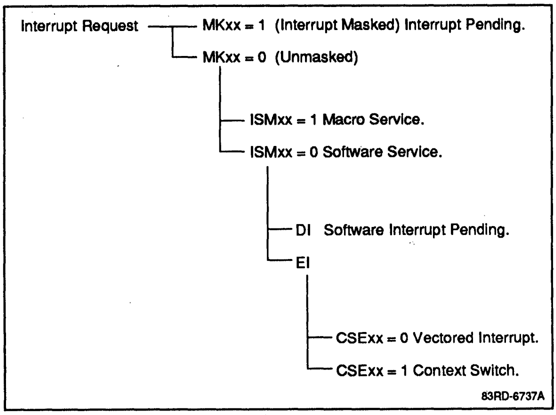

Interrupt Request
MKxx = 1 (Interrupt Masked) Interrupt Pending.
MKxx = 0 (Unmasked)
ISMxx = 1 Macro Service.
ISMxx = 0 Software Service.
DI Software Interrupt Pending.
EI
CSExx = 0 Vectored Interrupt.
CSExx = 1 Context Switch.
83RD-6737A

Table 1. Interrupt Sources

Request	Default Priority	Mnemonic	Source	Vector Address	Macroservice	Control Word*
Software	—	BRK	Break instruction	003EH	N	—
Software	—	TRAP	Opcode trap	003CH	N	—
Nonmaskable	—	NMI	External NMI	0002H	N	—
Nonmaskable	—	INTWDT	Watchdog timer	0004H	N	—
Maskable	0	INTOV	RPU	0006H	Y	FE06H
Maskable	1	INTP0	RPU/External	0008H	Y	FE08H
Maskable	2	INTP1	RPU/External	000AH	Y	FE0AH
Maskable	3	INTP2	RPU/External	000CH	Y	FE0CH
Maskable	4	INTP3	RPU/External	000EH	Y	FE0EH
Maskable	5	INTP4/INTCCX0	RPU/External	0010H	Y	FE10H
Maskable	6	INTP5/INTCC01	RPU/External	0012H	Y	FE12H
Maskable	7	INTP6	External	0014H	Y	FE14H
Maskable	8	INTCM00	RPU	0016H	Y	FE16H
Maskable	9	INTCM01	RPU	0018H	Y	FE18H
Maskable	10	INTCM02	RPU	001AH	Y	FE1AH
Maskable	11	INTCM03	RPU	001CH	Y	FE1CH
Maskable	12	INTCM10	RPU	001EH	Y	FE1EH
Maskable	13	INTCM11	RPU	0020H	Y	FE20H
Maskable	14	INTSER	UART	0022H	N	—
Maskable	15	INTSR	UART	0024H	Y	FE24H
Maskable	16	INTST	UART	0026H	Y	FE26H
Maskable	17	INTCSI	Clocked serial interface	0028H	Y	FE28H
Maskable	18	INTAD	A/D Converter	002AH	Y	FE2AH
RESET	—	RESET	External reset	0000H	N	—

*Address of macroservice control word in on-chip RAM.

Vectored Interrupt

When vectored interrupt is specified for a given interrupt request, the program status word and the program counter are saved on the stack, the processor's priority is raised to that specified for the interrupt, and the routine whose address is in the interrupt vector table is entered. At the completion of the service routine, the RETI instruction (or RETB instruction for software interrupts) reverses the process.

Context Switch

When context switching (figure 6) is specified for a given interrupt, the active register bank is changed to the register bank specified by the three low-order bits of the word in the interrupt vector table. The program counter is loaded from RP2 of the new register bank, and the program counter and program status word are saved in RP2 and RP3 of the new register bank. At the completion of the service routine, the RETCS instruction for routines entered from hardware requests, or the RETCSB instruction for routines entered from the BRKCS instruction, reverses the process. These instructions have a 16-bit immediate operand which must be set to the entry address of the service routine.

Macroservice

When macroservice is specified for a given interrupt, the macroservice hardware performs any one of nine functions during cycles "stolen" from the executing program. Control is then returned to the executing program, and the operation is therefore completely transparent. Macroservice significantly improves response time and makes it unnecessary to save any registers.

For each request on the interrupt line, one operation is performed, and a counter is decremented. When the counter reaches zero (or when some other completion condition is met), a software service routine is entered. Either vectored interrupt or context switch can be specified for entry to the completion routine, and the routine is entered according to the specified priority.

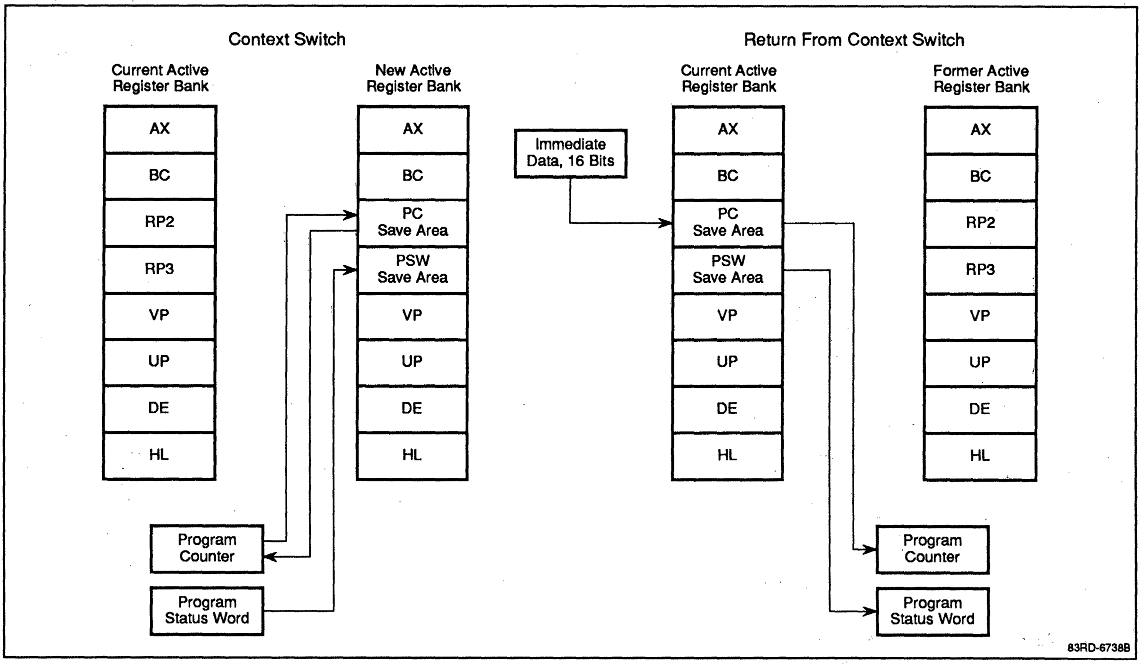
Macroservice is provided for all but one of the maskable interrupt requests, and each has a specific macroservice control word stored in on-chip RAM. The function to be performed is specified in the control word.

The nine macroservice functions are as follows:

Function	Description
EVTCNT	Event counter
DTACMP	Data compare
BITSHT	Bit shift
BITLOG	Bit logic
ADCBUF	A/D converter buffering
BLKTRS	Block transfer
DTADIF	Data difference
DTADIF-P	Data difference-pointer
DTADD	Data addition

The BLKTRS function moves either a byte or word of data in either direction between a specified special function register and a specified memory location. It therefore has an effect similar to that of a DMA channel.

Figure 6. Context Switching and Return



Special-Function Registers

The special-function registers (table 2) include the I/O ports, the counters and timers, all registers associated with peripherals, and all of the control and mode registers. They are memory mapped in the top 256 memory addresses and can be addressed either by main memory addressing or by the special one byte sfr addressing. Most can be either read or written, and individual bits within them can be modified or tested with a single instruction.

Table 2. Special-Function Registers

				Access Unit (Bits)			
Address	Register	Symbol	R/W	1	8	16	State after RESET
FF00H	Port 0	P0	R/W	X	X	—	Undefined
FF02H	Port 2	P2	R	—	X	—	Undefined
FF03H	Port 3	P3	R/W	X	X	—	Undefined
FF04H	Port 4	P4	R/W	X	X	—	Undefined
FF05H	Port 5	P5	R/W	X	X	—	Undefined
FF07H	Port 7	P7	R	—	X	—	Undefined
FF08H	Port 8	P8	R/W	X	X	—	Undefined
FF09H	Port 9	P9	R/W	X	X	—	Undefined
FF0AH-FF0BH	Free-running counter (lower 16 bits)*	TM0LW	R	—	—	X	0000H
FF10H-FF11H	Capture register X0 (lower 16 bits)*	CTX0LW	R	—	—	X	Undefined
FF12H-FF13H	Capture register 01 (lower 16 bits)*	CT01LW	R	—	—	X	Undefined
FF14H-FF15H	Capture register 02 (lower 16 bits)*	CT02LW	R	—	—	X	Undefined

Table 2. Special-Function Registers (cont)

				Access Unit (Bits)			
Address	Register	Symbol	R/W	1	8	16	State after RESET
FF16H-FF17H	Capture register 03 (lower 16 bits)*	CT03LW	R	—	—	X	Undefined
FF18H-FF19H	Capture/compare register X0 (lower 16 bits)*	CCX0LW	R/W	—	—	X	Undefined
FF1AH-FF1BH	Capture/compare register 01 (lower 16 bits)*	CC01LW	R/W	—	—	X	Undefined
FF20H	Port 0 mode register	PM0	W	—	X	—	FFH
FF23H	Port 3 mode register	PM3	W	—	X	—	xxx1 1111B
FF25H	Port 5 mode register	PM5	W	—	X	—	FFH
FF28H	Port 8 mode register	PM8	W	—	X	—	xx11 1111B
FF29H	Port 9 mode register	PM9	W	—	X	—	xxxx 1111B
FF2AH-FF2BH	Free running counter (high 16 bits)*	TM0UW	R	—	—	X	0000H
FF2CH-FF2DH	Timer register 1 (lower 16 bits)*	TM1	R	—	—	X	0000H
FF30H-FF31H	Capture register X0 (High 16 bits)*	CTX0UW	R	—	—	X	Undefined
FF32H-FF33H	Capture register 01 (High 16 bits)*	CT01UW	R	—	—	X	Undefined
FF34H-FF35H	Capture register 02 (High 16 bits)*	CT02UW	R	—	—	X	Undefined
FF36H-FF37H	Capture register 03 (High 16 bits)*	CT03UW	R	—	—	X	Undefined
FF38H-FF39H	Capture/compare register X0 (high 16 bits)*	CCX0UW	R/W	—	—	X	Undefined
FF3AH-FF3BH	Capture/compare register 01 (high 16 bits)*	CC01UW	R/W	—	—	X	Undefined
FF40H	Port 0 mode control register	PMC0	W	—	X	—	00H
FF41H	Real-time output port set register	RTPS	R/W	X	X	—	00H
FF43H	Port 3 mode control register	PMC3	W	—	X	—	xxx0 0000B
FF48H	Port 8 mode control register	PMC8	W	—	X	—	xx00 0000B
FF4CH-FF4DH	Baud rate generator	BRG	R/W	—	—	X	Undefined
FF60H	Real-time output port register	RTP	R/W	X	X	—	Undefined
FF61H	Real-time output port reset register	RTPR	R/W	X	X	—	00H
FF62H	Port read control register	PRDC	R/W	X	X	—	00H
FF68H	A/D converter mode register	ADM	R/W	X	X	—	00H
FF6AH	A/D converter result register (16-bit access)	ADCR	R	—	—	X	Undefined
FF6BH	A/D converter result register (high 8 bits)	ADCRH	R	—	X	—	Undefined
FF70H-FF71H	Compare register 00	CM00	R/W	—	—	X	Undefined
FF72H-FF73H	Compare register 01	CM01	R/W	—	—	X	Undefined
FF74H-FF75H	Compare register 02	CM02	R/W	—	—	X	Undefined
FF76H-FF77H	Compare register 03	CM03	R/W	—	—	X	Undefined
FF7CH-FF7DH	Compare register 10	CM10	R/W	—	—	X	Undefined
FF7EH-FF7FH	Compare register 11	CM11	R/W	—	—	X	Undefined
FF80H	Clock synchronized serial interface mode register	CSIM	R/W	X	X	—	00H
FF82H	Serial bus interface control register	SBIC	R/W	X	X	—	00H

Table 2. Special-Function Registers (cont)

				Access Unit (Bits)			
Address	**Register**	**Symbol**	**R/W**	**1**	**8**	**16**	**State after RESET**
FF86H	Serial I/O shift register	SIO	R/W	X	X	—	Undefined
FF88H	Asynchronous serial interface mode register	ASIM	R/W	X	X	—	80H
FF8AH	Asynchronous serial interface status register	ASIS	R	—	X	—	00H
FF8CH	Serial receive buffer: UART	RXB	R	—	X	—	Undefined
FF8EH	Serial transmit shift register: UART	TXS	W	—	X	—	Undefined
FFB0H	Timer control register	TMC	R/W	X	X	—	00H
FFB1H	Baud rate generator mode register	BRGM	R/W	X	X	—	00H
FFB2H	Prescalar mode register	PRM	R/W	X	X	—	00H
FFB8H	Timer output control register 0	TOC0	R/W	X	X	—	00H
FFB9H	Timer output control register 1	TOC1	R/W	X	X	—	00H
FFBFH	Real-time pulse unit mode register	RPUM	R/W	X	X	—	00H
FFC0H	Standby control register	STBC	R/W**	X	X	—	0000 X000B
FFC1H	CPU control word	CCW	R/W	X	X	—	00H
FFC2H	Watchdog timer mode register	WDM	R/W**	X	X	—	00H
FFC4H	Memory extension mode register	MM	R/W	X	X	—	00H
FFC6H	Programmable wait control register	PWC	R/W	X	X	—	22H
FFC9H	Fetch cycle control register	FCC	R/W	X	X	—	00H
FFD0H-FFDFH	External access area		R/W	X	X	—	Undefined
FFE0H	Interrupt request flag register 0L	IF0L/IF0	R/W	X	X	X	00H
FFE1H	Interrupt request flag register 0H	IF0H	R/W	X	X	—	00H
FFE2H	Interrupt request flag register 1L	IF1L/ IF1	R/W	X	X	X	00H
FFE4H	Interrupt mask flag register 0L	MK0L/ MK0	R/W	X	X	X	FFH
FFE5H	Interrupt mask flag register 0H	MK0H	R/W	X	X	—	FFH
FFE6H	Interrupt mask flag register 1L	MK1L/ MK1	R/W	X	X	X	xxxx x111B
FFE8H	Priority selection buffer register 0L	PB0L/ PB0	R/W	X	X	X	00H
FFE9H	Priority selection buffer register 0H	PB0H	R/W	X	X	—	00H
FFEAH	Priority selection buffer register 1L	PB1L/ PB1	R/W	X	X	X	00H
FFECH	Interrupt service mode selection register 0L	ISM0L/ ISM0	R/W	X	X	X	00H*
FFEDH	Interrupt service mode selection register 0H	ISM0H	R/W	X	X	—	00H
FFEEH	Interrupt service mode selection register 1L	ISM1L/ ISM1	R/W	X	X	X	00H
FFF0H	Context switch enable register 0L	CSE0L/ CSE0	R/W	X	X	X	00H
FFF1H	Context switch enable register 0H	CSE0H	R/W	X	X	—	00H

Table 2. Special-Function Registers (cont)

				Access Unit (Bits)			
Address	**Register**	**Symbol**	**R/W**	**1**	**8**	**16**	**State after RESET**
FFF2H	Context switch enable register 1L	CSE1L/ CSE1	R/W	X	X	X	00H
FFF4H	External interrupt mode register 0	INTM0	R/W	X	X	—	00H
FFF5H	External interrupt mode register 1	INTM1	R/W	X	X	—	00H
FFF8H	In-service priority register	ISPR	R	—	X	—	00H
FFF9H	Priority selection register	PRSL	R/W	X	X	—	00H

* Lower or upper 16 bits of an 18-bit register.

** Protected location: special instruction required for write.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings

$T_A = 25°C$

Supply voltage, V_{DD}	−0.5 to +7.0 V
Supply voltage, AV_{DD}	−0.5 to V_{DD} +0.5 V
Supply voltage, AV_{SS}	−0.5 to +0.5 V
Input voltage, V_I	-0.5 to V_{DD}+0.5 V
Output voltage, V_O	−0.5 to V_{DD}+0.5 V
Reference input voltage, AV_{REF} $f_{XX} \leq$ 16MHz	−0.5 to AV_{DD} +0.3 V
Output current, low; I_{OL} Each output pin	4.0 mA
Total	90 mA
Output current, high; I_{OH} Each output pin	−1.0 mA
Total	−20 mA
Operating temperature, T_{OPT}	−10 to +70°C
Storage temperature, T_{STG}	−65 to +150°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage.

Operating Conditions

Oscillator Frequency	T_A	V_{DD}
8 MHz ≤ f_{XX} ≤ 16 MHz	−10 to +70°C	+5.0 V ± 10%

Capacitance

T_A = 25°C; V_{DD} = V_{SS} = 0 V

Parameter	Symbol	Max	Unit	Conditions
Input pin capacitance	C_I	10	pF	f = 1 MHz; unmeasured pins returned to 0 V
Output pin capacitance	C_O	20	pF	
I/O pin capacitance	C_{IO}	20	pF	

DC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input voltage, low	V_{IL}	0		0.8	V	
Input voltage, high	V_{IH1}	2.2			V	Note 1
	V_{IH2}	$0.8V_{DD}$			V	Note 2
Output voltage, low	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output voltage, high	V_{OH}	V_{DD}−1.0			V	I_{OH} = −400 μA
Input leakage current	I_{LI}			±10	μA	0 V ≤ V_I ≤ V_{DD}
Output leakage current	I_{LO}			±10	μA	0 V ≤ V_O ≤ V_{DD}
V_{DD} supply current	I_{DD1}		40	65	mA	Operating mode
	I_{DD2}		20	35	mA	HALT mode
Data retention voltage	V_{DDDR}	2.5			V	STOP mode
Data retention current	I_{DDDR}		2	10	μA	STOP mode V_{DDDR} = 2.5 V
			10	50	μA	V_{DDDR} = 5.0 V ±10%

Notes:

(1) All except $\overline{\text{RESET}}$, X1, X2, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2, $P2_4$/INTP3, $P2_5$/INTP4, $P2_6$/INTP5, $P2_7$/INTP6/TI, $P3_2$/SB0/SO, $P3_3$/SB1/SI, $P3_4$/$\overline{\text{SCK}}$.

(2) $\overline{\text{RESET}}$, X1, X2, $P2_0$/NMI, $P2_1$/INTP0, $P2_2$/INTP1, $P2_3$/INTP2, $P2_4$/INTP3, $P2_5$/INTP4, $P2_6$/INTP5, $P2_7$/INTP6/TI, $P3_2$/SB0/SO, $P3_3$/SB1/SI, $P3_4$/$\overline{\text{SCK}}$.

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	Min	Max	Unit	Conditions
Normal External Memory Read/Write Operation ***Turbo Access Manager Data Read/Write Operation*** ***Turbo Access Manager Branch Operation (Fetch Pointer ← address)***					
System clock cycle time	t_{CYK}	125	250	ns	Twice the crystal or external clock input period
Address setup time to ASTB ↓	t_{SAST}	32		ns	t_{CYK} = 125 ns
Address hold after ASTB ↓	t_{HSTA}	32		ns	
Address to $\overline{\text{RD}}$ ↓ delay time	t_{DAR}	85		ns	
$\overline{\text{RD}}$ ↓ to address floating	t_{FRA}		0	ns	
Address to data input	t_{DAID}		222	ns	
$\overline{\text{RD}}$ ↓ to data input	t_{DRID1}		112	ns	
ASTB ↓ to $\overline{\text{RD}}$ ↓ delay time	t_{DSTR}	42		ns	
Data hold time from $\overline{\text{RD}}$ ↑	t_{HRID}	0		ns	
$\overline{\text{RD}}$ ↑ to address active	t_{DRA}	37		ns	
$\overline{\text{RD}}$ width low	t_{WRL}	157		ns	
ASTB width, high	t_{WSTH}	37		ns	
Address to $\overline{\text{WR}}$ ↓ delay	t_{DAW}	85		ns	
ASTB ↓ to data output	t_{DSTOD}		102	ns	

AC Characteristics (cont)

Parameter	Symbol	Min	Max	Unit	Conditions
Normal External Memory Read/Write Operation Turbo Access Manager Data Read/Write Operation Turbo Access Manager Branch Operation (Fetch Pointer ← address) (cont)					
$\overline{\text{WR}}$ to data output	t_{DWOD}		40	ns	t_{CYK} = 125 ns
ASTB ↓ to $\overline{\text{WR}}$ ↓ delay	t_{DSTW}	42		ns	
Data setup time to $\overline{\text{WR}}$ ↑	t_{SODW}	147		ns	
Data hold time after $\overline{\text{WR}}$ ↑	t_{HWOD}	32		ns	
$\overline{\text{WR}}$ ↑ to ASTB ↑ delay time	t_{DWST}	42		ns	
$\overline{\text{WR}}$ width, low	t_{WWL}	157		ns	
Opcode Fetch with Turbo Access Manager: Branch and Continuous Fetch					
$\overline{\text{TAS}}$ width, low	t_{WTAL}	37		ns	
$\overline{\text{TAS}}$ width, high	t_{WTAH}	42		ns	
$\overline{\text{TAS}}$ ↑ to data input	t_{DTAID}		55	ns	
TMD ↑ to $\overline{\text{TAS}}$ ↑	t_{DTMRTA}	157		ns	
$\overline{\text{RD}}$ ↓ to data input	t_{DRID2}		65	ns	
$\overline{\text{TAS}}$ setup to ASTB ↓	t_{STAST}	32		ns	
TMD setup to ASTB ↓	t_{STMST}	42		ns	
TMD ↓ to $\overline{\text{TAS}}$ ↑ delay time	t_{DTMFTA}	95		ns	
ASTB ↓ to TMD ↓ delay time	t_{DSTTM}	85		ns	
Data hold after $\overline{\text{TAS}}$ ↑	t_{HTMID}	0		ns	

Serial Port Operation

T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; V_{SS} = 0 V

Parameter	Symbol	Min	Max	Unit	Conditions
$\overline{\text{SCK}}$ cycle time	t_{CYSK}	1		μs	$\overline{\text{SCK}}$ output from internal clock
		1		μs	$\overline{\text{SCK}}$ input from external clock
$\overline{\text{SCK}}$ with low	t_{WSKL}	420		ns	$\overline{\text{SCK}}$ output from internal clock
		420		ns	$\overline{\text{SCK}}$ input from external clock
$\overline{\text{SCK}}$ width high	t_{WSKH}	420		ns	$\overline{\text{SCK}}$ output from internal clock
		420		ns	$\overline{\text{SCK}}$ input from external clock
SI setup time to $\overline{\text{SCK}}$ ↑	t_{SRXSK}	80		ns	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{HSKRX}	80		ns	
$\overline{\text{SCK}}$ ↓ to SO delay time	t_{DSKTX}		210	ns	

Timing Dependent on t_{CYK}

Symbol	Calculation Formula	Min/Max	Unit
t_{SAST}	0.5T - 30	Min	ns
t_{HSTA}	0.5T - 30	Min	ns
t_{DAR}	T - 40	Min	ns
t_{DAID}	(2.5+n)T - 90	Max	ns
t_{DRID1}	(1.5+n)T - 75	Max	ns
t_{DSTR}	0.5T - 20	Min	ns
t_{DRA}	0.5T - 25	Min	ns
t_{WRL}	(1.5+n)T - 30	Min	ns
t_{WSTH}	0.5T - 25	Min	ns
t_{DAW}	T - 40	Min	ns
t_{DSTOD}	0.5T + 40	Max	ns
t_{DSTW}	0.5T - 20	Min	ns
t_{SODW}	1.5T - 40	Min	ns
t_{HWOD}	0.5T - 30	Min	ns
t_{DWST}	0.5T - 20	Min	ns
t_{WWL}	(1.5+n)T - 30	Min	ns
t_{WTAL}	0.5T - 25	Min	ns
t_{WTAH}	0.5T - 20	Min	ns
t_{DTAID}	T - 45	Min	ns
t_{DTMRTA}	1.5T - 30	Min	ns
t_{DRID2}	T - 60	Max	ns
t_{STAST}	0.5T - 30	Min	ns
t_{STMST}	0.5T - 20	Min	ns
t_{DTMFTA}	T - 30	Min	ns
t_{DSTTM}	T - 40	Min	ns

Notes:

(1) n is the number of additional wait cycle specified by the PWC register.

(2) T = t_{CYK} = (ns).

(3) Parameters not included in this table are not dependent on t_{CYK}.

A/D Converter

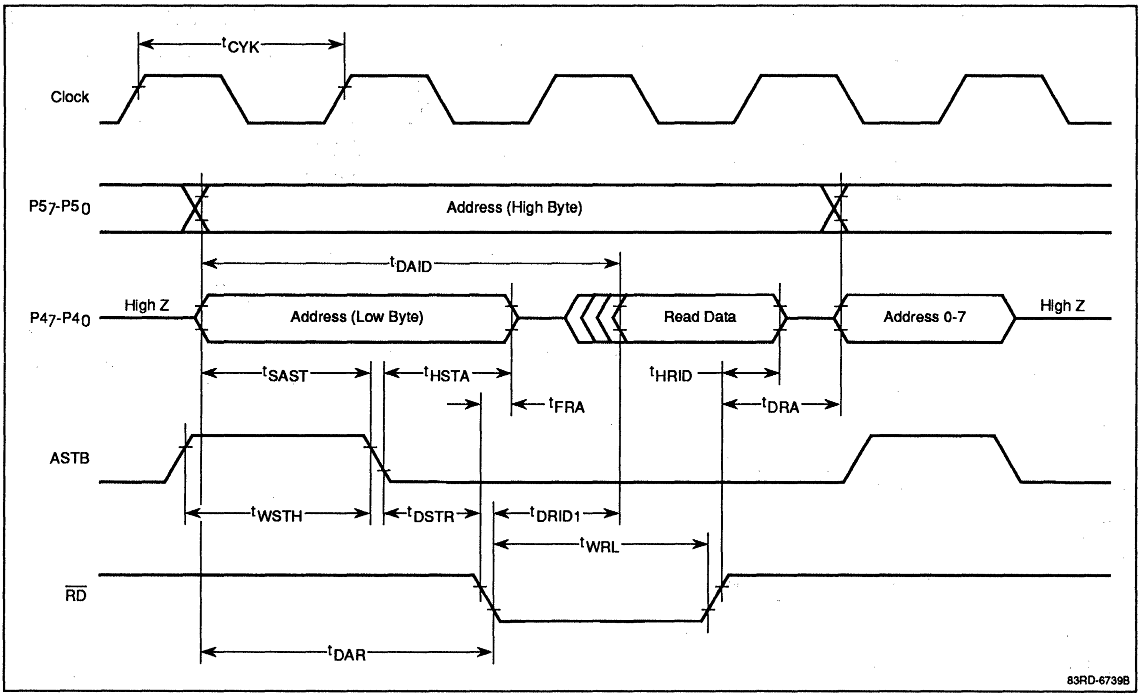
T_A = −10 to +70°C; V_{DD} = +5.0 V ±10%; AV_{SS} = V_{SS} = 0 V; V_{DD} −0.5 V ≤ AV_{DD} ≤ V_{DD}; 3.4 V ≤ AV_{REF} ≤ V_{DD}

Parameter	Symbol	Min	Typ	Max	Unit
Resolution		10			Bit
Relative accuracy				0.2%	FSR
Quantization error				±1/2	LSB
Conversion time	t_{CONV}	144			t_{CYK}
Sampling time	t_{SAMP}	24			t_{CYK}
Zero offset error			±1.5	LSB	
Full scale error			±1.5	LSB	
Linearity error			±1.5	LSB	
Analog input voltage	V_{IAN}	0		AV_{REF}	V
AV_{REF} current	AI_{REF}		1.0	3.0	mA
AV_{DD} current	AI_{DD}		2.0	6.0	mA

7

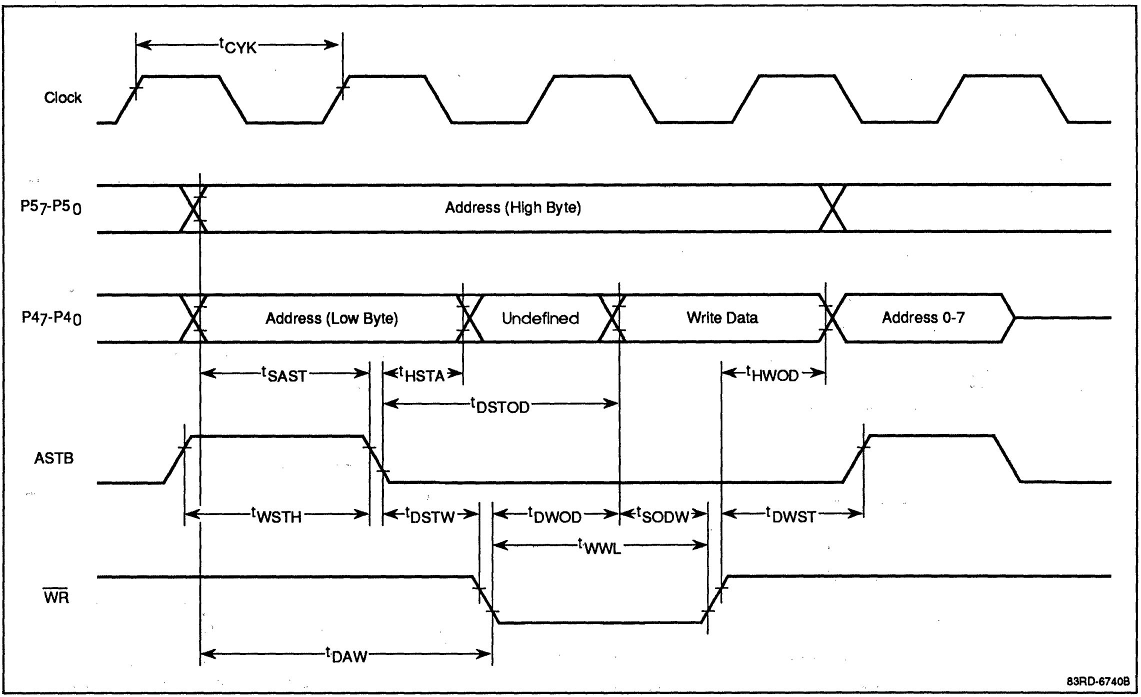
Timing Waveforms

Discontinuous Read Cycle



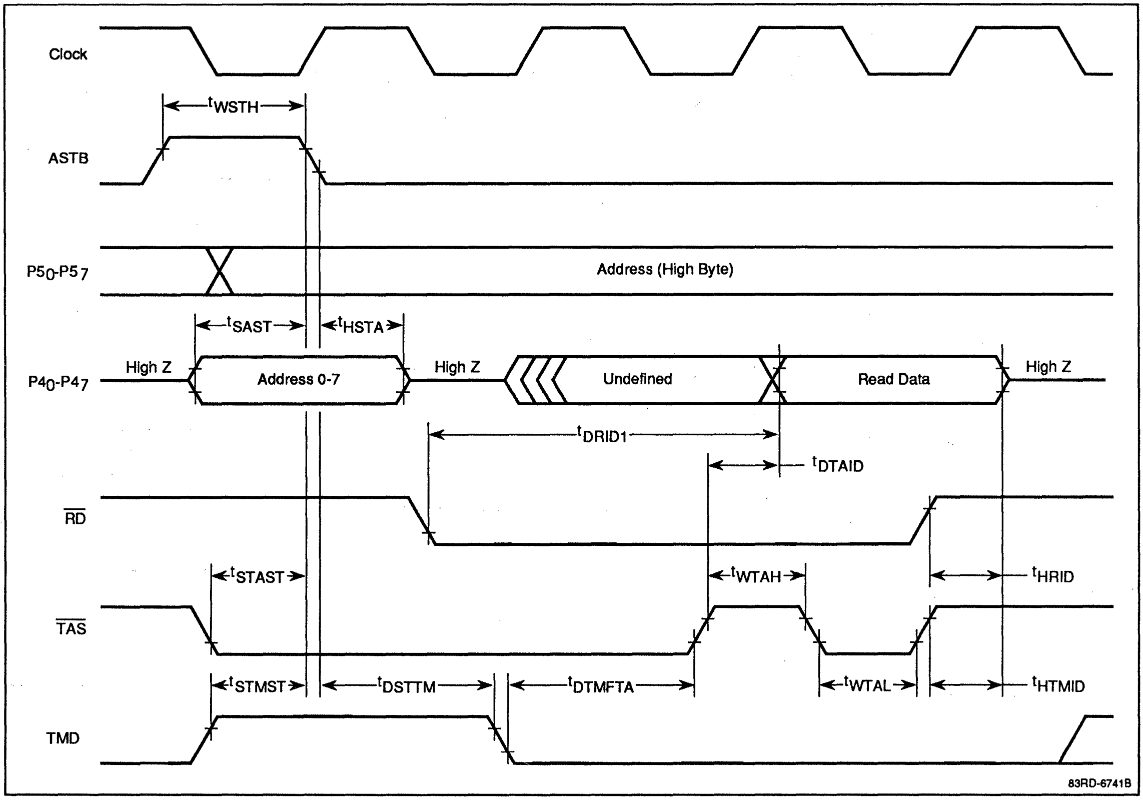
Timing Waveforms (cont)

Discontinuous Write Cycle



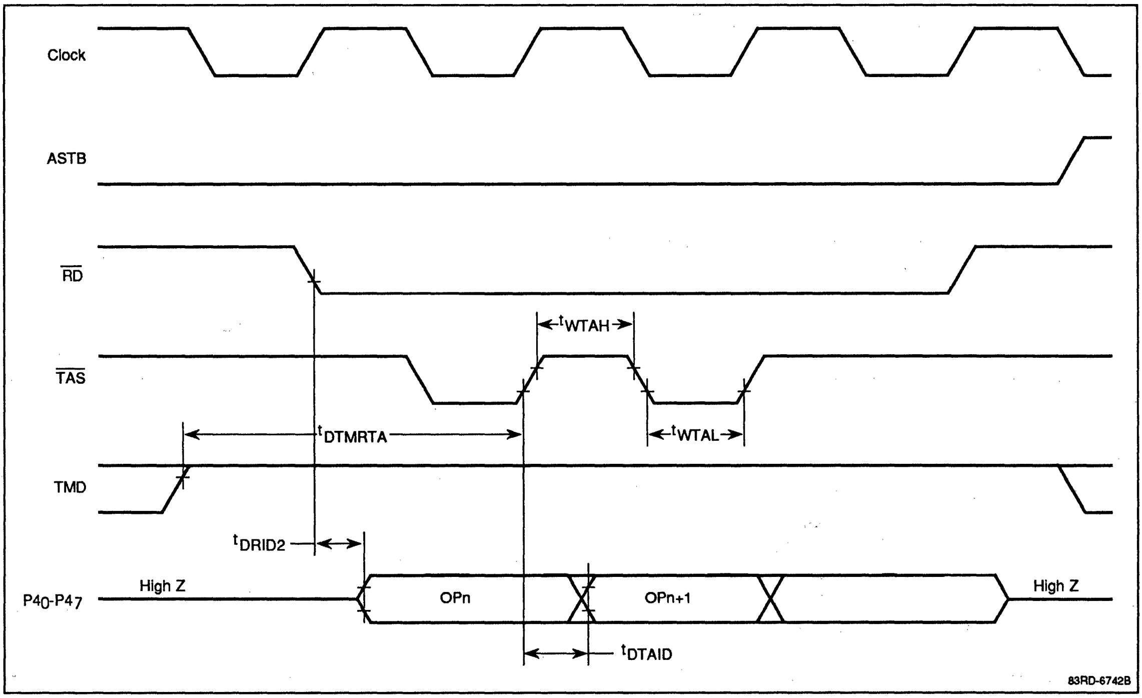
Timing Waveforms (cont)

Branch Cycle, TAM Interface

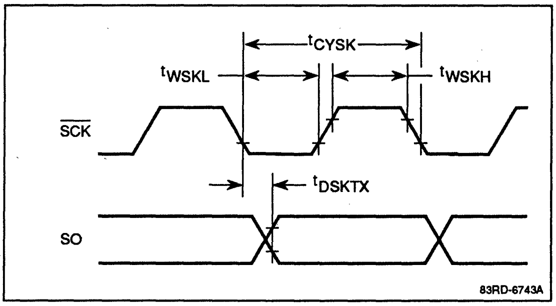


Timing Waveforms (cont)

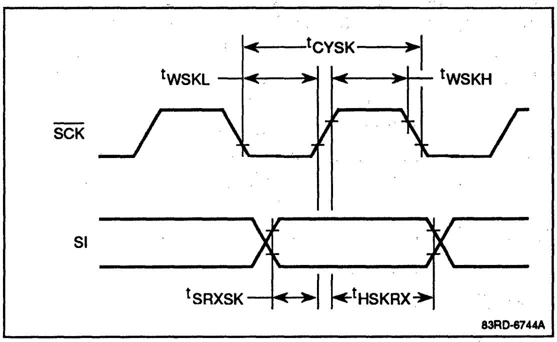
Continuous Instruction Fetch Cycle, TAM Interface



Data Transmit, Serial Port



Data Receive, Serial Port



INSTRUCTION SET

Addressing

On-chip RAM byte location FE20H through FEFFH can be addressed by saddr addressing, in which the machine code specifies the low-order byte only. This addressing mode is also used to address the first 20H special function registers, those with addresses FF00H through FF1FH. Similarly, saddrp addressing is used to specify 16-bit word locations within the same area. The saddrp addresses must be even.

When both source and destination are registers, the destination designation appears in the machine code before the source designation. Similarity, if source and destination are both saddr or saddrp, the destination appears before the source. Both saddr and saddrp addresses are expressed as offsets from either FE00H or FF00H.

Timing

Access to on-chip ROM and to main RAM (FE00H-FEFFH) requires one state per byte. Access to on-chip peripheral RAM (FC80H-FDFFH) and to external memory requires a minimum of three states per byte unless the TAM is used. Instructions can be fetched from the TAM at a rate of one state per byte.

Timing of the PUSH and POP Instructions

The post byte used by the PUSH post, PUSHU post, POP post, and POPU post instructions has a bit set for each register pair to be PUSHed or POPped. Bit 0 specifies RP0, bit 1 RP1,..., bit 7 RP7. The PUSH (and PUSHU) and the POP (and POPU) instructions scan the post byte to determine which register pairs are to be PUSHed or POPped. The PUSH (and PUSHU) instructions begin the scan at the high-order end (bit 7), while the POP (and POPU) instructions begin the scan at the low-order end (bit 0). If the stack is in main RAM (0FE00-0FEFF), the timing formulas are:

PUSH: $t = 3 + 4z + 6n$ states
PUSHU: $t = 4 + 4z + 6n$ states
POP: $t = 6 + 4z + 7n$ states
POPU: $t = 8 + 4z + 7n$ states

where n is the number of register pairs to be PUSHed or POPped, and z is the number of zero bits scanned before all remaining bits are zero.

Example: PUSH RP2, RP3: the post byte is 00001100B.

PUSH: $t = 3 + 4 \times 4 + 6 \times 2 = 31$ states
(4 zeros scanned from high-order end)
POP: $t = 6 + 4 \times 2 + 7 \times 2 = 28$ states
(2 zeros scanned from low-order end)

If the stack is in external RAM or peripheral RAM (0FC80-0FDFF) the formulas become:

PUSH: $t = 3 + 4z + (8 + 2w)n$ states
PUSHU: $t = 4 + 4z + (8 + 2w)n$ states
POP: $t = 6 + 4z + (14 + 2w)n$ states
POPU: $t = 8 + 4z + (14 + 2w)n$ states

where w is the number of additional wait states specified in the PWC register. The timing for the PUSH (and PUSHU) instructions is worst case, and it will improve if the external bus is not busy.

Interrupt Service Timing

Operation	States
Interrupt service by context switch	12
Vector Interrupt (stack in main RAM)	17
(stack in any other memory)	31 + 4n

Macroservice Timing

		States	
Operation		Normal End	Software Interrupt
EVCNT		10	12
DTACMP		15	17
BITSHT		17	19
BITLOG		19	19
ADCBUF		16	26
DTADIF	Byte	22	22
	Word	23	23
DATADIF-P	Byte (1)	24	24
	Byte (2)	26+ n	26+ n
	Word (1)	25	25
	Word (2)	30+2n	30+2n
DTADD		24	26
BLKTRS mem → sfr	Byte(1)	20	22
	Byte (2)	22+ n	24+ n
	Word (1)	21	23
	Word (2)	26+2n	28+2n

Macroservice Timing (cont)

Operation		States	
		Normal End	Software Interrupt
BLKTRS sfr → mem	Byte (1)	19	21
	Byte (2)	19	21
	Word (1)	20	22
	Word (2)	20	22

Notes:

(1) Destination is in main RAM (FE00H-FEFFH).

(2) Destination is anywhere but main RAM.

(3) n = number of additional wait states specified in the PWC register.

In the States column of the Instruction Set, the symbol "n" stands for a number as follows.

Operation	Number "n"
Stack	Register pairs operated on
Shift and rotate	Bits shifted or rotated
String	Characters in the string or the number operated upon before the condition is satisfied

In the States column, a number in parentheses for a conditional branch instruction is the number of states used if the branch is not taken.

Opcodes for Memory Addressing Modes

mod / mem	1 0110	1 0111	0 0110	0 1010
	Register Indirect	Base Index	Base	Index
0 0 0	[DE+] *	[DE+A]	[DE+byte]	word [DE]
0 0 1	[HL+] *	[HL+A]	[SP+byte]	word [A]
0 1 0	[DE–] *	[DE+B]	[HL+byte]	word [HL]
0 1 1	[HL–] *	[HL+B]	[UP+byte]	word [B]
1 0 0	[DE] *	[VP+DE]	[VP+byte]	–
1 0 1	[HL] *	[VP+HL]	–	–
1 1 0	[VP]	–	–	–
1 1 1	[UP]	–	–	–

*One-byte instructions: Defined by special OP Code & mem only.

83RD-6984A

Opcodes for Registers

r

R_3	R_2	R_1	R_0	reg
0	0	0	0	R0
0	0	0	1	R1
0	0	1	0	R2
0	0	1	1	R3
0	1	0	0	R4
0	1	0	1	R5
0	1	1	0	R6
0	1	1	1	R7
1	0	0	0	R8
1	0	0	1	R9
1	0	1	0	R10
1	0	1	1	R11
1	1	0	0	R12
1	1	0	1	R13
1	1	1	0	R14
1	1	1	1	R15

r1

R_2	R_1	R_0	reg
0	0	0	R0
0	0	1	R1
0	1	0	R2
0	1	1	R3
1	0	0	R4
1	0	1	R5
1	1	0	R6
1	1	1	R7

r2

C_0	reg
0	C
1	B

rp

P_2	P_1	P_0	reg-pair
0	0	0	RP0
0	0	1	RP1
0	1	0	RP2
0	1	1	RP3
1	0	0	RP4
1	0	1	RP5
1	1	0	RP6
1	1	1	RP7

rp1

Q_2	Q_1	Q_0	reg-pair
0	0	0	RP0
0	0	1	RP4
0	1	0	RP1
0	1	1	RP5
1	0	0	RP2
1	0	1	RP6
1	1	0	RP3
1	1	1	RP7

rp2

S_1	S_0	reg-pair
0	0	VP
0	1	UP
1	0	DE
1	1	HL

83RD-6985A

Flag Indicators

Symbol	Action
(blank)	No change
0	Set to 0
1	Set to 1
X	Set or cleared according to result
P	P/V indicated parity of result
V	P/V indicates arithmetic overflow
R	Restored from saved PSW

Instruction Set Symbols

Symbol	Definition
r	R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15
r1	R0, R1, R2, R3, R4, R5, R6, R7
r2	C, B
rp	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp1	RP0, RP1, RP2, RP3, RP4, RP5, RP6, RP7*
rp2	DE, HL, VP, UP
sfr	Special function register, 8 bits
sfrp	Special function register, 16 bits
post	RP0, RP1, RP2, RP3, RP4, RP5/PSW, RP6, RP7 Bits set to 1 indicate register pairs to be pushed/popped to/from stack; RP5 pushed/popped by PUSH/POP, SP is stack pointer; PSW pushed/popped by PUSHU/POPU, RP5 is stack pointer;
mem	Register indirect: [DE], [HL], [DE+], [HL+], [DE–], [HL–], [VP], [UP] Base Index Mode: [DE+A], [HL+A], [DE+B], [HL+B], [VP+DE], [VP+HL] Base Mode: [DE+byte], [HL+byte], [VP+byte], [UP+byte], [SP+byte] Index Mode: word [A], word [B], word [DE], word [HL]
saddr	FE20-FF1FH: Immediate byte addresses one byte in RAM, or label
saddrp	FE20-FF1FH: Immediate byte (bit 0=0) addresses one word in RAM, or label

Instruction Set Symbols (cont)

Symbol	Definition
word	16 bits of immediate date
byte	8 bits of immediate data
jdisp	8-bit two's complement displacement (immediate data)
f_0-f_{10}	Eleven bits of immediate data corresponding to addr11
t_0-t_4	Five bits of immediate data corresponding to addr5
bit	3 bits of immediate data (bit position in byte), or label
n	3 bits of immediate data
!addr16	16-bit absolute address specified by an immediate address or label
$addr16	Relative branch address [(PC)+jdisp] or label
addr16	16-bit address
!addr11	11-bit immediate address or label
addr11	0800H-0FFFH: 0800H + (11-bit immediate address), or label
addr5	0040H-007EH: 0040H + 2 X (5-bit immediate address), or label
A	A register
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
R0-R15	Register 0 to register 15
AX	Register pair AX (16-bit accumulator)
BC	Register pair BC
DE	Register pair DE
HL	Register pair HL

Instruction Set Symbols (cont)

Symbol	Definition
RP0-RP7	Register pair 0 to register pair 7
PC	Program counter
SP	Stack pointer
UP	User stack pointer (RP5)
PSW	Program status word
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
P/V	Parity/overflow flag
S	Sign flag
SUB	Subtract flag
TPF	Table position flag
RBS	Register bank select flag
RSS	Register set select flag
IE	Interrupt enable flag
STBC	Standby control register
WDM	Watchdog timer mode register

Symbol	Definition
( )	Contents of the location whose address is within parentheses; (+) and (–) indicate that the address is incremented after or decremented after it is used
(( ))	Contents of the memory location defined by the quantity within the sets of parentheses
xxH	Hexadecimal quantity
X_H, X_L	High-order 8 bits and low-order 8 bits of X

* rp and rp1 describe the same registers but generate different machine code.

Instruction Set

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
8-Bit Data Transfer																	
MOV	r1, #byte	r1 ← byte	2	2						1	0	1	1	1	R_2	R_1	R_0
										Data							
	saddr, #byte	(saddr) ← byte	3	3						0	0	1	1	1	0	1	0
										Saddr-offset							
										Data							
	sfr**, #byte	sfr ← byte	3	6						0	0	1	0	1	0	1	1
										Sfr-offset							
										Data							
	r, r1	r ← r1	2	3						0	0	1	0	0	1	0	0
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, r1	A ← r1	1	2						1	1	0	1	0	R_2	R_1	R_0
	A, saddr	A ← (saddr)	2	3						0	0	1	0	0	0	0	0
										Saddr-offset							
	saddr, A	(saddr) ← A	2	3						0	0	1	0	0	0	1	0
										Saddr-offset							
	saddr, saddr	(saddr) ← (saddr)	3	4						0	0	1	1	1	0	0	0
										Saddr-offset							
										Saddr-offset							
	A, sfr	A ← sfr	3	4						0	0	0	1	0	0	0	0
										Sfr-offset							
	sfr, A	sfr ← A	2	6						0	0	0	1	0	0	1	0
										Sfr-offset							
	A, mem*	A ← (mem)	1	6						0	1	0	1	1	mem		
	A, mem	A ← (mem)	2-4	8-10						0	0	0	mod				
										0	mem			0	0	0	0
										Low Offset							
										High Offset							
	mem, A*	(mem) ← A	1	4						0	1	0	1	0	mem		
	mem, A	(mem) ← A	2-4	6-8						0	0	0	mod				
										1	mem			0	0	0	0
										Low Offset							
										High Offset							
	A, [saddrp]	A ← ((saddrp))	2	6						0	0	0	1	1	0	0	0
										Saddr-offset							
	[saddrp], A	((saddrp)) ← A	2	4						0	0	0	1	1	0	0	1
										Saddr-offset							
	A, !addr16	A ← (addr16)	4	6						0	0	0	0	1	0	0	1
										1	1	1	1	0	0	0	0
										Low Addr							
										High Addr							

* One byte move instruction when [DE], [HL], [DE+], [DE–], [HL+], or [HL–] is specified for mem.

** A special instruction is used to write to STBC and WDM.

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
8-Bit Data Transfer (cont)																	
MOV (cont)	!addr16, A	(addr16) ← A	4	5						0	0	0	0	1	0	0	1
										1	1	1	1	0	0	0	1
										Low Addr							
										High Addr							
	PSWL, #byte	PSW_L ← byte	3	6	X	X	X	X	X	0	0	1	0	1	0	1	1
										1	1	1	1	1	1	1	0
										Data							
	PSWH, #byte	PSW_H ← byte	3	6						0	0	1	0	1	0	1	1
										1	1	1	1	1	1	1	1
										Data							
	PSWL, A	PSW_L ← A	2	6	X	X	X	X	X	0	0	0	1	0	0	1	0
										1	1	1	1	1	1	1	0
	PSWH, A	PSW_H ← A	2	6						0	0	0	1	0	0	1	0
										1	1	1	1	1	1	1	1
	A, PSWL	A ← PSW_L	2	6						0	0	0	1	0	0	0	0
										1	1	1	1	1	1	1	0
	A, PSWH	A ← PSW_H	2	6						0	0	0	1	0	0	0	0
										1	1	1	1	1	1	1	1
XCH	A, r1	A ↔ r1	1	4						1	1	0	1	1	R_2	R_1	R_0
	r, r1	r ↔ r1	2	4						0	0	1	0	0	1	0	1
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, mem	A ↔ (mem)	2-4	9-11						0	0	0	mod				
										0	mem			0	1	0	0
										Low Offset							
										High Offset							
	A, saddr	A ↔ (saddr)	2	5						0	0	1	0	0	0	0	1
										Saddr-offset							
	A, sfr	A ↔ sfr	3	13						0	0	0	0	0	0	0	1
										0	0	1	0	0	0	0	1
										Sfr-offset							
	A, [saddrp]	A ↔ ((saddrp))	2	7						0	0	1	0	0	0	1	1
										Saddr-offset							
	saddr, saddr	(saddr) ↔ (saddr)	3	8						0	0	1	1	1	0	0	1
										Saddr-offset							
										Saddr-offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
16-Bit Data Transfer																	
MOVW	rp1, #word	rp1 ← word	3	3						0	1	1	0	0	Q_2	Q_1	Q_0
										Low Byte							
										High Byte							
	saddrp, #word	(saddrp) ← word	4	4						0	0	0	0	1	1	0	0
										Saddr-offset							
										Low Byte							
										High Byte							
	sfrp, #word	sfrp ← word	4	7						0	0	0	0	1	0	1	1
										Sfr-offset							
										Low Byte							
										High Byte							
	rp, rp1	rp ← rp1	2	3						0	0	1	0	0	1	0	0
										P_2	P_1	P_0	0	1	Q_2	Q_1	Q_0
	AX, saddrp	AX ← (saddrp)	2	3						0	0	0	1	1	1	0	0
										Saddr-offset							
	saddrp, AX	(saddrp) ← AX	2	3						0	0	0	1	1	0	1	0
										Saddr-offset							
	saddrp, saddrp	(saddrp) ← (saddrp)	3	4						0	0	1	1	1	1	0	0
										Saddr-offset							
										Saddr-offset							
	AX, sfrp	AX ← sfrp	2	6						0	0	0	1	0	0	0	1
										Sfr-offset							
	sfrp, AX	sfrp ← AX	2	6						0	0	0	1	0	0	1	1
										Sfr-offset							
	rp1, !addr16	rp1 ← (addr16)	4	7						0	0	0	0	1	0	0	1
										1	0	0	0	0	Q_2	Q_1	Q_0
										Low Addr							
										High Addr							
	!addr16, rp1	(addr16) ← rp1	4	5						0	0	0	0	1	0	0	1
										1	0	0	1	0	Q_2	Q_1	Q_0
										Low Addr							
										High Addr							
	AX, mem	AX ← (mem)	2-4	6-10						0	0	0	mod				
										0	mem			0	0	0	1
										Low-offset							
										High-offset							
	mem, AX	(mem) ← AX	2-4	4-8						0	0	0	mod				
										1	mem			0	0	0	1
										Low-offset							
										High-offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
16-Bit Data Transfer (cont)																	
XCHW	AX, saddrp	AX ↔ (saddrp)	2	5						0	0	0	1	1	0	1	1
										Saddr-offset							
	AX, sfrp	AX ↔ sfrp	3	13						0	0	0	0	0	0	0	1
										0	0	0	1	1	0	1	1
										Sfr-offset							
	saddrp, saddrp	(saddrp) ↔ (saddrp)	3	8						0	0	1	0	1	0	1	0
										Saddr-offset							
										Saddr-offset							
	rp, rp1	rp ↔ rp1	2	4						0	0	1	0	0	1	0	1
										P_2	P_1	P_0	0	1	Q_2	Q_1	Q_0
	AX, mem	AX ↔ (mem)	2-4	9-11						0	0	0	mod				
										0	mem			0	1	0	1
										Low-offset							
										High-offset							
8-Bit Arithmetic																	
ADD	A, #byte	A, CY ← A + byte	2	2	X	X	X	V	X	1	0	1	0	1	0	0	0
										Data							
	saddr, #byte	(saddr), CY ← (saddr) + byte	3	4	X	X	X	V	X	0	1	1	0	1	0	0	0
										Saddr-offset							
										Data							
	sfr, #byte	sfr, CY ← sfr + byte	4	12	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	1	1	0	1	0	0	0
										Sfr-offset							
										Data							
	r, r1	r, CY ← r + r1	2	3	X	X	X	V	X	1	0	0	0	1	0	0	0
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A, CY ← A + (saddr)	2	4	X	X	X	V	X	1	0	0	1	1	0	0	0
										Saddr-offset							
	A, sfr	A, CY ← A + sfr	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										1	0	0	1	1	0	0	0
										Sfr-offset							
	saddr, saddr	(saddr), CY ← (saddr) + (saddr)	3	5	X	X	X	V	X	0	1	1	1	1	0	0	0
										Saddr-offset							
										Saddr-offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
8-Bit Arithmetic (cont)																	
ADD (cont)	A, mem	A, CY ← A + (mem)	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										0	mem			1	0	0	0
										Low Offset							
										High Offset							
	mem, A	(mem), CY ← (mem) + A	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										1	mem			1	0	0	0
										Low Offset							
										High Offset							
ADDC	A, #byte	A, CY ← A + byte + CY	2	2	X	X	X	V	X	1	0	1	0	1	0	0	1
										Data							
	saddr, #byte	(saddr), CY ← (saddr) + byte + CY	3	4	X	X	X	V	X	0	1	1	0	1	0	0	1
										Saddr-offset							
										Data							
	sfr, #byte	sfr, CY ← sfr + byte + CY	4	12	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	1	1	0	1	0	0	1
										Sfr-offset							
										Data							
	r, r1	r, CY ← r + r1 + CY	2	3	X	X	X	V	X	1	0	0	0	1	0	0	1
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A, CY ← A + (saddr) + CY	2	4	X	X	X	V	X	1	0	0	1	1	0	0	1
										Saddr-offset							
	A, sfr	A, CY ← A + sfr + CY	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										1	0	0	1	1	0	0	1
										Sfr-offset							
	saddr, saddr	(saddr), CY ← (saddr) + (saddr) + CY	3	5	X	X	X	V	X	0	1	1	1	1	0	0	1
										Saddr-offset							
										Saddr-offset							
	A, mem	A, CY ← A + (mem) + CY	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										0	mem			1	0	0	1
										Low Offset							
										High Offset							
	mem, A	(mem), CY ← (mem) + A + CY	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										1	mem			1	0	0	1
										Low Offset							
										High Offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
8-Bit Arithmetic (cont)																	
SUB	A, #byte	A, CY ← A – byte	2	2	X	X	X	V	X	1	0	1	0	1	0	1	0
										Data							
	saddr, #byte	(saddr), CY ← (saddr) – byte	3	4	X	X	X	V	X	0	1	1	0	1	0	1	0
										Saddr-offset							
										Data							
	sfr, #byte	sfr, CY ← sfr – byte	4	12	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	1	1	0	1	0	1	0
										Sfr-offset							
										Data							
	r, r1	r, CY ← r – r1	2	3	X	X	X	V	X	1	0	0	0	1	0	1	0
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A, CY ← A – (saddr)	2	4	X	X	X	V	X	1	0	0	1	1	0	1	0
										Saddr-offset							
	A, sfr	A, CY ← A – sfr	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										1	0	0	1	1	0	1	0
										Sfr-offset							
	saddr, saddr	(saddr), CY ← (saddr) – (saddr)	3	5	X	X	X	V	X	0	1	1	1	1	0	1	0
										Saddr-offset							
										Saddr-offset							
	A, mem	A, CY ← A – (mem)	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										0	mem			1	0	1	0
										Low Offset							
										High Offset							
	mem, A	(mem), CY ← (mem) – A	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										1	mem			1	0	1	0
										Low Offset							
										High Offset							
SUBC	A, #byte	A, CY ← A – byte – CY	2	2	X	X	X	V	X	1	0	1	0	1	0	1	1
										Data							
	saddr, #byte	(saddr), CY ← (saddr) – byte – CY	3	4	X	X	X	V	X	0	1	1	0	1	0	1	1
										Saddr-offset							
										Data							
	sfr, #byte	sfr, CY ← sfr – byte – CY	4	12	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	1	1	0	1	0	1	1
										Sfr-offset							
										Data							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	**Operand**	**Operation**	**Bytes**	**States**	**S**	**Z**	**AC**	**P/V**	**CY**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
8-Bit Arithmetic (cont)																	
SUBC	r, r1	r, CY ← r−r1−CY	2	3	X	X	X	V	X	1	0	0	0	1	0	1	1
(cont)										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A, CY ← A−(saddr)−CY	2	4	X	X	X	V	X	1	0	0	1	1	0	1	1
										Saddr-offset							
	A, sfr	A, CY ← A−sfr−CY	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										1	0	0	1	1	0	1	1
										Sfr-offset							
	saddr, saddr	(saddr), CY ← (saddr)−(saddr)−CY	3	5	X	X	X	V	X	0	1	1	1	1	0	1	1
										Saddr-offset							
										Saddr-offset							
	A, mem	A, CY ← A−(mem)−CY	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										0	mem			1	0	1	1
										Low Offset							
										High Offset							
	mem, A	(mem), CY ← (mem)−A−CY	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										1	mem			1	0	1	1
										Low Offset							
										High Offset							
8-Bit Logic																	
AND	A, #byte	A ← A AND byte	2	2	X	X		P		1	0	1	0	1	1	0	0
										Data							
	saddr, #byte	(saddr) ← (saddr) AND byte	3	4	X	X		P		0	1	1	0	1	1	0	0
										Saddr-offset							
										Data							
	sfr, #byte	sfr ← sfr AND byte	4	12	X	X		P		0	0	0	0	0	0	0	1
										0	1	1	0	1	1	0	0
										Sfr-offset							
										Data							
	r, r1	r ← r AND r1	2	3	X	X		P		1	0	0	0	1	1	0	0
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A ← A AND (saddr)	2	4	X	X		P		1	0	0	1	1	1	0	0
										Saddr-offset							
	A, sfr	A ← A AND sfr	3	9	X	X		P		0	0	0	0	0	0	0	1
										1	0	0	1	1	1	0	0
										Sfr-offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
8-Bit Logic (cont)																	
AND (cont)	saddr, saddr	(saddr) ← (saddr) AND (saddr)	3	5	X	X		P		0	1	1	1	1	1	0	0
										Saddr-offset							
										Saddr-offset							
	A, mem	A ← A AND (mem)	2-4	8-9	X	X		P		0	0	0		mod			
										0	mem			1	1	0	0
										Low Offset							
										High Offset							
	mem, A	(mem) ← (mem) AND A	2-4	8-9	X	X		P		0	0	0		mod			
										1	mem			1	1	0	0
										Low Offset							
										High Offset							
OR	A, #byte	A ← A OR byte	3	4	X	X		P		1	0	1	0	1	1	1	0
										Data							
	saddr, #byte	(saddr) ← (saddr) OR byte	3	4	X	X		P		0	1	1	0	1	1	1	0
										Saddr-offset							
										Data							
	sfr, #byte	sfr ← sfr OR byte	4	12	X	X		P		0	0	0	0	0	0	0	
										0	1	1	0	1	1	1	0
										Sfr-offset							
										Data							
	r, r1	r ← r OR r1	2	3	X	X		P		1	0	0	0	1	1	1	0
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A ← A OR (saddr)	2	4	X	X		P		1	0	0	1	1	1	1	0
										Saddr-offset							
	A, sfr	A ← A OR sfr	3	9	X	X		P		0	0	0	0	0	0	0	1
										1	0	0	1	1	1	1	0
										Sfr-offset							
	saddr, saddr	(saddr) ← (saddr) OR (saddr)	3	5	X	X		P		0	1	1	1	1	1	1	0
										Saddr-offset							
										Saddr-offset							
	A, mem	A ← A OR (mem)	2-4	8-9	X	X		P		0	0	0		mod			
										0	mem			1	1	1	0
										Low Offset							
										High Offset							
	mem, A	(mem) ← (mem) OR A	2-4	8-9	X	X		P		0	0	0		mod			
										1	mem			1	1	1	0
										Low Offset							
										High Offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
8-Bit Logic (cont)																	
XOR	A, #byte	A ← A XOR byte	2	2	X	X		P		1	0	1	0	1	1	0	1
										Data							
	saddr, #byte	(saddr) ← (saddr) XOR byte	3	4	X	X		P		0	1	1	0	1	1	0	1
										Saddr-offset							
										Data							
	sfr, #byte	sfr ← sfr XOR byte	4	12	X	X		P		0	0	0	0	0	0	0	1
										0	1	1	0	1	1	0	1
										Sfr-offset							
										Data							
	r, r1	r ← r XOR r1	2	3	X	X		P		1	0	0	0	1	1	0	1
										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A ← A XOR (saddr)	2	4	X	X		P		1	0	0	1	1	1	0	1
										Saddr-offset							
	A, sfr	A ← A XOR sfr	3	9	X	X		P		0	0	0	0	0	0	0	1
										1	0	0	1	1	1	0	1
										Sfr-offset							
	saddr, saddr	(saddr) ← (saddr) XOR (saddr)	3	5	X	X		P		0	1	1	1	1	1	0	1
										Saddr-offset							
										Saddr-offset							
	A, mem	A ← A XOR (mem)	2-4	8-9	X	X		P		0	0	0	mod				
										0	mem			1	1	0	1
										Low Offset							
										High Offset							
	mem, A	(mem) ← (mem) XOR A	2-4	8-9	X	X		P		0	0	0	mod				
										1	mem			1	1	0	1
										Low Offset							
										High Offset							
CMP	A, #byte	A − byte	2	2	X	X	X	V	X	1	0	1	0	1	1	1	1
										Data							
	saddr, #byte	(saddr) − byte	3	4	X	X	X	V	X	0	1	1	0	1	1	1	1
										Saddr-offset							
										Data							
	sfr, #byte	sfr − byte	4	12	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	1	1	0	1	1	1	1
										Sfr-offset							
										Data							

Instruction Set (cont)

8-Bit Logic (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
CMP	r, r1	r − r1	2	3	X	X	X	V	X	1	0	0	0	1	1	1	1
(cont)										R_3	R_2	R_1	R_0	0	R_2	R_1	R_0
	A, saddr	A − (saddr)	2	4	X	X	X	V	X	1	0	0	1	1	1	1	1
										Saddr-offset							
	A, sfr	A − sfr	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										1	0	0	1	1	1	1	1
										Sfr-offset							
	saddr, saddr	(saddr) − (saddr)	3	5	X	X	X	V	X	0	1	1	1	1	1	1	1
										Saddr-offset							
										Saddr-offset							
	A, mem	A − (mem)	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										0	mem			1	1	1	1
										Low Offset							
										High Offset							
	mem, A	(mem) − A	2-4	8-9	X	X	X	V	X	0	0	0	mod				
										1	mem			1	1	1	1
										Low Offset							
										High Offset							

16-Bit Arithmetic

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
ADDW	AX, #word	AX, CY ← AX + word	3	3	X	X	X	V	X	0	0	1	0	1	1	0	1
										Low Byte							
										High Byte							
	saddrp, #word	(saddrp), CY ← (saddrp) + word	4	5	X	X	X	V	X	0	0	0	0	1	1	0	1
										Saddr-offset							
										Low Byte							
										High Byte							
	sfrp, #word	sfrp, CY ← sfrp + word	5	10	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	0	0	0	1	1	0	1
										Sfr-offset							
										Low Byte							
										High Byte							
	rp, rp1	rp, CY ← rp + rp1	2	3	X	X	X	V	X	1	0	0	0	1	0	0	0
										P_2	P_1	P_0	0	1	Q_2	Q_1	Q_0
	AX, saddrp	AX, CY ← AX + (saddrp)	2	4	X	X	X	V	X	0	0	0	1	1	1	0	1
										Saddr-offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
16-Bit Arithmetic (cont)																	
ADDW (cont)	AX, sfrp	AX, CY ← AX + sfrp	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	0	0	1	1	1	0	1
										Sfr-offset							
	saddrp, saddrp	(saddrp), CY ← (saddrp) + (saddrp)	3	5	X	X	X	V	X	0	0	1	1	1	1	0	1
										Saddr-offset							
										Saddr-offset							
SUBW	AX, #word	AX, CY ← AX – word	3	3	X	X	X	V	X	0	0	1	0	1	1	1	0
										Low Byte							
										High Byte							
	saddrp, #word	(saddrp), CY ← (saddrp) – word	4	5	X	X	X	V	X	0	0	0	0	1	1	1	0
										Saddr-offset							
										Low Byte							
										High Byte							
	sfrp, #word	sfrp, CY ← sfrp – word	5	10	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	0	0	0	1	1	1	0
										Sfr-offset							
										Low Byte							
										High Byte							
	rp, rp1	rp, CY ← rp – rp1	2	3	X	X	X	V	X	1	0	0	0	1	0	1	0
										P_2	P_1	P_0	0	1	Q_2	Q_1	Q_0
	AX, saddrp	AX, CY ← AX – (saddrp)	2	4	X	X	X	V	X	0	0	0	1	1	1	1	0
										Saddr-offset							
	AX, sfrp	AX, CY ← AX – sfrp	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	0	0	1	1	1	1	0
										Sfr-offset							
	saddrp, saddrp	(saddrp), CY ← (saddrp) – (saddrp)	3	5	X	X	X	V	X	0	0	1	1	1	1	1	0
										Saddr-offset							
										Saddr-offset							
CMPW	AX, #word	AX – word	3	3	X	X	X	V	X	0	0	1	0	1	1	1	1
										Low Byte							
										High Byte							
	saddrp, #word	(saddrp) – word	4	5	X	X	X	V	X	0	0	0	0	1	1	1	1
										Saddr-offset							
										Low Byte							
										High Byte							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
16-Bit Arithmetic (cont)																	
CMPW (cont)	sfrp, #word	sfrp – word	5	10	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	0	0	0	1	1	1	1
										Sfr-offset							
										Low Byte							
										High Byte							
	rp, rp1	rp – rp1	2	3	X	X	X	V	X	1	0	0	0	1	1	1	1
										P_2	P_1	P_0	0	1	Q_2	Q_1	Q_0
	AX, saddrp	AX – (saddrp)	2	4	X	X	X	V	X	0	0	0	1	1	1	1	1
										Saddr-offset							
	AX, sfrp	AX – sfrp	3	9	X	X	X	V	X	0	0	0	0	0	0	0	1
										0	0	0	1	1	1	1	1
										Sfr-offset							
	saddrp, saddrp	(saddrp) – (saddrp)	3	5	X	X	X	V	X	0	0	1	1	1	1	1	1
										Saddr-offset							
										Saddr-offset							
Multiplication/Division																	
MULU	r1	AX ← A × r1	2	14						0	0	0	0	0	1	0	1
										0	0	0	0	1	R_2	R_1	R_0
DIVUW	r1	AX (Quotient), r1 (Remainder) ← AX ÷ r1	2	23						0	0	0	0	0	1	0	1
										0	0	0	1	1	R_2	R_1	R_0
MULUW	rp1	AX (High Order 16 Bits), rp1 (Low Order 16 Bits), ← AX × rp1	2	22						0	0	0	0	0	1	0	1
										0	0	1	0	1	Q_2	Q_1	Q_0
DIVUX	rp1	AXDE (Quotient), rp1 (Remainder) ← AXDE ÷ rp1	2	43						0	0	0	0	0	1	0	1
										1	1	1	0	1	Q_2	Q_1	Q_0
MULW*	rp1	AX (High Order 16 Bits), rp1 (Low Order 16 Bits), ← AX × rp1	2	24-28						0	0	0	0	0	1	0	1
										0	0	1	1	1	Q_2	Q_1	Q_0
Increment/Decrement																	
INC	r1	r1 ← r1 + 1	1	2	X	X	X	V		1	1	0	0	0	R_2	R_1	R_0
	saddr	(saddr) ← (saddr) + 1	2	3	X	X	X	V		0	0	1	0	0	1	1	0
										Saddr-offset							
DEC	r1	r1 ← r1 – 1	1	2	X	X	X	V		1	1	0	0	1	R_2	R_1	R_0
	saddr	(saddr) ← (saddr) – 1	2	3	X	X	X	V		0	0	1	0	0	1	1	1
										Saddr-offset							
INCW	rp2	rp2 ← rp2 + 1	1	2						0	1	0	0	0	1	S_1	S_0
	saddrp	(saddrp) ← (saddrp) + 1	3	4						0	0	0	0	0	1	1	1
										1	1	1	0	1	0	0	0
										Saddr-offset							

* 16-bit signed multiply instruction.

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
Increment/Decrement (cont)																	
DECW	rp2	rp2 ← rp2 − 1	1	2						0	1	0	0	1	1	S_1	S_0
	saddrp	(saddrp) ← (saddrp) − 1	3	4						0	0	0	0	0	1	1	1
										1	1	1	0	1	0	0	1
										Saddr-offset							
Shift/Rotate																	
ROR	r1, n	$(CY, r1_7 \leftarrow r1_0, r1_{m-1} \leftarrow r1_m) \times n$	2	6+n				P	X	0	0	1	1	0	0	0	0
										0	1	N_2	N_1	N_0	R_2	R_1	R_0
ROL	r1, n	$(CY, r1_0 \leftarrow r1_7, r1_{m+1} \leftarrow r1_m) \times n$	2	6+n				P	X	0	0	1	1	0	0	0	1
										0	1	N_2	N_1	N_0	R_2	R_1	R_0
RORC	r1, n	$(CY \leftarrow r1_0, r1_7 \leftarrow CY, r1_{m-1} \leftarrow r1_m) \times n$	2	6+n				P	X	0	0	1	1	0	0	0	0
										0	0	N_2	N_1	N_0	R_2	R_1	R_0
ROLC	r1, n	$(CY \leftarrow r1_7, r1_0 \leftarrow CY, r1_{m+1} \leftarrow r1_m) \times n$	2	6+n				P	X	0	0	1	1	0	0	0	1
										0	0	N_2	N_1	N_0	R_2	R_1	R_0
SHR	r1, n	$(CY \leftarrow r1_0, r1_7 \leftarrow 0, r1_{m-1} \leftarrow r1_m) \times n$	2	6+n	X	X	0	P	X	0	0	1	1	0	0	0	0
										1	0	N_2	N_1	N_0	R_2	R_1	R_0
SHL	r1, n	$(CY \leftarrow r1_7, r1_0 \leftarrow 0, r1_{m+1} \leftarrow r1_m) \times n$	2	6+n	X	X	0	P	X	0	0	1	1	0	0	0	1
										1	0	N_2	N_1	N_0	R_2	R_1	R_0
SHRW	rp1, n	$(CY \leftarrow rp1_0, rp1_{15} \leftarrow 0, rp1_{m-1} \leftarrow rp1_m) \times n$	2	6+n	X	X	0	P	X	0	0	1	1	0	0	0	0
										1	1	N_2	N_1	N_0	Q_2	Q_1	Q_0
SHLW	rp1, n	$(CY \leftarrow rp1_{15}, rp1_0 \leftarrow 0, rp1_{m+1} \leftarrow rp1_m) \times n$	2	6+n	X	X	0	P	X	0	0	1	1	0	0	0	1
										1	1	N_2	N_1	N_0	Q_2	Q_1	Q_0
ROR4	[rp1]	$A_{3-0} \leftarrow (rp1)_{3-0}, (rp1)_{7-4} \leftarrow A_{3-0}, (rp1)_{3-0} \leftarrow (rp1)_{7-4}$	2	8						0	0	0	0	0	1	0	1
										1	0	0	0	1	Q_2	Q_1	Q_0
ROL4	[rp1]	$A_{3-0} \leftarrow (rp1)_{7-4}, (rp1)_{3-0} \leftarrow A_{3-0}, (rp1)_{7-4} \leftarrow (rp1)_{3-0}$	2	8						0	0	0	0	0	1	0	1
										1	0	0	1	1	Q_2	Q_1	Q_0
BCD Adjustment																	
ADJBA		Decimal Adjust Accumulator after add	2	5	X	X	X	P	X	0	0	0	0	0	1	0	1
										1	1	1	1	1	1	1	0
ADJBS		Decimal Adjust Accumulator after subtract	2	5	X	X	X	P	X	0	0	0	0	0	1	0	1
										1	1	1	1	1	1	1	1
Data Expansion																	
CVTBW		$X \leftarrow A, A_{6-0} \leftarrow A_7$	1	3						0	0	0	0	0	1	0	0
Bit Manipulation																	
MOV1	CY, saddr.bit	CY ← (saddr.bit)	3	6					X	0	0	0	0	1	0	0	0
										0	0	0	0	0	B_2	B_1	B_0
										Saddr-offset							
	CY, sfr.bit	CY ← sfr.bit	3	9					X	0	0	0	0	1	0	0	0
										0	0	0	0	1	B_2	B_1	B_0
										Sfr-offset							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	**Operand**	**Operation**	**Bytes**	**States**	**S**	**Z**	**AC**	**P/V**	**CY**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Manipulation (cont)																	
MOV1 (cont)	CY, A.bit	CY ← A.bit	2	6					X	0	0	0	0	0	0	1	1
										0	0	0	0	1	B_2	B_1	B_0
	CY, X.bit	CY ← X.bit	2	6					X	0	0	0	0	0	0	1	1
										0	0	0	0	0	B_2	B_1	B_0
	CY, PSWH.bit	CY ← PSW$_H$.bit	2	6					X	0	0	0	0	0	0	1	0
										0	0	0	0	1	B_2	B_1	B_0
	CY, PSWL.bit	CY ← PSW$_L$.bit	2	6					X	0	0	0	0	0	0	1	0
										0	0	0	0	0	B_2	B_1	B_0
	saddr.bit, CY	(saddr.bit) ← CY	3	5						0	0	0	0	1	0	0	0
										0	0	0	1	0	B_2	B_1	B_0
										Saddr-offset							
	sfr.bit, CY	sfr.bit ← CY	3	8						0	0	0	0	1	0	0	0
										0	0	0	1	1	B_2	B_1	B_0
										Sfr-offset							
	A.bit, CY	A.bit ← CY	2	7						0	0	0	0	0	0	1	1
										0	0	0	1	1	B_2	B_1	B_0
	X.bit, CY	X.bit ← CY	2	7						0	0	0	0	0	0	1	1
										0	0	0	1	0	B_2	B_1	B_0
	PSWH.bit, CY	PSW$_H$.bit ← CY	2	8						0	0	0	0	0	0	1	0
										0	0	0	1	1	B_2	B_1	B_0
	PSWL.bit, CY	PSW$_L$.bit ← CY	2	8	X	X	X	X		0	0	0	0	0	0	1	0
										0	0	0	1	0	B_2	B_1	B_0
AND1	CY, saddr.bit	CY ← CY AND (saddr.bit)	3	6					X	0	0	0	0	1	0	0	0
										0	0	1	0	0	B_2	B_1	B_0
										Saddr-offset							
	CY, /saddr.bit	CY ← CY AND ($\overline{\text{saddr.bit}}$)	3	6					X	0	0	0	0	1	0	0	0
										0	0	1	1	0	B_2	B_1	B_0
										Saddr-offset							
	CY, sfr.bit	CY ← CY AND sfr.bit	3	9					X	0	0	0	0	1	0	0	0
										0	0	1	0	1	B_2	B_1	B_0
										Sfr-offset							
	CY, /sfr.bit	CY ← CY AND $\overline{\text{sfr.bit}}$	3	9					X	0	0	0	0	1	0	0	0
										0	0	1	1	1	B_2	B_1	B_0
										Sfr-offset							
	CY, A.bit	CY ← CY AND A.bit	2	6					X	0	0	0	0	0	0	1	1
										0	0	1	0	1	B_2	B_1	B_0
	CY, /A.bit	CY ← CY AND $\overline{\text{A.bit}}$	2	6					X	0	0	0	0	0	0	1	1
										0	0	1	1	1	B_2	B_1	B_0
	CY, X.bit	CY ← CY AND X.bit	2	6					X	0	0	0	0	0	0	1	1
										0	0	1	0	0	B_2	B_1	B_0

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	**Operand**	**Operation**	**Bytes**	**States**	**S**	**Z**	**AC**	**P/V**	**CY**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Manipulation (cont)																	
AND1	CY, /X.bit	CY ← CY AND $\overline{\text{X.bit}}$	2	6					X	0	0	0	0	0	0	1	1
(cont)										0	0	1	1	0	B_2	B_1	B_0
	CY, PSWH.bit	CY ← CY AND PSW_H.bit	2	6					X	0	0	0	0	0	0	1	0
										0	0	1	0	1	B_2	B_1	B_0
	CY, /PSWH.bit	CY ← CY AND $\overline{\text{PSW}_H\text{.bit}}$	2	6					X	0	0	0	0	0	0	1	0
										0	0	1	1	1	B_2	B_1	B_0
	CY, PSWL.bit	CY ← CY AND PSW_L.bit	2	6					X	0	0	0	0	0	0	1	0
										0	0	1	0	0	B_2	B_1	B_0
	CY, /PSWL.bit	CY ← CY AND $\overline{\text{PSW}_L\text{.bit}}$	2	6					X	0	0	0	0	0	0	1	0
										0	0	1	1	0	B_2	B_1	B_0
OR1	CY, saddr.bit	CY ← CY OR (saddr.bit)	3	6					X	0	0	0	0	1	0	0	0
										0	1	0	0	0	B_2	B_1	B_0
										Saddr-offset							
	CY, /saddr.bit	CY ← CY OR $\overline{\text{(saddr.bit)}}$	3	6					X	0	0	0	0	1	0	0	0
										0	1	0	1	0	B_2	B_1	B_0
										Saddr-offset							
	CY, sfr.bit	CY ← CY OR sfr.bit	3	9					X	0	0	0	0	1	0	0	0
										0	1	0	0	1	B_2	B_1	B_0
										Sfr-offset							
	CY, /sfr.bit	CY ← CY OR $\overline{\text{sfr.bit}}$	3	9					X	0	0	0	0	1	0	0	0
										0	1	0	1	1	B_2	B_1	B_0
										Sfr-offset							
	CY, A.bit	CY ← CY OR A.bit	2	6					X	0	0	0	0	0	0	1	1
										0	1	0	0	1	B_2	B_1	B_0
	CY, /A.bit	CY ← CY OR $\overline{\text{A.bit}}$	2	6					X	0	0	0	0	0	0	1	1
										0	1	0	1	1	B_2	B_1	B_0
	CY, X.bit	CY ← CY OR X.bit	2	6					X	0	0	0	0	0	0	1	1
										0	1	0	0	0	B_2	B_1	B_0
	CY, /X.bit	CY ← CY OR $\overline{\text{X.bit}}$	2	6					X	0	0	0	0	0	0	1	1
										0	1	0	1	0	B_2	B_1	B_0
	CY, PSWH.bit	CY ← CY OR PSW_H.bit	2	6					X	0	0	0	0	0	0	1	0
										0	1	0	0	1	B_2	B_1	B_0
	CY, /PSWH.bit	CY ← CY OR $\overline{\text{PSW}_H\text{.bit}}$	2	6					X	0	0	0	0	0	0	1	0
										0	1	0	1	1	B_2	B_1	B_0
	CY, PSWL.bit	CY ← CY OR PSW_L.bit	2	6					X	0	0	0	0	0	0	1	0
										0	1	0	0	0	B_2	B_1	B_0
	CY, /PSWL.bit	CY ← CY OR $\overline{\text{PSW}_L\text{.bit}}$	2	6					X	0	0	0	0	0	0	1	0
										0	1	0	1	0	B_2	B_1	B_0

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
Bit Manipulation (cont)																	
XOR1	CY, saddr.bit	CY ← CY XOR (saddr.bit)	3	6					X	0	0	0	0	1	0	0	0
										0	1	1	0	0	B_2	B_1	B_0
										Saddr-offset							
	CY, sfr.bit	CY ← CY XOR sfr.bit	3	9					X	0	0	0	0	1	0	0	0
										0	1	1	0	1	B_2	B_1	B_0
										Sfr-offset							
	CY, A.bit	CY ← CY XOR A.bit	2	6					X	0	0	0	0	0	0	1	1
										0	1	1	0	1	B_2	B_1	B_0
	CY, X.bit	CY ← CY XOR X.bit	2	6					X	0	0	0	0	0	0	1	1
										0	1	1	0	0	B_2	B_1	B_0
	CY, PSWH.bit	CY ← CY XOR PSW_H.bit	2	6					X	0	0	0	0	0	0	1	0
										0	1	1	0	1	B_2	B_1	B_0
	CY, PSWL.bit	CY ← CY XOR PSW_L.bit	2	6					X	0	0	0	0	0	0	1	0
										0	1	1	0	0	B_2	B_1	B_0
SET1	saddr.bit	(saddr.bit) ← 1	2	4						1	0	1	1	0	B_2	B_1	B_0
										Saddr-offset							
	sfr.bit	sfr.bit ← 1	3	11						0	0	0	0	1	0	0	0
										1	0	0	0	1	B_2	B_1	B_0
										Sfr-offset							
	A.bit	A.bit ← 1	2	6						0	0	0	0	0	0	1	1
										1	0	0	0	1	B_2	B_1	B_0
	X.bit	X.bit ← 1	2	6						0	0	0	0	0	0	1	1
										1	0	0	0	0	B_2	B_1	B_0
	PSWH.bit	PSW_H.bit ← 1	2	7						0	0	0	0	0	0	1	0
										1	0	0	0	1	B_2	B_1	B_0
	PSWL.bit	PSW_L.bit ← 1	2	7	X	X	X	X	X	0	0	0	0	0	0	1	0
										1	0	0	0	0	B_2	B_1	B_0
CLR1	saddr.bit	(saddr.bit) ← 0	2	4						1	0	1	0	0	B_2	B_1	B_0
										Saddr-offset							
	sfr.bit	sfr.bit ← 0	3	11						0	0	0	0	1	0	0	0
										1	0	0	1	1	B_2	B_1	B_0
										Sfr-offset							
	A.bit	A.bit ← 0	2	6						0	0	0	0	0	0	1	1
										1	0	0	1	1	B_2	B_1	B_0
	X.bit	X.bit ← 0	2	6						0	0	0	0	0	0	1	1
										1	0	0	1	0	B_2	B_1	B_0
	PSWH.bit	PSW_H.bit ← 0	2	7						0	0	0	0	0	0	1	0
										1	0	0	1	1	B_2	B_1	B_0
	PSWL.bit	PSW_L.bit ← 0	2	7	X	X	X	X	X	0	0	0	0	0	0	1	0
										1	0	0	1	0	B_2	B_1	B_0

Instruction Set (cont)

Mnemonic	Operand	Operation	Bytes	States	Flags S	Z	AC	P/V	CY	Operation Code 7	6	5	4	3	2	1	0
Bit Manipulation (cont)																	
NOT1	saddr.bit	(saddr.bit) ← $\overline{\text{(saddr.bit)}}$	3	5						0	0	0	0	1	0	0	0
										0	1	1	1	0	B_2	B_1	B_0
										Saddr-offset							
	sfr.bit	sfr.bit ← $\overline{\text{sfr.bit}}$	3	11						0	0	0	0	1	0	0	0
										0	1	1	1	1	B_2	B_1	B_0
										Sfr-offset							
	A.bit	A.bit ← $\overline{\text{A.bit}}$	2	6						0	0	0	0	0	0	1	1
										0	1	1	1	1	B_2	B_1	B_0
	X.bit	X.bit ← $\overline{\text{X.bit}}$	2	6						0	0	0	0	0	0	1	1
										0	1	1	1	0	B_2	B_1	B_0
	PSWH.bit	PSW_H.bit ← $\overline{PSW_H\text{.bit}}$	2	7						0	0	0	0	0	0	1	0
										0	1	1	1	1	B_2	B_1	B_0
	PSWL.bit	PSW_L.bit ← $\overline{PSW_L\text{.bit}}$	2	7	X	X	X	X	X	0	0	0	0	0	0	1	0
										0	1	1	1	0	B_2	B_1	B_0
SET1	CY	CY ← 1	1	2					1	0	1	0	0	0	0	0	1
CLR1	CY	CY ← 0	1	2					0	0	1	0	0	0	0	0	0
NOT1	CY	CY ← $\overline{\text{CY}}$	1	2					X	0	1	0	0	0	0	1	0
Subroutine Linkage																	
CALL	!addr16	(SP−1) ← (PC + 3)$_H$, (SP−2) ← (PC + 3)$_L$, PC ← addr16, SP ← SP−2	3	6						0	0	1	0	1	0	0	0
										Low Addr							
										High Addr							
	rp1	(SP−1) ← (PC + 2)$_H$, (SP−2) ← (PC + 2)$_L$, PC_H ← rp1$_H$, PC_L ← rp1$_L$, SP ← SP−2	2	7						0	0	0	0	0	1	0	1
										0	1	0	1	1	Q_2	Q_1	Q_0
	[rp1]	(SP−1) ← (PC + 2)$_H$, (SP−2) ← (PC + 2)$_L$, PC_H ← (rp1 + 1), PC_L ← (rp1), SP ← SP−2	2	10						0	0	0	0	0	1	0	1
										0	1	1	1	1	Q_2	Q_1	Q_0
CALLF	!addr11	(SP−1) ← (PC + 2)$_H$, (SP−2) ← (PC + 2)$_L$, $PC_{15\text{-}11}$ ← 00001, $PC_{10\text{-}0}$ ← addr11, SP ← SP−2	2	6						1	0	0	1	0	f_{10}	f_9	f_8
										f_7	f_6	f_5	f_4	f_3	f_2	f_1	f_0
CALLT	[addr5]	(SP−1) ← (PC + 1)$_H$, (SP−2) ← (PC + 1)$_L$, PC_H ← (TPFx8000H + 2 × addr5 + 41H), PC_L ← (TPFx8000H + 2 × addr5 + 40H), SP ← SP−2	1	9						1	1	1	t_4	t_3	t_2	t_1	t_0
BRK		(SP−1) ← PSW_H, (SP−2) ← PSW_L, (SP−3) ← (PC + 1)$_H$, (SP−4) ← (PC + 1)$_L$, PC_L ← (003EH), PC_H ← (003FH), SP ← SP−4, IE ← 0	1	12						0	1	0	1	1	1	1	0

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
Subroutine Linkage (cont)																	
RET		$PC_L \leftarrow (SP), PC_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$	1	6						0	1	0	1	0	1	1	0
RETB		$PC_L \leftarrow (SP), PC_H \leftarrow (SP+1)$, $PSW_L \leftarrow (SP+2), PSW_H \leftarrow (SP+3)$, $SP \leftarrow SP+4$	1	10	R	R	R	R	R	0	1	0	1	1	1	1	1
RETI		$PC_L \leftarrow (SP), PC_H \leftarrow (SP+1)$, $PSW_L \leftarrow (SP+2), PSW_H \leftarrow (SP+3)$, $SP \leftarrow SP+4$	1	10	R	R	R	R	R	0	1	0	1	0	1	1	1
Stack Manipulation																	
PUSH	sfrp	$(SP-1) \leftarrow sfr_H, (SP-2) \leftarrow sfr_L$, $SP \leftarrow SP-2$	3	9						0	0	0	0	0	1	1	1
										1	1	0	1	1	0	0	1
										Sfr-offset							
	post	$\left\{ (SP-1) \leftarrow rpp_H, (SP-2) \leftarrow rpp_L, SP \leftarrow SP-2 \right\} \times n$*	2	9-51**						0	0	1	1	0	1	0	1
										Post Byte							
	PSW	$(PS-1) \leftarrow PSW_H, (SP-2) \leftarrow PSW_L$, $SP \leftarrow SP-2$	1	3						0	1	0	0	1	0	0	1
PUSHU	post	$\left\{ (UP-1) \leftarrow rpp_H, (UP-2) \leftarrow rpp_L, UP \leftarrow UP-2 \right\} \times n$*	2	10-52**						0	0	1	1	0	1	1	1
										Post Byte							
POP	sfrp	$sfr_L \leftarrow (SP), sfr_H \leftarrow (SP+1)$, $SP \leftarrow SP+2$	3	10						0	0	0	0	0	1	1	1
										1	1	0	1	1	0	0	0
										Sfr-offset							
	post	$\left\{ rpp_L \leftarrow (SP), rpp_H \leftarrow (SP+1)\ SP \leftarrow SP+2 \right\} \times n$*	2	13-62**						0	0	1	1	0	1	0	0
										Post Byte							
	PSW	$PSW_L \leftarrow (SP), PSW_H \leftarrow (SP+1)$ $SP \leftarrow SP+2$	1	5	R	R	R	R	R	0	1	0	0	1	0	0	0
POPU	post	$\left\{ rpp_L \leftarrow (UP), rpp_H \leftarrow (UP+1), UP \leftarrow UP+2 \right\} \times n$*	2	15-64**						0	0	1	1	0	1	1	0
										Post Byte							
MOVW	SP, #word	$SP \leftarrow word$	4	7						0	0	0	0	1	0	1	1
										1	1	1	1	1	1	0	0
										Low Byte							
										High Byte							
	SP, AX	$SP \leftarrow AX$	2	6						0	0	0	1	0	0	1	1
										1	1	1	1	1	1	0	0
	AX, SP	$AX \leftarrow SP$	2	6						0	0	0	1	0	0	0	1
										1	1	1	1	1	1	0	0
INCW	SP	$SP \leftarrow SP+1$	2	3						0	0	0	0	0	1	0	1
										1	1	0	0	1	0	0	0
DECW	SP	$SP \leftarrow SP-1$	2	3						0	0	0	0	0	1	0	1
										1	1	0	0	1	0	0	1

* rpp refers to register pairs specified in post byte. n is the number of register pairs specified in post byte.

** The details of the timing are described under "Timing of the PUSH and POP Instructions."

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
Pin Level Test																	
CHKL	sfr	(Pin level) XOR (internal signal level)	3	12	X	X		P		0	0	0	0	0	1	1	1
										1	1	0	0	1	0	0	0
										Sfr-offset							
CHKLA	sfr	A ← (Pin level) XOR (internal signal level)	3	12	X	X		P		0	0	0	0	0	1	1	1
										1	1	0	0	1	0	0	1
										Sfr-offset							
Unconditional Branch																	
BR	!addr16	PC ← addr16	3	4						0	0	1	0	1	1	0	0
										Low Addr							
										High Addr							
	rp1	$PC_H \leftarrow rp1_H, PC_L \leftarrow rp1_L$	2	4						0	0	0	0	0	1	0	1
										0	1	0	0	1	Q_2	Q_1	Q_0
	[rp1]	$PC_H \leftarrow (rp1 + 1), PC_L \leftarrow (rp1)$	2	8						0	0	0	0	0	1	0	1
										0	1	1	0	1	Q_2	Q_1	Q_0
	$addr16	PC ← addr16	2	4						0	0	0	1	0	1	0	0
										jdisp							
Conditional Branch																	
BC,	$addr16	PC ← addr16 if CY = 1	2	4						1	0	0	0	0	0	1	1
BL										jdisp							
BNC,	$addr16	PC ← addr16 if CY = 0	2	4						1	0	0	0	0	0	1	0
BNL										jdisp							
BZ,	$addr16	PC ← addr16 if Z = 1	2	4						1	0	0	0	0	0	0	1
BE										jdisp							
BNZ,	$addr16	PC ← addr16 if Z = 0	2	4						1	0	0	0	0	0	0	0
BNE										jdisp							
BV,	$addr16	PC ← addr16 if P/V = 1	2	4						1	0	0	0	0	1	0	1
BPE										jdisp							
BNV,	$addr16	PC ← addr16 if P/V = 0	2	4						1	0	0	0	0	1	0	0
BPO										jdisp							
BN	$addr16	PC ← addr16 if S = 1	2	4						1	0	0	0	0	1	1	1
										jdisp							
BP	$addr16	PC ← addr16 if S = 0	2	4						1	0	0	0	0	1	1	0
										jdisp							
BGT	$addr16	PC ← addr16 if (P/V XOR S) OR Z = 0	3	5						0	0	0	0	0	1	1	1
										1	1	1	1	1	0	1	1
										jdisp							
BGE	$addr16	PC ← addr16 if P/V XOR S = 0	3	5						0	0	0	0	0	1	1	1
										1	1	1	1	1	0	0	1
										jdisp							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	**Operand**	**Operation**	**Bytes**	**States**	**S**	**Z**	**AC**	**P/V**	**CY**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Conditional Branch (cont)																	
BLT	\$addr16	PC ← addr16 if P/V XOR S = 1	3	5						0	0	0	0	0	1	1	1
										1	1	1	1	1	0	0	0
										jdisp							
BLE	\$addr16	PC ← addr16 if (P/V XOR S) OR Z = 1	3	5						0	0	0	0	0	1	1	1
										1	1	1	1	1	0	1	0
										jdisp							
BH	\$addr16	PC ← addr16 if Z OR CY = 0	3	5						0	0	0	0	0	1	1	1
										1	1	1	1	1	1	0	1
										jdisp							
BNH	\$addr16	PC ← addr16 if Z OR CY = 1	3	5						0	0	0	0	0	1	1	1
										1	1	1	1	1	1	0	0
										jdisp							
BT	saddr.bit, \$addr16	PC ← addr16 if (saddr.bit) = 1	3	7						0	1	1	1	0	B_2	B_1	B_0
										Saddr-offset							
										jdisp							
	sfr.bit, \$addr16	PC ← addr16 if sfr.bit = 1	4	8						0	0	0	0	1	0	0	0
										1	0	1	1	1	B_2	B_1	B_0
										Sfr-offset							
										jdisp							
	A.bit, \$addr16	PC ← addr16 if A.bit = 1	3	8						0	0	0	0	0	0	1	1
										1	0	1	1	1	B_2	B_1	B_0
										jdisp							
	X.bit, \$addr16	PC ← addr16 if X.bit = 1	3	8						0	0	0	0	0	0	1	1
										1	0	1	1	0	B_2	B_1	B_0
										jdisp							
	PSWH.bit, \$addr16	PC ← addr16 if PSW_H.bit = 1	3	8						0	0	0	0	0	0	1	0
										1	0	1	1	1	B_2	B_1	B_0
										jdisp							
	PSWL.bit, \$addr16	PC ← addr16 if PSW_L.bit = 1	3	8						0	0	0	0	0	0	1	0
										1	0	1	1	0	B_2	B_1	B_0
										jdisp							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
Conditional Branch (cont)																	
BF	saddr.bit, $addr16	PC ← addr16 if (saddr.bit) = 0	4	7						0	0	0	0	1	0	0	0
										1	0	1	0	0	B$_2$	B$_1$	B$_0$
										Saddr-offset							
										jdisp							
	sfr.bit, $addr16	PC ← addr16 if sfr.bit = 0	4	8						0	0	0	0	1	0	0	0
										1	0	1	0	1	B$_2$	B$_1$	B$_0$
										Sfr-offset							
										jdisp							
	A.bit, $addr16	PC ← addr16 if A.bit = 0	3	8						0	0	0	0	0	0	1	1
										1	0	1	0	1	B$_2$	B$_1$	B$_0$
										jdisp							
	X.bit, $addr16	PC ← addr16 if X.bit = 0	3	8						0	0	0	0	0	0	1	1
										1	0	1	0	0	B$_2$	B$_1$	B$_0$
										jdisp							
	PSWH.bit, $addr16	PC ← addr16 if PSW$_H$.bit = 0	3	8						0	0	0	0	0	0	1	0
										1	0	1	0	1	B$_2$	B$_1$	B$_0$
										jdisp							
	PSWL.bit, $addr16	PC ← addr16 if PSW$_L$.bit = 0	3	8						0	0	0	0	0	0	1	0
										1	0	1	0	0	B$_2$	B$_1$	B$_0$
										jdisp							
BTCLR	saddr.bit, $addr16	PC ← addr16 if (saddr.bit) = 1 then reset (saddr.bit)	4	8/10						0	0	0	0	1	0	0	0
										1	1	0	1	0	B$_2$	B$_1$	B$_0$
										Saddr-offset							
										jdisp							
	sfr.bit, $addr16	PC ← addr16 if sfr.bit = 1 then reset sfr.bit	4	8/10						0	0	0	0	1	0	0	0
										1	1	0	1	1	B$_2$	B$_1$	B$_0$
										Sfr-offset							
										jdisp							
	A.bit, $addr16	PC ← addr16 if A.bit = 1 then reset A.bit	3	8/10						0	0	0	0	0	0	1	1
										1	1	0	1	1	B$_2$	B$_1$	B$_0$
										jdisp							
	X.bit, $addr16	PC ← addr16 if X.bit = 1 then reset X.bit	3	8/10						0	0	0	0	0	0	1	1
										1	1	0	1	0	B$_2$	B$_1$	B$_0$
										jdisp							
	PSWH.bit, $addr16	PC ← addr16 if PSW$_H$.bit = 1 then reset PSW$_H$.bit	3	8/10						0	0	0	0	0	0	1	0
										1	1	0	1	1	B$_2$	B$_1$	B$_0$
										jdisp							

Instruction Set (cont)

Conditional Branch (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
BTCLR	PSWL.bit,	PC ← addr16 if PSW_L.bit = 1	3	8/10	X	X	X	X	X	0	0	0	0	0	0	1	0
(cont)	$addr16	then reset PSW_L.bit								1	1	0	1	0	B_2	B_1	B_0
										jdisp							
BFSET	saddr.bit,	PC ← addr16 if (saddr.bit) = 0	4	8/10						0	0	0	0	1	0	0	0
	$addr16	then set (saddr.bit)								1	1	0	0	0	B_2	B_1	B_0
										Saddr-offset							
										jdisp							
	sfr.bit, $addr16	PC ← addr16 if sfr.bit = 0	4	8/10						0	0	0	0	1	0	0	0
		then set sfr.bit								1	1	0	0	1	B_2	B_1	B_0
										Sfr-offset							
										jdisp							
	A.bit, $addr16	PC ← addr16 if A.bit = 0	3	8/10						0	0	0	0	0	0	1	1
		then set A.bit								1	1	0	0	1	B_2	B_1	B_0
										jdisp							
	X.bit, $addr16	PC ← addr16 if X.bit = 0	3	8/10						0	0	0	0	0	0	1	1
		then set X.bit								1	1	0	0	0	B_2	B_1	B_0
										jdisp							
	PSWH.bit,	PC ← addr16 if PSW_H.bit = 0	3	8/10						0	0	0	0	0	0	1	0
	$addr16	then set PSW_H.bit								1	1	0	0	1	B_2	B_1	B_0
										jdisp							
	PSWL.bit,	PC ← addr16 if PSW_L.bit = 0	3	8/10	X	X	X	X	X	0	0	0	0	0	0	1	0
	$addr16	then set PSW_L.bit								1	1	0	0	0	B_2	B_1	B_0
										jdisp							
DBNZ	r2, $addr16	r2 ← r2 − 1,	2	5/6						0	0	1	1	0	0	1	C_0
		then PC ← addr16 if r2 ≠ 0								jdisp							
	saddr, $addr16	(saddr) ← (saddr) − 1,	3	6/7						0	0	1	1	1	0	1	1
		then PC ← addr16 if saddr ≠ 0								Saddr-offset							
										jdisp							

Context Switching

Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
BRKCS	RBn	PC_H ↔ R5, PC_L ↔ R4, R7 ← PSW_H, R6 ← PSW_L,	2	7						0	0	0	0	0	1	0	1
		RBS_{2-0} ← n, RSS ← 0, IE ← 0								1	1	0	1	1	N_2	N_1	N_0
RETCS	!addr16	PC_H ← R5, PC_L ← R4,	3	5	R	R	R	R	R	0	0	1	0	1	0	0	1
		R5, R4 ← addr16, PSW_H ← R7								Low Addr							
		PSW_L ← R6 (priority change)								High Addr							
RETCSB	!addr16	PC_H ← R5, PC_L ← R4,	4	5	R	R	R	R	R	0	0	0	0	1	0	0	1
		R5, R4 ← addr16, PSW_H ← R7								1	1	1	0	0	0	0	0
		PSW_L ← R6 (no priority change)								Low Addr							
										High Addr							

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	Operand	Operation	Bytes	States	S	Z	AC	P/V	CY	7	6	5	4	3	2	1	0
String Manipulation																	
MOVM	[DE+], A	(DE+) ← A, C ← C−1	2	3+6n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	0	0	0	0	0	0
	[DE−], A	(DE−) ← A, C ← C−1	2	3+6n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	0	1	0	0	0	0
MOVBK	[DE+], [HL+]	(DE+) ← (HL+), C ← C−1	2	3+9n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	1	0	0	0	0	0
	[DE−], [HL−]	(DE−) ← (HL−), C ← C−1	2	3+9n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	1	1	0	0	0	0
XCHM	[DE+], A	(DE+) ↔ A, C ← C−1	2	3+10n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	0	0	0	0	0	1
	[DE−], A	(DE−) ↔ A, C ← C−1	2	3+10n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	0	1	0	0	0	1
XCHBK	[DE+], [HL+]	(DE+) ↔ (HL+), C ← C−1	2	3+16n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	1	0	0	0	0	1
	[DE−], [HL−]	(DE−) ↔ (HL−), C ← C−1	2	3+16n						0	0	0	1	0	1	0	1
		End if C = 0								0	0	1	1	0	0	0	1
CMPME	[DE+], A	(DE+) − A, C ← C−1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 0								0	0	0	0	0	1	0	0
	[DE−], A	(DE−) − A, C ← C−1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 0								0	0	0	1	0	1	0	0
CMPBKE	[DE+], [HL+]	(DE+) − (HL+), C ← C−1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 0								0	0	1	0	0	1	0	0
	[DE−], [HL−]	(DE−) − (HL−), C ← C−1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 0								0	0	1	1	0	1	0	0
CMPMNE	[DE+], A	(DE+) − A, C ← C−1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 1								0	0	0	0	0	1	0	1
	[DE−], A	(DE−) − A, C ← C−1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 1								0	0	0	1	0	1	0	1
CMPBKNE	[DE+], [HL+]	(DE+) − (HL+), C ← C−1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 1								0	0	1	0	0	1	0	1
	[DE−], [HL−]	(DE−) − (HL−), C ← C−1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or Z = 1								0	0	1	1	0	1	0	1
CMPMC	[DE+], A	(DE+) − A, C ← C−1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or CY = 0								0	0	0	0	0	1	1	1
	[DE−], A	(DE−) − A, C ← C−1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or CY = 0								0	0	0	1	0	1	1	1
CMPBKC	[DE+], [HL+]	(DE+) − (HL+), C ← C−1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or CY = 0								0	0	1	0	0	1	1	1
	[DE−], [HL−]	(DE−) − (HL−), C ← C−1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
		End if C = 0 or CY = 0								0	0	1	1	0	1	1	1

Instruction Set (cont)

					Flags					Operation Code							
Mnemonic	**Operand**	**Operation**	**Bytes**	**States**	**S**	**Z**	**AC**	**P/V**	**CY**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
String Manipulation (cont)																	
CMPMNC	[DE+], A	(DE+) − A, C ← C − 1 End if C = 0 or CY = 1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
										0	0	0	0	0	1	1	0
	[DE−], A	(DE−) − A, C ← C − 1 End if C = 0 or CY = 1	2	3+10n	X	X	X	V	X	0	0	0	1	0	1	0	1
										0	0	0	1	0	1	1	0
CMPBKNC	[DE+], [HL+]	(DE+) − (HL+), C ← C − 1 End if C = 0 or CY = 1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
										0	0	1	0	0	1	1	0
	[DE−], [HL−]	(DE−) − (HL−), C ← C − 1 End if C = 0 or CY = 1	2	3+13n	X	X	X	V	X	0	0	0	1	0	1	0	1
										0	0	1	1	0	1	1	0
CPU Control																	
MOV	STBC, #byte	STBC ← byte*	4	11						0	0	0	0	1	0	0	1
										1	1	0	0	0	0	0	0
										$\overline{\text{Data}}$							
										Data							
	WDM, #byte	WDM ← byte*	4	11						0	0	0	0	1	0	0	1
										1	1	0	0	0	0	1	0
										$\overline{\text{Data}}$							
										Data							
SWRS		RSS ← $\overline{\text{RSS}}$	1	2						0	1	0	0	0	0	1	1
SEL	RBn	RSS ← 0, RBS_{2-0} ← n	2	3						0	0	0	0	0	1	0	1
										1	0	1	0	1	N_2	N_1	N_0
	RBn, ALT	RSS ← 1, RBS_{2-0} ← n	2	3						0	0	0	0	0	1	0	1
										1	0	1	1	1	N_2	N_1	N_0
NOP		No Operation	1	2						0	0	0	0	0	0	0	0
EI		IE ← 1 (Enable Interrupt)	1	3						0	1	0	0	1	0	1	1
DI		IE ← 0 (Disable Interrupt)	1	3						0	1	0	0	1	0	1	0

* Trap if data bytes are not ones complement.
If trap, then: (SP−1) ← PSW_H,
(SP−2) ← PSW_L, (SP−3) ← $(PC-4)_H$, (SP−4) ← $(PC-4)_L$,
PC_L ← (003CH), PC_H ← (003DH),
SP ← SP−4, IE ← 0.

NEC
NEC Electronics Inc.

µPD71P301
Memory Extender and Port Re-Creation Logic (Turbo Access Manager)

Description

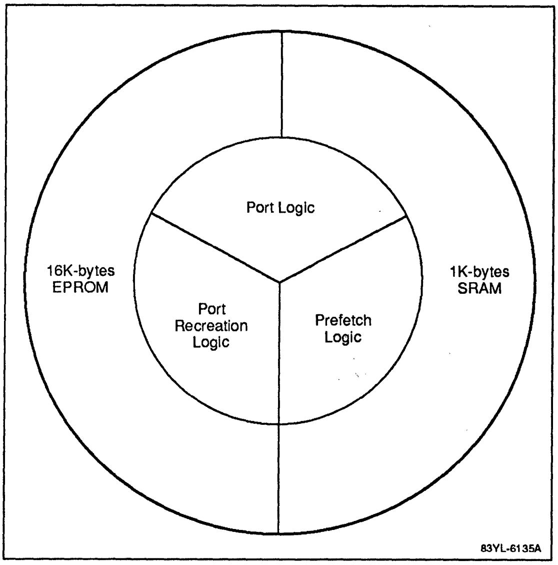
The µPD71P301 is a very high-performance port and memory expansion device that is designed to complement the µPD7832X microcomputer. The µPD71P301 contains special logic which allows the µPD7832X to perform full-speed memory access, as well as utilize lost I/O ports normally used for the external memory interface. In addition to the port re-creation logic, the part contains 1K bytes of static RAM and 16K bytes of EPROM or OTP memory. The µPD71P301 also has chip-select logic that allows cascading of multiple devices to form additional ports and memory.

The µPD71P301 is ideal for systems where external memory is required but access speed is critical to the application. This two-chip solution is also an excellent development system option since software and hardware can be fully emulated without high part count and speed limitations.

Features

- □ 16K-bytes UV EPROM or OTP; compatible with 27C256A
- □ 1K-bytes SRAM
- □ Two 8-bit I/O ports
- □ One cycle/byte instruction fetch
- □ 8-or 16-bit bus interface
- □ Instruction pre-fetch pointer
- □ Address latch
- □ Chip-select logic
- □ Address/data distinction
- □ Single 5 V supply
- □ CMOS silicon gate technology

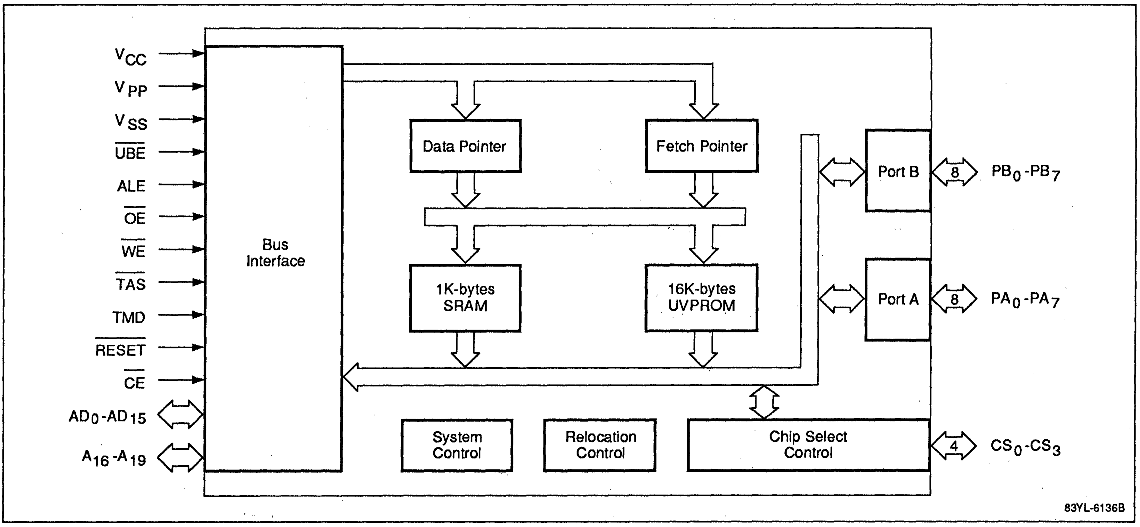
µPD71P301 Architecture



Ordering Information

Part Number	Package	Availability
µPD71P301GF-3BE	64-pin plastic QFP (OTP)	Now
µPD71P301GQ-36	64-pin plastic QUIP (OTP)	Now
µPD71P301KA	44-pin ceramic LCC (EPROM)	Now
µPD71P301KB	64-pin ceramic LCC (EPROM)	Now
µPD71P301L	44-pin PLCC (OTP)	Now
µPD71P301RQ	64-pin ceramic QUIP (EPROM)	Now

Block Diagram



Section 8
μPD722x Series:
Intelligent LCD Controller/Drivers

NEC
NEC Electronics Inc.

μPD7225
CMOS, Intelligent, Alphanumeric LCD Controller/Driver

Description

The μPD7225 is an intelligent peripheral device designed to interface most microprocessors with a wide variety of alphanumeric LCDs. It can directly drive any static or multiplexed LCD containing up to 4 backplanes and up to 32 segments and is easily cascaded for larger LCD applications. The μPD7225 communicates with a host microprocessor through an 8-bit serial interface. It includes a 7-segment numeric and a 14-segment alphanumeric segment decoder to reduce system software requirements. The μPD7225 is manufactured with a low power consumption CMOS process allowing use of a single power supply between 2.7 V and 5.5 V. It is available in a space-saving 52-pin plastic flat package.

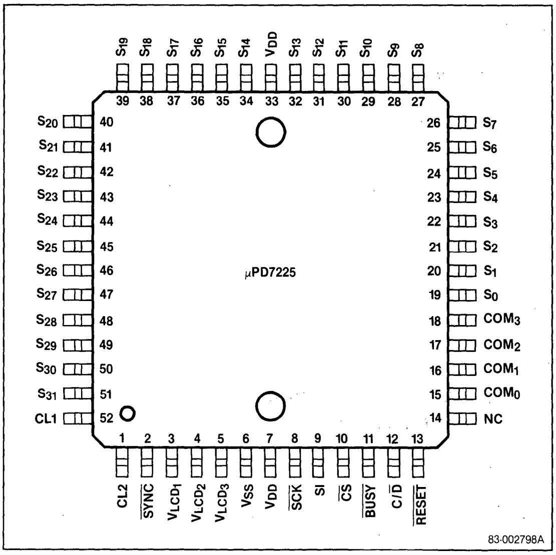
Features

- □ Single chip LCD controller with direct LCD drive
- □ Low cost serial interface to most microprocessors
- □ Compatible with
  - — 7-segment numeric LCD configurations up to 16 digits
  - — 14-segment alphanumeric LCD configurations up to 8 characters
- □ Selectable LCD drive configuration:
  - — Static, biplexed, triplexed, or quadruplexed
- □ 32-segment drivers
- □ Cascadable for larger LCD applications
- □ Selectable LCD bias voltage configuration:
  - — Static, 1/2 or 1/3
- □ Hardware logic blocks reduce system software requirements
  - — 8-bit serial interface
  - — Two 32 × 4-bit static RAMs for display data and blinking data storage
  - — Programmable segment decoding capability:
    - - 16-character, 7-segment numeric decoder
    - - 64-character, 14-segment USASCII alphanumeric decoder
  - — Programmable segment blinking capability
  - — Automatic synchronization of segment drivers with sequentially multiplexed backplane drivers
- □ Single power supply, variable from 2.7 V to 5.5 V
- □ Low power consumption CMOS technology
- □ Extended −40°C to +85°C temperature range

Ordering Information

Part Number	Package Type	Max Frequency of Operation
μPD7225G-00	52-pin plastic QFP	1 MHz

Pin Configuration



Pin Identification

No.	Symbol	Function
1	CL2	System clock output
2	$\overline{\text{SYNC}}$	Synchronization port
3–5	V_{LCD1}– V_{LCD3}	LCD bias voltage supply inputs
6	V_{SS}	Ground
7, 33	V_{DD}	Power
8	$\overline{\text{SCK}}$	Serial clock input
9	SI	Serial input
10	$\overline{\text{CS}}$	Chip select
11	$\overline{\text{BUSY}}$	Busy output
12	C/$\overline{\text{D}}$	Command or data select input
13	$\overline{\text{RESET}}$	Reset input
14	NC	No connection
15–18	COM_0–COM_3	LCD backplane driver outputs
19–32, 34–51	S_0–S_{31}	LCD segment driver outputs
52	CL1	System clock input

Pin Functions

COM$_0$-COM$_3$

LCD backplane driver outputs.

S$_0$-S$_{31}$

LCD segment driver outputs.

V$_{LCD1}$-V$_{LCD3}$

LCD bias voltage supply inputs to the LCD voltage controller. Apply appropriate voltages from a voltage ladder connected across V$_{DD}$.

SI

Serial input from the microprocessor.

$\overline{\text{SCK}}$

Serial clock input. Synchronizes 8-bit serial data transfer from the microprocessor to the µPD7225.

$\overline{\text{BUSY}}$

Handshake output indicates the µPD7225 is ready to receive the next data byte.

C/$\overline{\text{D}}$

Command/data select input. Distinguishes serially input data byte as a command or as display data.

$\overline{\text{CS}}$

Chip select input. Enables the µPD7225 for data input from the microprocessor. When $\overline{\text{CS}}$ is deselected, the display can be updated.

$\overline{\text{SYNC}}$

Synchronization port. For multichip operation, tie all $\overline{\text{SYNC}}$ lines together.

CL1

System clock input. Connect CL1 either to CL2 with a 180 kΩ resistor, or to an external clock source.

CL2

System clock output. Connect CL2 to CL1 with a 180 kΩ resistor, or leave open.

$\overline{\text{RESET}}$

Reset input. R/C circuit or pulse initializes the µPD7225 after power-up.

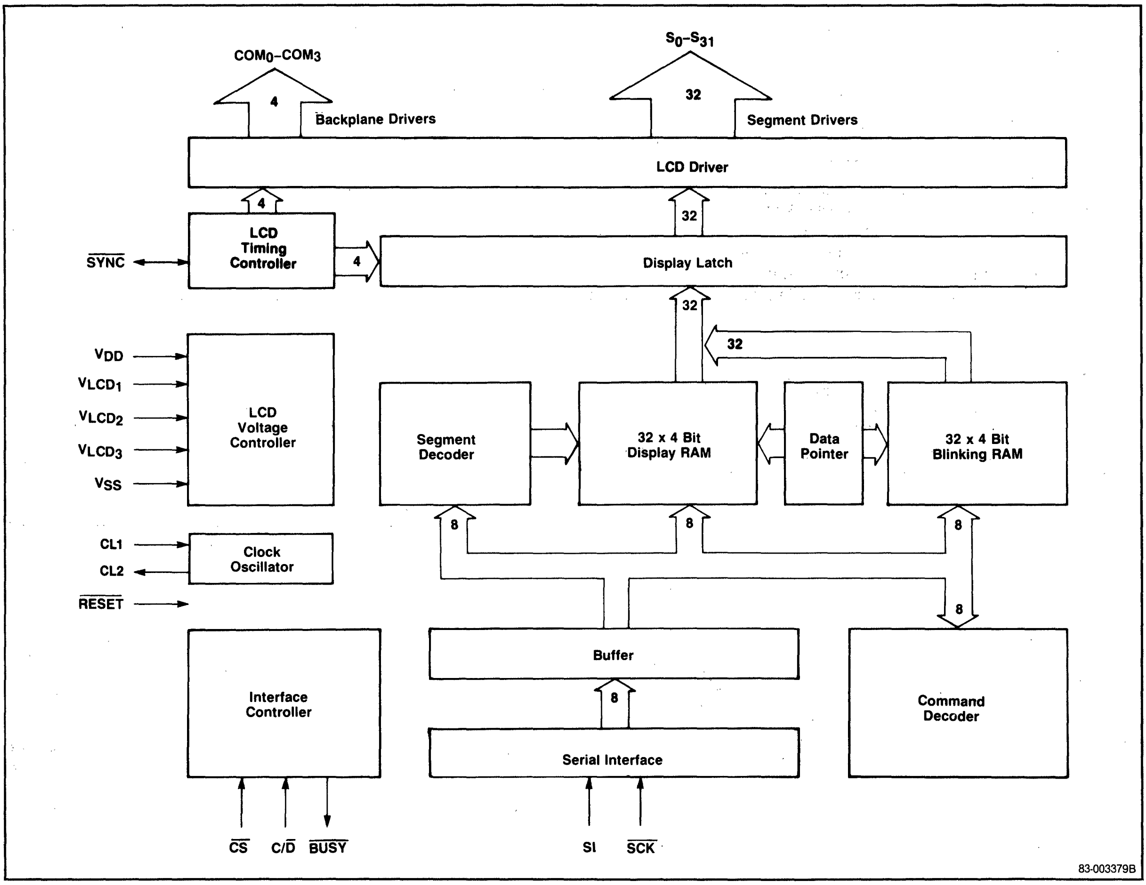
V$_{DD}$

Power supply positive. Apply single voltage ranging from 2.7 to 5.5 V for proper operation.

V$_{SS}$

Ground.

Block Diagram



8

Absolute Maximum Ratings

$T_A = 25°C$

Power supply voltage, V_{DD}	−0.3 V to +7 V
Input voltage, V_I	−0.3 V to V_{DD} +0.3 V
Output voltage, V_O	−0.3 V to V_{DD} +0.3 V
Operating temperature, T_{OPT}	−40°C to +85°C
Storage temperature, T_{STG}	−65°C to +150°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of the specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

$T_A = -10°C$ to $+70°C$, $V_{DD} = +5\text{ V} \pm 10\%$

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage low	V_{IL}	0		0.3 V_{DD}	V	
Input voltage high	V_{IH}	0.7 V_{DD}		V_{DD}	V	
Output voltage low	V_{OL1}			0.5	V	$\overline{\text{BUSY}}$, $I_{OL} = 100\ \mu A$
	V_{OL2}			1.0	V	$I_{OL} = 900\ \mu A$, $\overline{\text{SYNC}}$
Output voltage high	V_{OH}	V_{DD} −0.5			V	$\overline{\text{BUSY}}$, $\overline{\text{SYNC}}$ $I_{OH} = -7\ \mu A$
Input leakage current low	I_{LIL}			−2	μA	$V_{IL} = 0\text{ V}$
Input leakage current high	I_{LIH}			2	μA	$V_{IH} = V_{DD}$
Output leakage current	I_{LOL}			−2	μA	$V_{OL} = 0\text{ V}$
	I_{LOH}			2	μA	$V_{OH} = V_{DD}$
Output short circuit current	I_{OS}			−300	μA	$\overline{\text{SYNC}}$, $V_O = 1.0\text{ V}$
Backplane driver output impedance	R_{COM}		5	7	kΩ	COM_0–COM_3, $V_{DD} \geqslant V_{LCD}$ (Note 1)
Segment driver output impedance	R_{SEG}		7	14	kΩ	S_0–S_{31}, $V_{DD} \geqslant V_{LCD}$ (Note 1)
Supply current	I_{DD}		100	250	μA	CL1 external clock, $f_\phi = 200\text{ kHz}$

Note:

(1) Applies to static-, 1/2-, and 1/3-LCD bias voltage schemes.

DC Characteristics (cont)

$T_A = -40°C$ to $+85°C$, $V_{DD} = +5\text{ V} \pm 10\%$

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage low	V_{IL1}	0		0.3 V_{DD}	V	Except $\overline{\text{SCK}}$
	V_{IL2}	0		0.25 V_{DD}	V	$\overline{\text{SCK}}$
Input voltage high	V_{IH1}	0.7 V_{DD}		V_{DD}	V	Except $\overline{\text{SCK}}$
	V_{IH2}	0.75 V_{DD}		V_{DD}	V	$\overline{\text{SCK}}$
Output voltage low	V_{OL1}			0.5	V	$\overline{\text{BUSY}}$, $I_{OL} = 100\ \mu A$
	V_{OL2}			1.0	V	$I_{OL} = 1.05\text{ mA}$ $\overline{\text{SYNC}}$
Output voltage high	V_{OH}	V_{DD} −0.75			V	$\overline{\text{BUSY}}$, $\overline{\text{SYNC}}$, $I_{OH} = -7\ \mu A$
Input leakage current low	I_{LIL}			−2	μA	$V_{IL} = 0\text{ V}$
Input leakage current high	I_{LIH}			2	μA	$V_{IH} = V_{DD}$
Output leakage current	I_{LOL}			−2	μA	$V_{OL} = 0\text{ V}$
	I_{LOH}			2	μA	$V_{OH} = V_{DD}$
Output short circuit current	I_{OS}			−350		$\overline{\text{SYNC}}$, $V_O = 1.0\text{ V}$
Backplane driver output impedance	R_{COM}		5	8	kΩ	COM_0–COM_3, $V_{DD} \geqslant V_{LCD}$ (Note 1)
Segment driver output impedance	R_{SEG}		7	20	kΩ	S_0–S_{31}, $V_{DD} \geqslant V_{LCD}$ (Note 1)
Supply current	I_{DD}		90	250	μA	CL1 external clock, $V_{DD} = 3.0\text{ V} \pm 10\%$, $f_\phi = 180\text{ kHz}$

Note:

(1) Applies to static-, 1/2-, and 1/3-LCD bias voltage schemes.

Capacitance

$T_A = 25°C$, $f_\phi = 1\text{ MHz}$

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions(1)**
Input capacitance	C_I			10	pF	
Output capacitance	C_{O1}			20	pF	Except $\overline{\text{BUSY}}$
	C_{O2}			15	pF	$\overline{\text{BUSY}}$
I/O capacitance	C_{IO}			15	pF	$\overline{\text{SYNC}}$
Clock capacitance	C_ϕ			30	pF	CL1 input

Note:

(1) All unmeasured pins returned to 0 V.

AC Characteristics

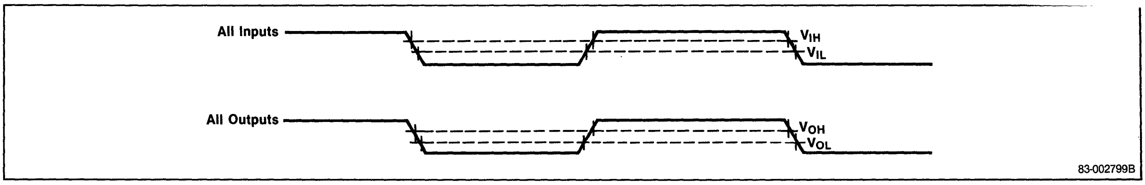
T_A = −40°C to +85°C, V_{DD} = +5 V ±10%

Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Clock frequency	f_{ϕ}	75		180	kHz	
	f_{OSC}	80	130	180	kHz	R = 180 kΩ +5%
Clock pulse width low	$t_{\phi WL}$	2		10	μs	CL1, external clock
Clock pulse width high	$t_{\phi WH}$	2		10	μs	CL1, external clock
$\overline{\text{SCK}}$ cycle	t_{CYK}	1.2			μs	
$\overline{\text{SCK}}$ pulse width low	t_{KWL}	500			ns	
$\overline{\text{SCK}}$ pulse width high	t_{KWH}	500			ns	
$\overline{\text{BUSY}}$ ↑ to $\overline{\text{SCK}}$ ↓ hold time	t_{BHK}	0			ns	
SI setup time to $\overline{\text{SCK}}$ ↑	t_{ISK}	100			ns	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{IHK}	200			ns	
8th $\overline{\text{SCK}}$ ↑ to $\overline{\text{BUSY}}$ ↓ delay time	t_{KDB}			3	μs	C_L = 50 pF
$\overline{\text{CS}}$ ↓ to $\overline{\text{BUSY}}$ ↓ delay time	t_{CDB}			1.5	μs	C_L = 50 pF
C / $\overline{\text{D}}$ setup time to 8th $\overline{\text{SCK}}$ ↑	t_{DSK}	9			μs	
C / $\overline{\text{D}}$ hold time after 8th $\overline{\text{SCK}}$ ↑	t_{DHK}	1			μs	
$\overline{\text{CS}}$ hold time after 8th $\overline{\text{SCK}}$ ↑	t_{CHK}	1			μs	
$\overline{\text{CS}}$ pulse width low	t_{CWL}	8 / f_{ϕ}			μs	
$\overline{\text{CS}}$ pulse width high	t_{CWH}	8 / f_{ϕ}			μs	
$\overline{\text{SYNC}}$ load capacitance	C_L			50	pF	f_0 = 200 kHz

T_A = 0°C to +70°C, V_{DD} = 2.7 V to 5.5 V

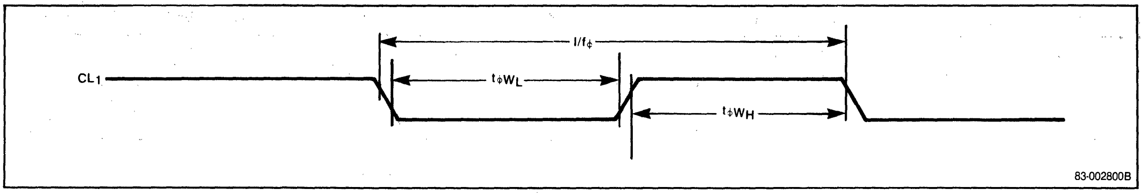
Parameter	Symbol	Limits Min	Limits Typ	Limits Max	Unit	Test Conditions
Clock frequency	f_{ϕ}	50		140	kHz	
	f_{OSC}	50	100	140	kHz	R = 180 kΩ +5%, V_{DD} = 3.0 V ± 10%
Clock pulse width low	$t_{\phi WL}$	3		16	μs	CL1, external clock
Clock pulse width high	$t_{\phi WH}$	3		16	μs	CL1, external clock
$\overline{\text{SCK}}$ cycle	t_{CYK}	4			μs	
$\overline{\text{SCK}}$ pulse width low	t_{KWL}	1.8			μs	
$\overline{\text{SCK}}$ pulse width high	t_{KWH}	1.8			μs	
$\overline{\text{BUSY}}$ ↑ to $\overline{\text{SCK}}$ ↓ hold time	t_{BHK}	0			ns	
SI setup time to $\overline{\text{SCK}}$ ↑	t_{ISK}	1			μs	
SI hold time after $\overline{\text{SCK}}$ ↑	t_{IHK}	1			μs	
8th $\overline{\text{SCK}}$ ↑ to $\overline{\text{BUSY}}$ ↓ delay time	t_{KDB}			5	μs	C_L = 50 pF
$\overline{\text{CS}}$ ↓ to $\overline{\text{BUSY}}$ ↓ delay time	t_{CDB}			5	μs	C_L = 50 pF
C / $\overline{\text{D}}$ setup time to 8th $\overline{\text{SCK}}$ ↑	t_{DSK}	18			μs	
C / $\overline{\text{D}}$ hold time after 8th $\overline{\text{SCK}}$ ↑	t_{DHK}	1			μs	
$\overline{\text{CS}}$ hold time after 8th $\overline{\text{SCK}}$ ↑	t_{CHK}	1			μs	
$\overline{\text{CS}}$ pulse width low	t_{CWL}	8 / f_{ϕ}			μs	
$\overline{\text{CS}}$ pulse width high	t_{CWH}	8 / f_{ϕ}			μs	
$\overline{\text{SYNC}}$ load capacitance	C_L			50	pF	f_{ϕ} = 200 kHz

AC Timing Characteristics

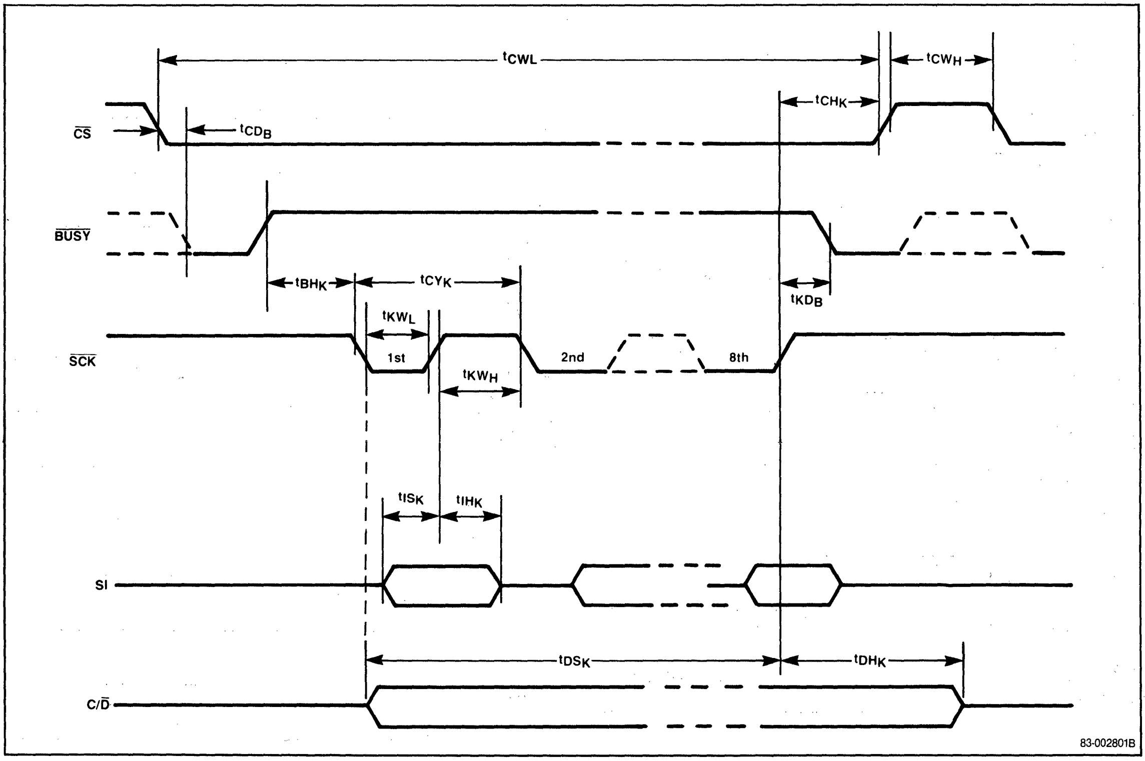


Timing Waveforms

Clock



Serial Interface



Instruction Set (Note 1)

		Hex	Operation Code							
Command	**Description**	**Code**	**D_7**	**D_6**	**D_5**	**D_4**	**D_3**	**D_2**	**D_1**	**D_0**
Mode Set	Initialize the μPD7225, including selection of: 1) LCD drive configuration 2) LCD bias voltage configuration 3) LCD frame frequency	40–5F	0	1	0	d_4	d_3	d_2	d_1	d_0
Unsynchronous Data Transfer	Synchronize display RAM data transfer to display latch with $\overline{\text{CS}}$	30	0	0	1	1	0	0	0	0
Synchronous Data Transfer	Synchronize display RAM data transfer to display latch with LCD drive cycle	31	0	0	1	1	0	0	0	1
Interrupt Data Transfer	Interrupt display RAM data transfer to display latch	38	0	0	1	1	1	0	0	0
Load Data Pointer	Load data pointer with 5 bits of immediate data	E0–FF	1	1	1	d_4	d_3	d_2	d_1	d_0
Clear Display RAM	Clear the display RAM and reset the data pointer	20	0	0	1	0	0	0	0	0
Write Display RAM	Write 4 bits of immediate data to the display RAM location addressed by the data pointer; increment data pointer	D0–DF	1	1	0	1	d_3	d_2	d_1	d_0
AND Display RAM	Perform a logical AND between the display RAM data addressed by the data pointer and 4 bits of immediate data; write result to same display RAM location. Increment data pointer	90–9F	1	0	0	1	d_3	d_2	d_1	d_0
OR Display RAM	Perform a logical OR between the display RAM data addressed by the data pointer and 4 bits of immediate data; write result to same display RAM location; increment data pointer	B0–BF	1	0	1	1	d_3	d_2	d_1	d_0
Enable Segment Decoder	Start use of the segment decoder	15	0	0	0	1	0	1	0	1
Disable Segment Decoder	Stop use of the segment decoder	14	0	0	0	1	0	1	0	0
Enable Display	Turn on the LCD	11	0	0	0	1	0	0	0	1
Disable Display	Turn off the LCD	10	0	0	0	1	0	0	0	0
Clear Blinking RAM	Clear the blinking RAM and reset the data pointer	00	0	0	0	0	0	0	0	0
Write Blinking RAM	Write 4 bits of immediate data to the blinking RAM location addressed by the data pointer; increment data pointer	C0–CF	1	1	0	0	d_3	d_2	d_1	d_0
AND Blinking RAM	Perform a logical AND between blinking RAM data addressed by the data pointer and 4 bits of immediate data; write result to same blinking location; increment data pointer	80–8F	1	0	0	0	d_3	d_2	d_1	d_0
OR Blinking RAM	Perform a logical OR between blinking RAM data addressed by the data pointer and 4 bits of immediate data; write result to same blinking location; increment data pointer	A0–AF	1	0	1	0	d_3	d_2	d_1	d_0
Enable Blinking	Start segment blinking at the frequency specified by 1 bit of immediate data	1A–1B	0	0	0	1	1	0	1	d_0
Disable Blinking	Stop segment blinking	18	0	0	0	1	1	0	0	0

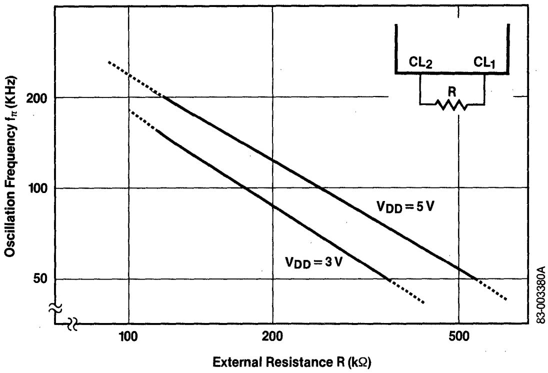
Note:

(1) Details of operation and application examples can be found in the μPD7225 Intelligent Alphanumeric LCD Controller/Driver Technical Manual.

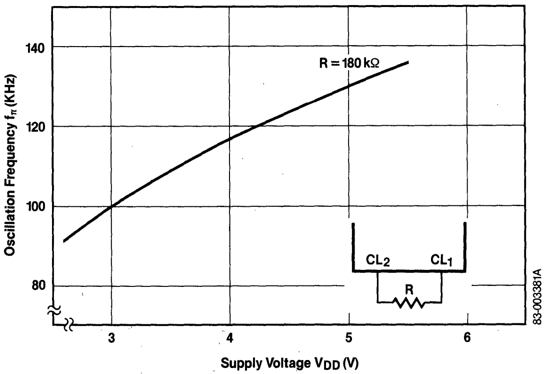
Operating Characteristics

$T_A = 25°C$

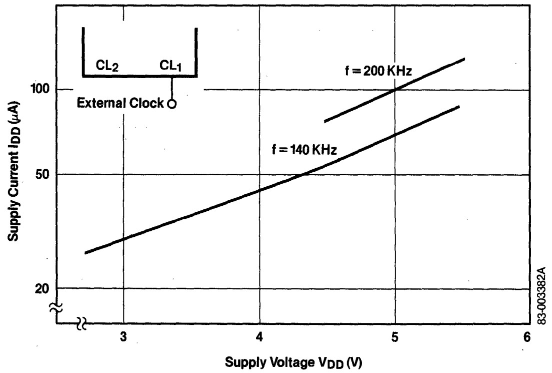
External Resistance vs Oscillation Frequency



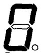
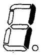
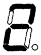
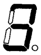
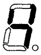
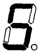
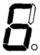
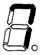
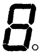
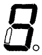
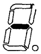
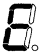
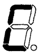
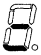
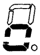
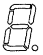
Supply Voltage vs Oscillation Frequency



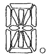
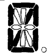
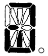
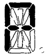
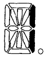
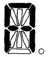
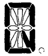
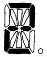
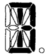
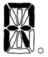
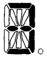
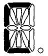
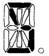
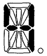
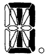
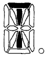
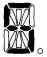
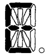
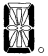
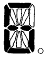
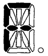
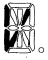
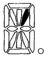
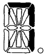
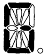
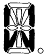
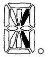
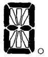
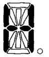
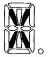
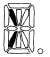
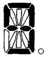
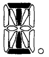
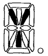
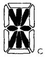
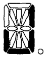
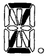
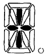
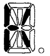
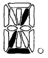
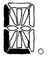
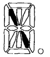
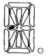
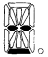
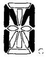
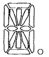
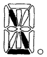
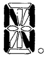
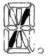
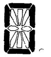
Supply Voltage vs Supply Current



7-Segment Numeric Data Decoder Character Set

		Decoded Display RAM Data				
		Triplexed			Quadruplexed	
		Display RAM Address			Display RAM Address	
Display Byte (HEX)	Character	n+2	n+1	n	n+1	n
00		3	5	3	D	7
01		0	0	3	0	6
02		2	7	1	E	3
03		0	7	3	A	7
04		1	2	3	3	6
05		1	7	2	B	5
06		3	7	2	F	5
07		0	1	3	0	7
08		3	7	3	F	7
09		1	7	3	B	7
0A		3	2	0	2	0
0B		3	7	0	F	1
0C		3	5	0	D	1
0D		0	6	0	A	0
0E		2	6	2	E	4
0F		0	0	0	0	0

14-Segment Alphanumeric Data Decoder Character Set

Display Byte (HEX)	Char.	Display RAM Address				Display Byte (HEX)	Char.	Display RAM Address				Display Byte (HEX)	Char.	Display RAM Address				Display Byte (HEX)	Char.	Display RAM Address			
		n+3	n+2	n+1	n			n+3	n+2	n+1	n			n+3	n+2	n+1	n			n+3	n+2	n+1	n
A0		0	0	0	0	B0		4	7	E	2	C0		A	7	C	0	D0		2	3	6	4
A1		Invalid				B1		0	6	0	0	C1		2	7	6	4	D1		0	7	E	8
A2		Invalid				B2		2	3	C	4	C2		8	7	8	5	D2		2	3	6	C
A3		Invalid				B3		2	7	8	4	C3		0	1	E	0	D3		1	5	8	4
A4		Invalid				B4		2	6	2	4	C4		8	7	8	1	D4		8	1	0	1
A5		Invalid				B5		2	5	A	4	C5		2	1	E	4	D5		0	6	E	0
A6		Invalid				B6		2	5	E	4	C6		2	1	6	4	D6		4	0	6	2
A7		0	0	0	2	B7		0	7	0	0	C7		0	5	E	4	D7		4	6	6	8
A8		0	0	0	A	B8		2	7	E	4	C8		2	6	6	4	D8		5	0	0	A
A9		5	0	0	0	B9		2	7	A	4	C9		8	1	8	1	D9		9	0	0	2
AA		F	0	0	F	BA		Invalid				CA		0	6	C	0	DA		4	1	8	2
AB		A	0	0	5	BB		Invalid				CB		2	0	6	A	DB		Invalid			
AC		Invalid				BC		4	0	8	2	CC		0	0	E	0	DC		1	0	0	8
AD		2	0	0	4	BD		2	0	8	4	CD		1	6	6	2	DD		Invalid			
AE		Invalid				BE		1	0	8	8	CE		1	6	6	8	DE		Invalid			
AF		4	0	0	2	BF		Invalid				CF		0	7	E	0	DF		Invalid			

NEC
NEC Electronics Inc.

μPD7227 CMOS, Intelligent, Dot-Matrix LCD Controller/Driver

Description

The μPD7227 intelligent dot-matrix LCD controller/driver is a peripheral device designed to interface most microprocessors with a wide variety of dot matrix LCDs. It can directly drive any multiplexed LCD organized as 8 rows by 40 columns, and is easily cascaded up to 16 rows and 280 columns. The μPD7227 is equipped with several hardware logic blocks, such as an 8-bit serial interface, ASCII character generator, 40 x 16 static RAM with full read/write capability, and an LCD timing controller; all of which reduce microprocessor system software requirements. The μPD7227 is manufactured with a single 5 V CMOS process, and is available in a space-saving 64-pin plastic flat package.

Features

□ Single-chip LCD controller with direct LCD drive
□ Compatible with most microprocessors
□ Eight row drives
- — Designed for dot-matrix LCD configurations up to 280 dots
- — Designed for 5 x 7 dot-matrix character LCD configuration up to 8 characters
- — Cascadable to 16 row drives

□ 40 column drives
- — Cascadable to 280 column drives

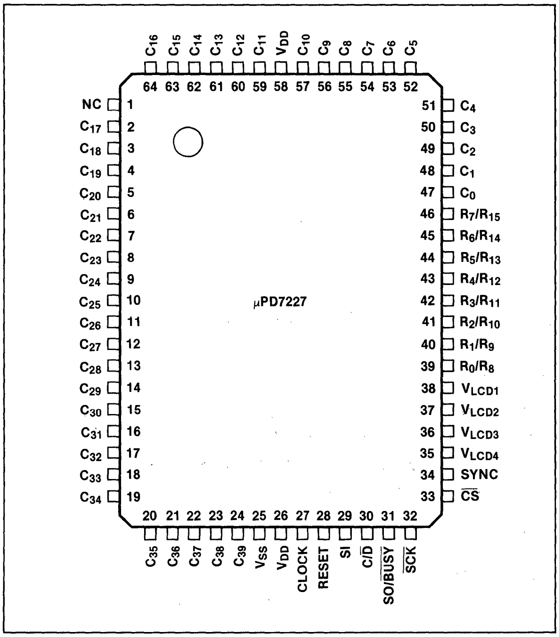
□ Hardware logic blocks reduce system software requirements
- — 8-bit serial interface for communication
- — ASCII 5 x 7 dot-matrix character generator with 64-character vocabulary
- — 40 x 16-bit static RAM for data storage, retrieval, and complete back-up memory capability.
- — Voltage controller generates LCD bias voltages
- — Timing controller synchronizes column drives with sequentially-multiplexed row drives

□ Single +5 V power supply
□ CMOS technology

Ordering Information

Part Number	Package Type	Max Frequency of Operation
μPD7227G-12	64-pin plastic QFP	1000 kHz

Pin Configuration



Pin Identification

No.	Symbol	Function
1	NC	No connection
2-24, 47-57, 59-64	C_0-C_{39}	LCD column driver outputs
25	V_{SS}	Ground
26, 58	V_{DD}	Power
27	CLOCK	System clock input
28	RESET	Reset input
29	SI	Serial input
30	C/$\overline{D}$	Command or data select input
31	SO/$\overline{BUSY}$	Serial output or busy output
32	$\overline{SCK}$	Serial clock input
33	CS	Chip select input
34	SYNC	Synchronization port
35-38	V_{LCD1}-V_{LCD4}	LCD bias voltage supply inputs
39-46	R_0/R_8-R_7/R_{15}	LCD row driver outputs

Pin Functions

C_0-C_{39}

LCD column driver outputs.

R0/8-R7/15

LCD row driver outputs.

V_{LCD1}-V_{LCD4}

LCD bias voltage supply inputs to the LCD voltage controller. Apply appropriate voltages from a voltage ladder connected across VDD.

SI

Serial input from the microprocessor.

SO/$\overline{\text{BUSY}}$

Serial output from the μPD7227 to the microprocessor when in read mode and C/$\overline{\text{D}}$ is low. When $\overline{\text{BUSY}}$ (active low), handshake output indicates the μPD7227 is ready to receive/send the next data byte.

$\overline{\text{SCK}}$

Serial clock input. Synchronizes 8-bit serial data transfer between the microprocessor and μPD7227.

C/$\overline{\text{D}}$

Command/data select input. Distinguishes serially input data byte as a command or as display data.

$\overline{\text{CS}}$

Chip select input. Enables the μPD7227 for communication with the microprocessor.

SYNC

Synchronization port. For multichip operation, tie all SYNC lines together and configure with the MODE SET command.

CLOCK

System clock input. Connect to external clock source.

RESET

Reset input. RC circuit or pulse initializes the μPD7227 after power-up.

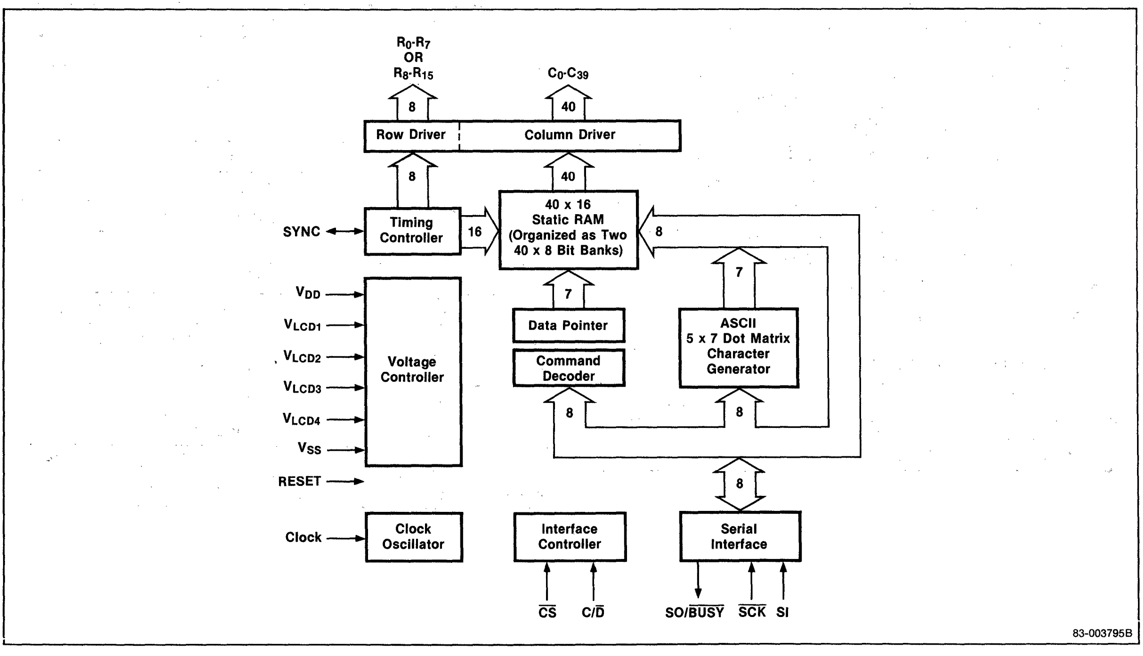
V_{DD}

Power supply positive. Apply single voltage 5 V ± 10% for proper operation.

V_{SS}

Ground.

Block Diagram



Absolute Maximum Ratings

T_A = 25°C

Power supply, V_{DD}	−0.3 V to +7.0 V
All inputs and outputs with respect to V_{CC}	−0.3 V to V_{DD} +0.3 V
Storage temperature, T_{STG}	−65°C to +150°C
Operating temperature, T_{OPT}	−10°C to +70°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

T_A = 25°C, V_{DD} = 0V

		Limits			
Parameter	**Symbol**	**Min**	**Max**	**Unit**	**Test Conditions**
Input capacitance	C_I		10	pF	fϕ = 1 MHz
Output capacitance	C_O		25	pF	Unmeasured pins returned to ground.
Input/output capacitance	C_{IO}		15	pF SYNC	

DC Characteristics

T_A = −10°C to +70°C, V_{DD} = +5.0V ± 10%

		Limits				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input voltage, high	V_{IH}	0.7 V_{DD}		V_{DD}	V	
Input voltage, low	V_{IL}	0		0.3 V_{DD}	V	
Input leakage current, high	I_{LIH}			+10	μA	V_{IH} = V_{DD}
Input leakage current, low	I_{LIL}			−10	μA	V_{IH} = 0V
Output voltage, high	V_{OH1}	V_{DD}-0.5			V	SO/BUSY, I_{OH} = −400 μA
	V_{OH2}	V_{DD}-0.5			V	SYNC, I_{OH} = −100 μA
Output voltage, low	V_{OL1}			0.45	V	SO/BUSY, I_{OL} = +1.7 mA
				0.45	V	SYNC, I_{OL} = +100 μA
Output leakage current, high	I_{LOH}			+10	μA	V_{OH} = V_{DD}
Output leakage current, low	I_{LOL}			−10	μA	V_{OL} = 0V
LCD operating voltage	V_{LCD}	3.0		V_{DD}	V	8-row multiplexed LCD drive configuration
			V_{DD}		V	16-row multiplexed LCD drive configuration
Row drive output impedance	R_{ROW}		4	8	kΩ	
Column drive output impedance	R_{COLUMN}		10	15	kΩ	
Supply current	I_{DD}		200	400	μA	f0 = 400 KHz

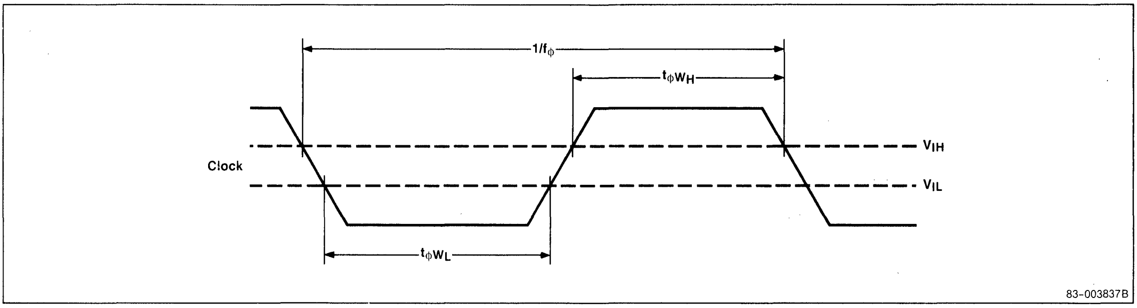
AC Characteristics

T_A = −10°C to +70°C, V_{DD} = +5.0V ± 10%

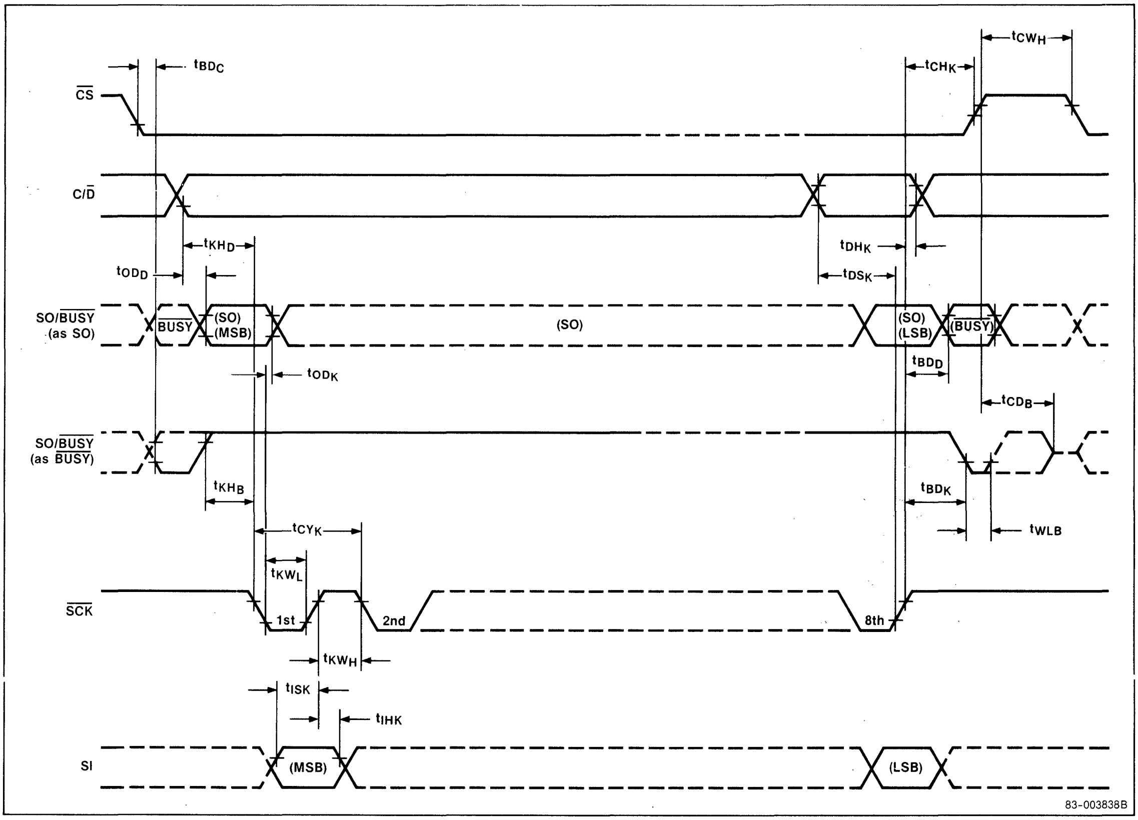
		Limits			Test
Parameter	Symbol	Min	Max	Unit	Conditions
Clock frequency	f_ϕ	100	1000	KHz	
Clock pulse width high	$t_{\phi WH}$	400		ns	
Clock pulse width low	$t_{\phi WL}$	400		ns	
$\overline{\text{SCK}}$ cycle	t_{CYK}	0.9		μs	
$\overline{\text{SCK}}$ pulse width high	t_{KWH}	400		ns	
$\overline{\text{SCK}}$ pulse width low	t_{KWL}	400		ns	
$\overline{\text{SCK}}$ hold time after $\overline{\text{BUSY}}$↑	t_{KHB}	0		ns	
SI setup time to $\overline{\text{SCK}}$↑	t_{ISK}	100		ns	
SI hold time after $\overline{\text{SCK}}$↑	t_{IHK}	250		ns	
SO delay time after $\overline{\text{SCK}}$↓	t_{ODK}		320	ns	C_{LOAD} = 50 pF
SO delay time after C/$\overline{\text{D}}$↓	t_{ODD}		2	μs	
$\overline{\text{SCK}}$ hold time after C/$\overline{\text{D}}$↓	t_{KHD}	2		μs	
$\overline{\text{BUSY}}$ delay time after 8th $\overline{\text{SCK}}$↑	t_{BDK}		3	μs	C_{LOAD} = 50 pF
$\overline{\text{BUSY}}$ delay time after C/$\overline{\text{D}}$↑	t_{BDD}		2	μs	
$\overline{\text{BUSY}}$ delay time after $\overline{\text{CS}}$↓	t_{BDC}		2	μs	
C/$\overline{\text{D}}$ setup time to 8th $\overline{\text{SCK}}$↑	t_{DSK}	2		μs	
C/$\overline{\text{D}}$ hold time after 8th $\overline{\text{SCK}}$↑	t_{DHK}	2		μs	
$\overline{\text{CS}}$ hold time after 8th $\overline{\text{SCK}}$↑	t_{CHK}	2		μs	
$\overline{\text{CS}}$ pulse width high	t_{CWH}	2/fϕ		μs	
$\overline{\text{CS}}$↑ delay time to $\overline{\text{BUSY}}$ floating	t_{CDB}	2		μs	C_{LOAD} = 50 pF
SYNC load capacitance	C_{LOADS}		100	pF	
$\overline{\text{BUSY}}$ low level width	t_{WLB}	18	64	1/fϕ	C_{LOAD} = 50 pF

Timing Waveforms

Clock Waveform



Serial Interface

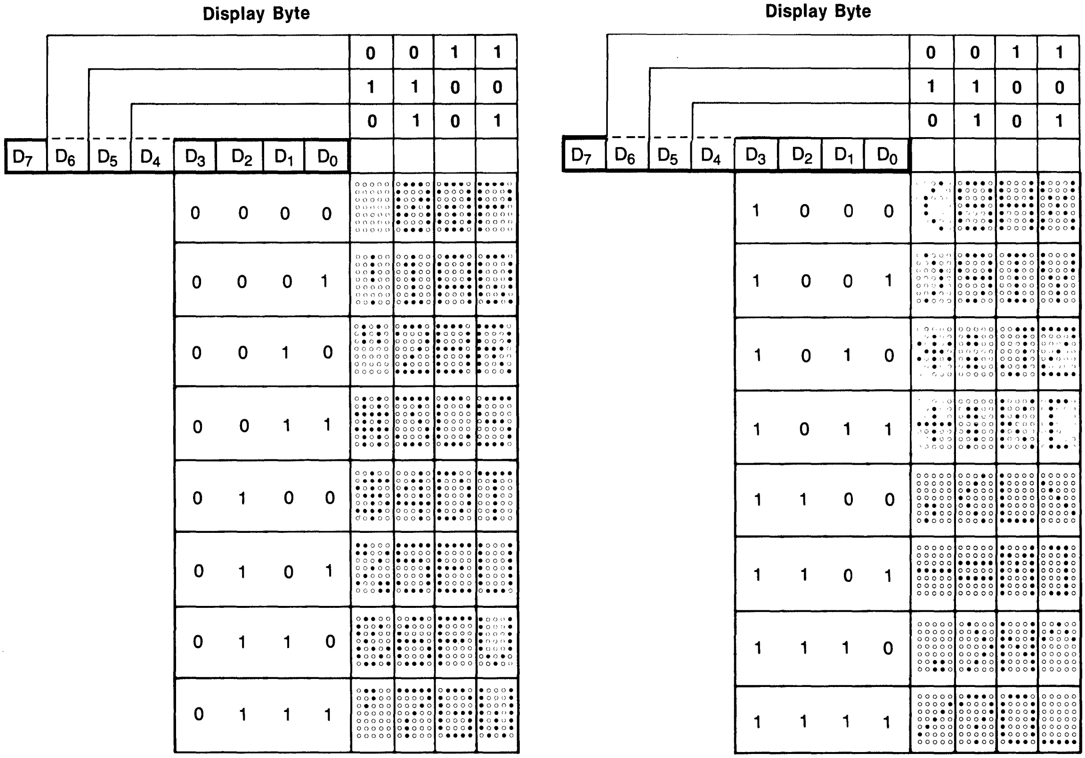


Command Summary

		Instruction Code								
		Binary								
Command	**Description**	**D_7**	**D_6**	**D_5**	**D_4**	**D_3**	**D_2**	**D_1**	**D_0**	**HEX**
Mode Set	Initialize the μPD7227, including selection of 1. LCD drive configuration 2. Row driver port function 3. RAM bank 4. SYNC port function	0	0	0	1	1	D_2	D_1	D_0	18-1F
Frame Frequency Set	Set LCD frame frequency	0	0	0	1	0	D_2	D_1	D_0	10-14
Load Data Pointer	Load data pointer with 7 bits of immediate data	1	D_6	D_5	D_4	D_3	D_2	D_1	D_0	80-E7
Write Mode	Write display byte in serial register to RAM location addressed by data pointer; modify data pointer	0	1	1	0	0	1	D_1	D_0	64-67
Read Mode	Load RAM contents addressed by data pointer into serial register for output; modify data pointer	0	1	1	0	0	0	D_1	D_0	60-63
AND Mode	Perform a logical AND between the display byte in the serial register and the RAM contents addressed by data pointer; write result to same RAM location; modify data pointer	0	1	1	0	1	1	D_1	D_0	6C-6F
OR Mode	Perform a logical OR between the display byte in the serial register and the RAM contents addressed by data pointer; write result to same RAM location; modify data pointer	0	1	1	0	1	0	D_1	D_0	68-6B
Character Mode	Decode display byte in serial register into 5 x 7 character with character generator; write character to RAM location addressed by data pointer; increment data pointer by 5	0	1	1	1	0	0	1	0	72
Set Bit	Set single bit of RAM location addressed by data pointer; modify data pointer	0	1	0	D_4	D_3	D_2	D_1	D_0	40-5F
Reset Bit	Reset single bit of RAM location addressed by data pointer; modify data pointer	0	0	1	D_4	D_3	D_2	D_1	D_0	20-3F
Enable Display	Turn on the LCD	0	0	0	0	1	0	0	1	09
Disable Display	Turn off the ICD	0	0	0	0	1	0	0	0	08

Further details of operation can be found in the μPD7227 intelligent dot-matrix LCD controller/driver technical manual.

5 × 7 Character Set as Generated in μPD7227



NEC
NEC Electronics Inc.

μPD7228/28A CMOS, Intelligent, Dot-Matrix LCD Controller/Driver

Description

The μPD72228/28A controller/driver is a peripheral CMOS device designed to interface most microprocessors with a wide variety of dot-matrix LCDs. It can directly drive any multiplexed LCD organized as 8 rows by 50 columns or 16 rows by 42 columns.

The μPD7228/28A has a standby function to conserve power. It is equipped with an 8-bit serial interface, a 4-bit parallel interface, character generators, a 50 x 16 static RAM with full read/write capability, and an LCD timing controller, all of which reduce microprocessor system software requirements.

The μPD7228/28A operates with a single +5-volt power supply and is available in a space-saving 80-pin plastic QFP package.

Features

□ LCD direct drive

□ 8-or 16-line multiplexing drive possible with single-chip
— 8-line multiplexing: 400 (50 x 8) dots
— 16-line multiplexing: 672 (42 x 16) dots

□ 8-line or 16-line multiplexing drive with n chip configuration
— 8-line multiplexing: n x 400 (n x 50 x 8) dots
— 16-line multiplexing: n x 800 (n x 50 x 16) dots

□ RAM: 2 x 50 x 8 bits for display data storage

□ Programmer designated dot (graphics) display

□ 5 x 7 dot-matrix display by on-chip character generator
— ASCII (alphanumerics, others): 96 characters
— JIS (Japan Industrial Standard), Katakana and others: 64 characters.

□ Cursor operating command

□ 8-bit serial interface compatible with μPD7500, μCOM-87/87LC

□ 4-bit parallel interface compatible with μPD7500, μCOM-84/84C

□ Standby function

□ CMOS technology

□ Single +5-volt power supply

□ Extended −40 to +85°C temperature range (μPD7228A)

Ordering Information

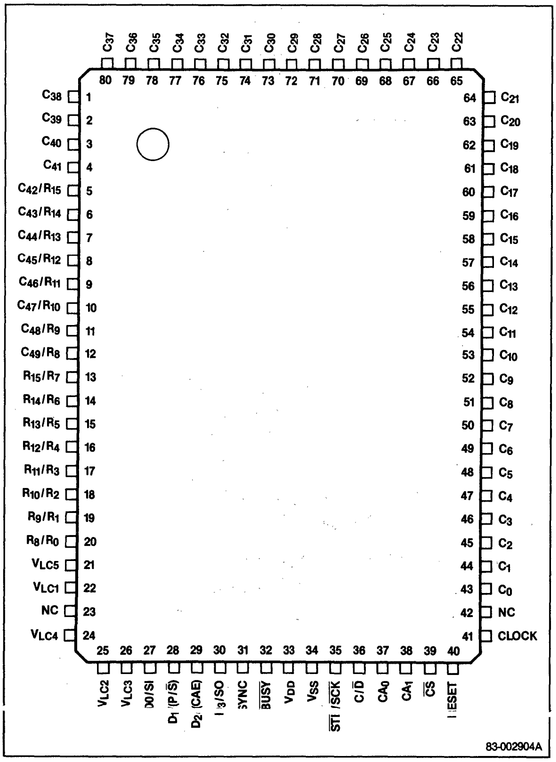
Part No.	Package
μPD7228G-12	80-pin plastic QFP
μPD7228AG-12 (Note 1)	80-pin plastic QFP

Notes:

(1) μPD7228A version has extended temperature range and LCD voltage range.

Pin Configuration

80-Pin Plastic QFP



Pin Identification

Symbol	Function
C_0-C_{41}	LCD column drive outputs
C_{42}-C_{49}/R_{15}-R_8	LCD column/row drive outputs
R_{15}-R_8/R_7-R_8	LCD row drive outputs
V_{LC1}-V_{LC5}	LCD power supply
NC	No connection
D_0/S_I	Data bus 0/Serial input
D_1(P/$\overline{S}$)	Data bus 1 (Parallel/serial select)
D_2(CAE)	Data bus 2 (Chip address enable)
D_3/SO	Data bus 3/Serial output
SYNC	Synchronization signal input/output
$\overline{BUSY}$	Busy signal output
V_{DD}	Power supply
V_{SS}	Ground
$\overline{STB}/\overline{SCK}$	Strobe/Serial clock input
C/$\overline{D}$	Command/data select input
CA_0, CA_1	Chip address select inputs
$\overline{CS}$	Chip select input
RESET	Reset signal input
CLOCK	System clock input

PIN FUNCTIONS

D_0-D_3 (Data Bus)

In parallel interface mode, D_0-D_3 are input/output pins for 4-bit parallel data. Data on these lines is read at the rising edge of $\overline{STB}$. The 4 bits read on the first $\overline{STB}$ are loaded into the highest 4 bits of the serial/parallel register. The 4 bits read on the second $\overline{STB}$ are loaded into the lowest 4 bits of the register.

The contents of the serial/parallel register are output to these pins on the falling edge of $\overline{STB}$. As in the above case, the high-order 4 bits correspond to the first $\overline{STB}$, and the low-order 4 bits to the second $\overline{STB}$.

In serial interface mode, D_0 is a serial data input pin and D_3 is a serial data output pin. D_1 selects serial or parallel interface mode (P/$\overline{S}$), and D_2 is the chip address enable pin (CAE).

SI Serial Data-In (Input Common to D_0)

In serial interface mode, SI inputs serial data. Data on SI is loaded into the serial/parallel register at the rising edge of $\overline{SCK}$. The first data loaded is the most significant bit. To eliminate noise errors, SI uses the Schmitt-trigger input.

SO Serial Data-Out (Output Common to D_3)

In serial interface mode, SO is an output pin for serial data. The contents of the serial/parallel register are output to the SO pin, beginning with the most significant bit, on the falling edge of $\overline{SCK}$.

P/S Parallel/Serial Select (Input Common to D_1)

This pin sets parallel interface mode if it is high at the falling edge of RESET (at reset release). If it is low at the falling edge of RESET, it selects serial interface mode. The Schmitt-trigger prevents noise errors.

CAE Chip Address Enable (Input Common to D_2)

This pin is used only during serial interface mode; that is, when P/$\overline{S}$ is low at the falling edge of RESET. To enable chip addressing, the CAE line must be high at the falling edge of RESET. In parallel interface mode (when P/$\overline{S}$ is high at the falling edge of RESET), the chip addressing function is enabled regardless of the logic state of CAE at the falling edge of RESET. The Schmitt-trigger input prevents noise errors.

CA_0-CA_1 (Chip Address)

These input pins allow you to address the µPD7228/28A in a multichip configuration used for driving logic displays. During parallel interface mode, CA_0 and CA_1 are compared to chip address data sent from the CPU regardless of CAE status during a reset.

However, during serial interface mode, CA_0 and CA_1 are compared with chip address data from the CPU only when CAE enables chip addressing.

In multichip configurations, the device is selected if $\overline{CS}$ = 0 and CA_0 and CA_1 match the chip address generated by the CPU. This address is the low 2 bits of the first 8-bit data input after $\overline{CS}$ = 0.

In serial interface mode, if chip address selection is not used, connect CA_0 and CA_1 to ground.

$\overline{CS}$ (Chip Select)

$\overline{CS}$ is an active-low chip select input pin. When you are not using the chip address selection function, the STB/$\overline{SCK}$ and C/$\overline{D}$ inputs are enabled if a low input is sent to $\overline{CS}$.

When you are using the chip address select function, if $\overline{CS}$ is brought low and the chip address data matches CA_0-CA_1, then $\overline{STB}/\overline{SCK}$ and C/$\overline{D}$ are enabled.

When $\overline{CS}$ is made high, D_0-D_3 and $\overline{BUSY}$ are placed in a high-impedance state. The Schmitt-trigger input prevents noise errors.

$\overline{\text{STB}}/\overline{\text{SCK}}$ (Strobe/Serial Clock)

In parallel interface mode, this is the strobe signal input pin ($\overline{\text{STB}}$) for 4-bit parallel input and output data, In serial interface mode, this is the serial clock input pin ($\overline{\text{SCK}}$) for serial input and output data.

C/$\overline{\text{D}}$ (Command/Data)

This pin specifies whether the parallel or serial input is a command or data. Bring C/$\overline{\text{D}}$ high to input a command, and low to input data.

In parallel interface mode, the contents of C/$\overline{\text{D}}$ are latched at the rising edge of the second $\overline{\text{STB}}$. Perform any changes to the C/$\overline{\text{D}}$ input before the falling edge of the first $\overline{\text{STB}}$. When outputting data, hold C/$\overline{\text{D}}$ low, whether serial or parallel.

In serial interface mode, the contents of C/$\overline{\text{D}}$ are latched at the rising edge of the eighth $\overline{\text{SCK}}$.

The Schmitt-trigger input prevents noise errors.

$\overline{\text{BUSY}}$ (Busy)

This pin outputs a busy signal to the CPU to warn that the μPD7228/28A is internally busy. When this signal is low, the CPU cannot read/write the μPD7228/28A.

In the parallel interface mode, $\overline{\text{BUSY}}$ is forced low at the rising edge of the second $\overline{\text{STB}}$. In the serial interface mode, $\overline{\text{BUSY}}$ is forced low at the rising edge of the eighth $\overline{\text{SCK}}$.

If a chip is deselected ($\overline{\text{CS}}$ = high or chip address data does not match), the $\overline{\text{BUSY}}$ pin is placed in the high-impedance state.

SYNC (Synchronous)

In a multichip configuration, the SYNC signal synchronizes the phases of the LCD drive ac signals (row/column signal) among all the μPD7228/28A's within the frame period. It uses the row drive signal as a common signal.

If one chip is designated master, its SYNC pin is in output mode and the remaining chips are made slaves. Their SYNC pins are put in input mode. The SMM command selects input or output mode. The master chip outputs a SYNC pulse in the last cycle of each frame. The slave chip reads the SYNC pulse from its own SYNC input for synchronization with the master chip.

In a single-chip configuration, set the SYNC pin in the input or output mode. If you choose input mode, connect the SYNC pin to V_{SS}; conversely, if you choose output mode, the SYNC pin must be open.

Figures 1 and 2 show the output timing for the SYNC pulse in 8- and 16-line multiplexing.

Figure 1. SYNC Signal in 8-Line Multiplexing

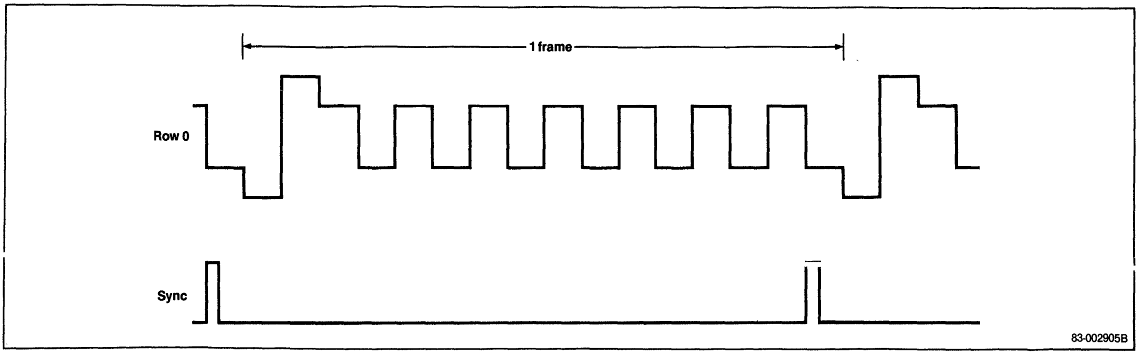
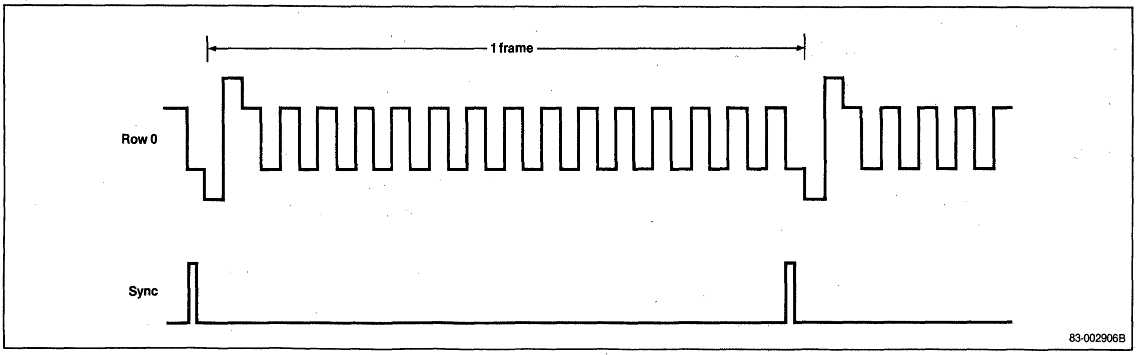


Figure 2. SYNC Signal in 16-Line Multiplexing



C_0-C_{41} (Column)

These pins output the column drive signals for the LCD.

C_{42}-C_{49}/R_{15}-R_8 (Column/Row)

These pins are column drive outputs (C_{42}-C_{49}, 50 x 8 mode) or row drive outputs (R_{15}-R_8, 42 x 16 mode), according to the SMM command.

R_{15}-R_8/R_7-R_0 (Row)

These pins are row drive outputs for rows R_{15}-R_8 or R_7-R_0, according to the SMM command.

V_{LC1}-V_{LC5} (LCD Drive Voltage Supply)

These are reference voltage input pins for determining the voltage level of the LCD column/row drive signals.

CLOCK (Clock)

This is the external clock input pin.

RESET (Reset)

This is the active-high reset signal input pin. It has priority over all operations. You can also use it to release standby mode and begin low power data retention.

V_{DD} (Power Supply)

This is a positive power supply pin.

V_{SS} (Ground)

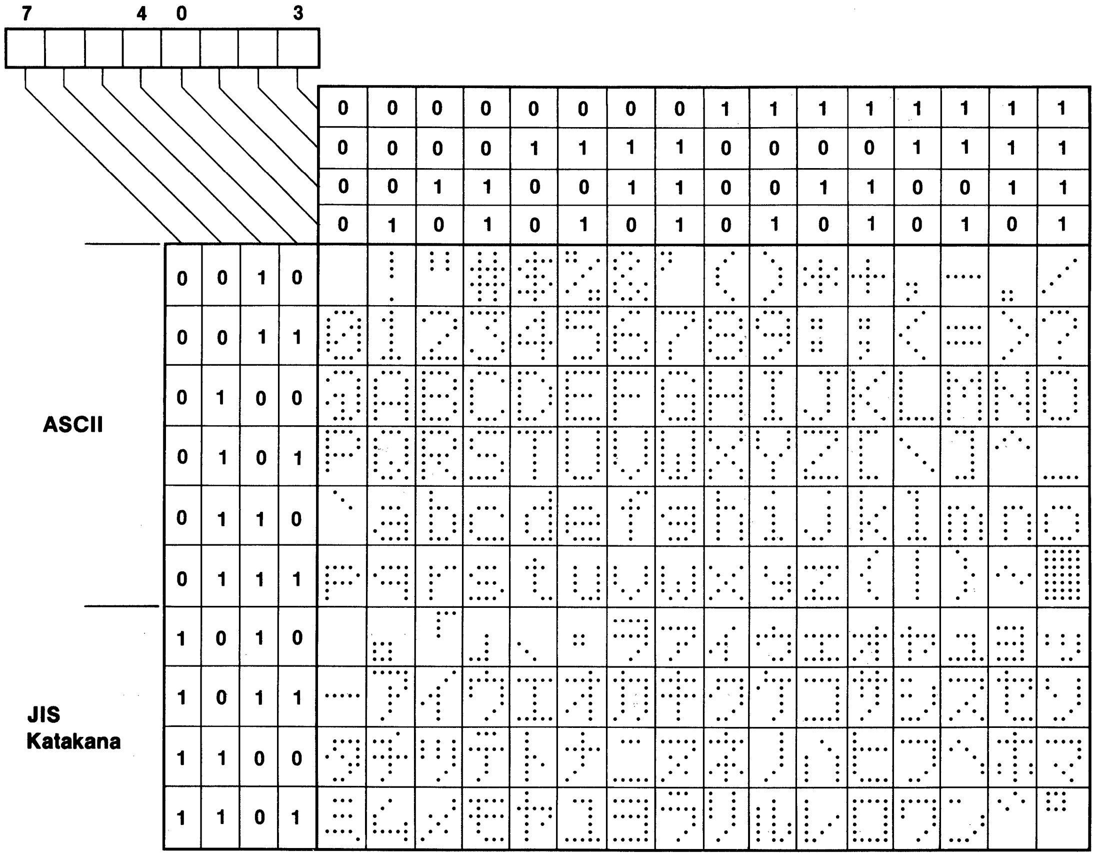
This is ground (GND).

COMMANDS FOR μPD7228/28A

The μPD7228/28A has 16 types of commands, each command consisting of one byte (8 bits).

Figure 3 shows the character codes and display patterns.

Figure 3. Character Codes and Display Patterns


7
4
0
3
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1
0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1
0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1
0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1
ASCII
0 0 1 0
0 0 1 1
0 1 0 0
0 1 0 1
0 1 1 0
0 1 1 1
JIS
Katakana
1 0 1 0
1 0 1 1
1 1 0 0
1 1 0 1

Notes:

(1) The character generator transfers 7-bit dot patterns five times to the five contiguous addresses of data memory.

(2) ASCII Characters, 96 (20H-7FH)

Upper case	26
Lower case	26
Numbers	10
Symbols	34

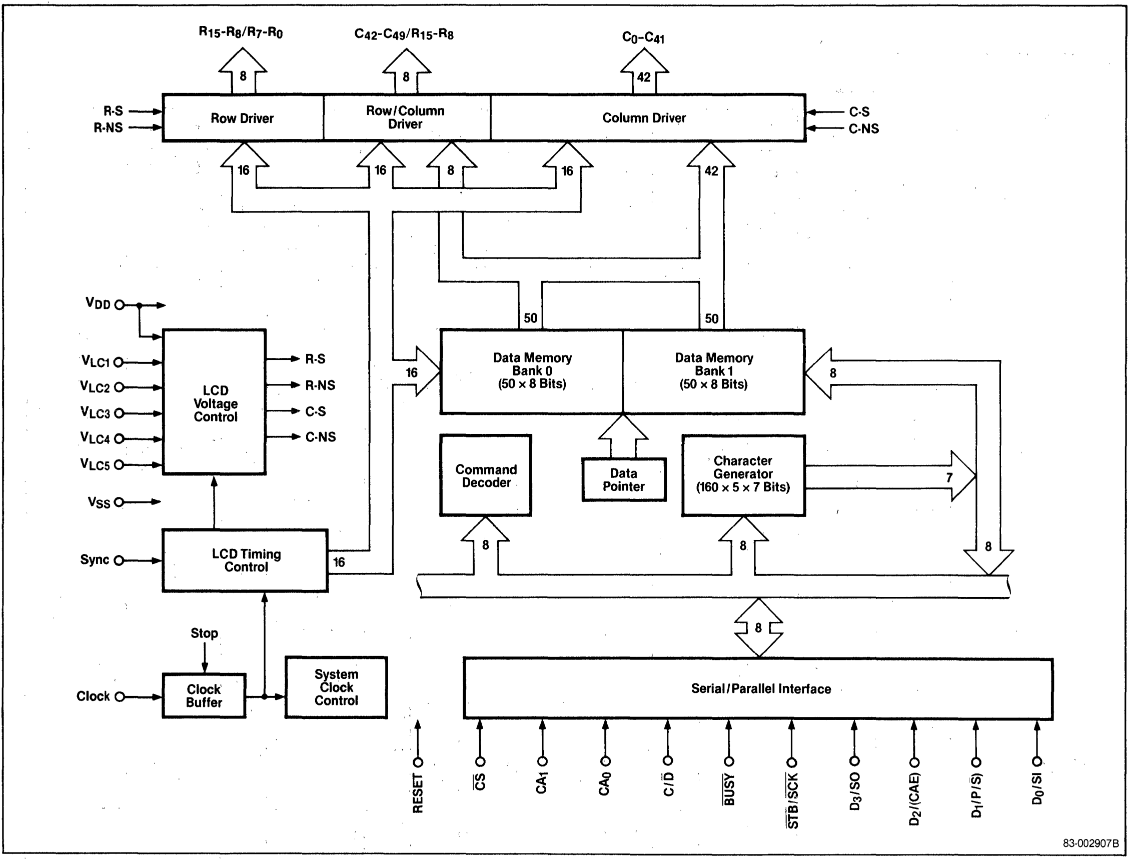
(3) JIS Characters, 64 (A0H-DFH)

Katakana	55
Symbols	9

(4) Because the character generator does not use bit 7 of data memory, dot R7 in 8 time-division mode and dots R7 and R15 in 16 time-division mode, corresponding to the most significant bits, can be used as cursor independent of the character generator. Use the cursor manipulation commands WRCURS and CLCURS.

83-002913B

Block Diagram



Absolute Maximum Ratings

T_A = 25°C

Supply voltage, V_{DD}	−0.3 V to +7 V
Input voltage, V_I	−0.3 V to V_{DD} +0.3 V
Output voltage, V_O	−0.3 V to V_{DD} +0.3 V
LCD operating voltage, V_{LCD} (7228A)	12.5 V
Operating temperature, T_{OPT}	
7228	−10 to +70°C
7228A	−40 to +85°C
Storage temperature, T_{STG}	-65 to +85°C

Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The device should be operated within the limits specified under DC and AC Characteristics.

Capacitance

T_A = 25°C; V_{DD} = 0 V; f = 1 MHz

Parameter	Symbol	Min	Typ	Max	Unit	Conditions
Input capacitance	C_I			10	pF	Return unmeasured pins to 0 V.
Output capacitance	C_O			25	pF	
I/O capacitance	C_{IO}			15	pF	

DC Characteristics

T_A = −10 to +70°C V_{DD} = +5 V ±10% (μPD7228); T_A = −40 to +85°C; V_{DD} = +5 V ±10% (μPD7228A)

		μPD7228			μPD7228A				
Parameter	Symbol	Min	Typ	Max	Min	Typ	Max	Unit	Conditions
Input voltage, high	V_{IH1}	0.7 V_{DD}		V_{DD}	0.7 V_{DD}		V_{DD}	V	Except $\overline{SCK}$
	V_{IH2}	0.8 V_{DD}		V_{DD}	0.8 V_{DD}		V_{DD}	V	$\overline{SCK}$
Input voltage, low	V_{IL}	0		0.3 V_{DD}	0		0.3 V_{DD}	V	
Output voltage, high	V_{OH1}	V_{DD} − 0.5			V_{DD} − 0.5			V	$\overline{BUSY}$, D_0-D_3; I_{OH} = −400 μA
	V_{OH2}	V_{DD} − 0.5			V_{DD} − 0.5			V	SYNC; I_{OH} = −100 μA
Output voltage, low	V_{OL1}			0.45			0.5	V	$\overline{BUSY}$, D_0-D_3; I_{OL} = 1.7 mA
	V_{OL2}			0.45			0.5	V	SYNC; I_{OL} = 100 μA
Input leakage current, high	I_{LIH}			10			10	μA	$V_I = V_{DD}$
Input leakage current, low	I_{LIL}			−10			−10	μA	V_I = 0 V
Output leakage current, high	I_{LOH}			10			10	μA	$V_O = V_{DD}$
Output leakage current, low	I_{LOL}			−10			−10	μA	V_I = 0 V
LCD operating voltage	V_{LCD}	3.0		V_{DD}	V_{DD}		12.5	V	
Row output impedance	R_{ROW}		4	8		6	16	kΩ	
Row/column output impedance	$R_{ROW/COL}$		5	10		7.5	20	kΩ	
Column output impedance	R_{COL}		10	15		15	30	kΩ	
Supply current	I_{DD1}		200	400		250	600	μA	Operating mode; f_C = 400 kHz
	I_{DD2}			20			25	μA	Stop mode; CLK = 0 V

AC Characteristics

T_A = −10 to +70°C; V_{DD} = +5 V ±10% (μPD7228); T_A = −40 to +85°C; V_{DD} = +5 V ±10% (μPD7228A)

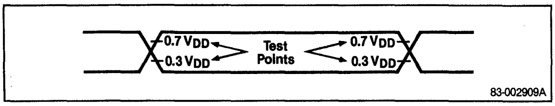
		μPD7228			μPD7228A				
Parameter	Symbol	Min	Typ	Max	Min	Typ	Max	Unit	Conditions
Common Operation									
Clock frequency	f_C	100		1100	100		1100	kHz	
Clock pulse width, high	t_{WHC}	350			350			ns	
Clock pulse width, low	t_{WLC}	350			350			ns	
$\overline{RESET}$ pulse width, high	t_{HRS}	4			4			μs	
$\overline{BUSY}$ delay time from $\overline{CS}$ ↓	t_{DCSB}			2			3	μs	C_L = 50 pF
$\overline{CS}$ ↑ delay time to $\overline{BUSY}$ floating	t_{DCSBF}			4			5	μs	C_L = 50 pF
$\overline{CS}$ high-level time	t_{WHCS}	4			4			μs	
SYNC load capacitance	C_{LSY}			100			100	pF	
Data setup time to $\overline{RESET}$ ↓	t_{SDR}	0			5			μs	
Data hold time from RESET ↓	t_{HRD}	4			5			μs	
Serial Interface Operation									
$\overline{SCK}$ cycle	t_{CYK}	0.9			0.9			μs	
$\overline{SCK}$ pulse width, high	t_{WHK}	400			400			ns	
$\overline{SCK}$ pulse width, low	t_{WLK}	400			400			ns	
$\overline{SCK}$ hold time from $\overline{BUSY}$ ↑	t_{HBK}	0			0			ns	
SI setup time to $\overline{SCK}$ ↑	t_{SIK}	100			120			ns	

AC Characteristics (cont)

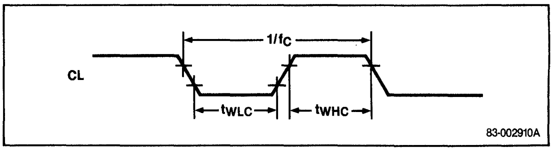
		µPD7228			µPD7228A				
Parameter	**Symbol**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
SI hold time from $\overline{\text{SCK}}$ ↑	t_{HKI}	250			270			ns	
SO delay time from $\overline{\text{SCK}}$ ↓	t_{DKO}			320			350	ns	C_L = 50 pF
$\overline{\text{BUSY}}$ delay time from eighth $\overline{\text{SCK}}$ ↑	t_{DKB}			3			4	µs	
$\overline{\text{BUSY}}$ low-level time	t_{WLB}	18		64	18		64	$1/f_C$	
C/$\overline{\text{D}}$ setup time to first $\overline{\text{SCK}}$ ↓	t_{SDK}	0			0			µs	
C/$\overline{\text{D}}$ hold time from eighth $\overline{\text{SCK}}$ ↑	t_{HKD}	2			3			µs	
$\overline{\text{CS}}$ hold time from eighth $\overline{\text{SCK}}$ ↑	t_{HKCS}	2			5			µs	
Parallel Interface Operation									
Input command setup time to $\overline{\text{STB}}$ ↓	t_A	100			120			ns	C_L = 80 pF
Input command hold time from $\overline{\text{STB}}$ ↓	t_B	90			110			ns	C_L = 20 pF
Input data setup time to $\overline{\text{STB}}$ ↑	t_C	230			250			ns	C_L = 80 pF
Input data hold time from $\overline{\text{STB}}$ ↑	t_D	50			70			ns	C_L = 20 pF
Output data delay time	t_{ACC}	90		650	90		750	ns	C_L = 80 pF
Output data hold time	t_H	0		150	0		150	ns	C_L = 20 pF
$\overline{\text{STB}}$ pulse width low	t_{SL}	700			700			ns	
$\overline{\text{STB}}$ high-level time	t_{SH}	1			1			µs	
$\overline{\text{STB}}$ hold time from $\overline{\text{BUSY}}$ ↑	t_{HBS}	0			0			µs	
$\overline{\text{BUSY}}$ delay time from second $\overline{\text{STB}}$ ↑	t_{DSB}			3			4	µs	
C/$\overline{\text{D}}$ setup time to first $\overline{\text{STB}}$ ↓	t_{SDS}	0			0			µs	
C/$\overline{\text{D}}$ hold time from second $\overline{\text{STB}}$ ↑	t_{HSD}	2			3			µs	
$\overline{\text{CS}}$ hold time from second $\overline{\text{STB}}$ ↑	t_{HSCS}	2			3			µs	

Timing Waveforms

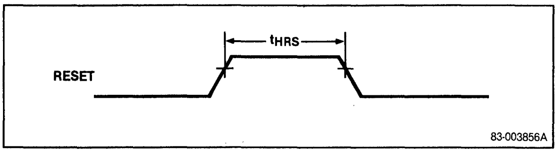
AC Timing Test Points



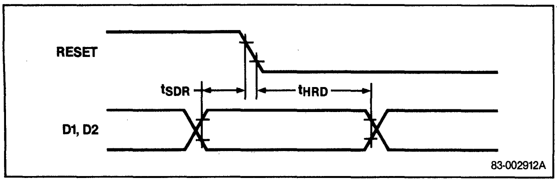
Clock Waveform



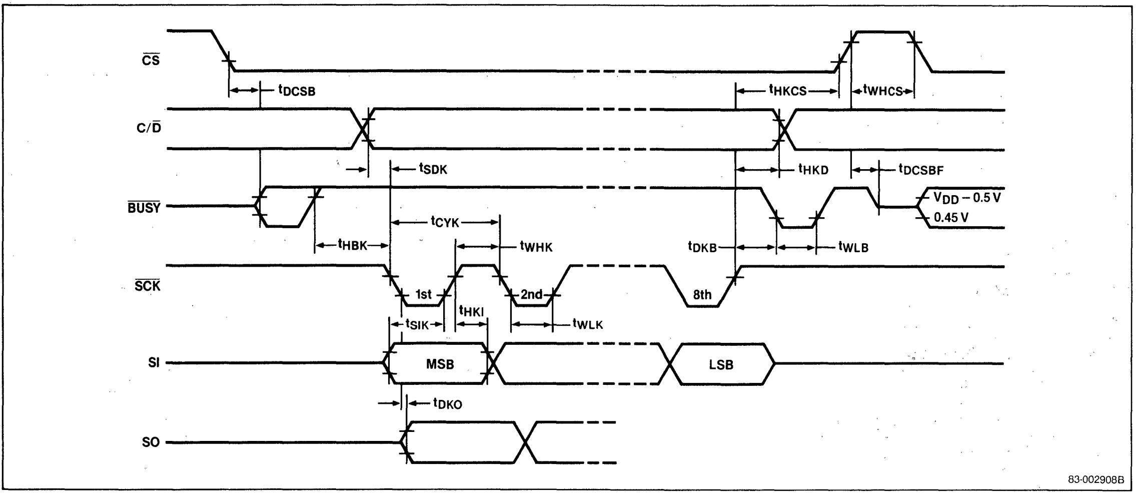
Reset Signal



Interface

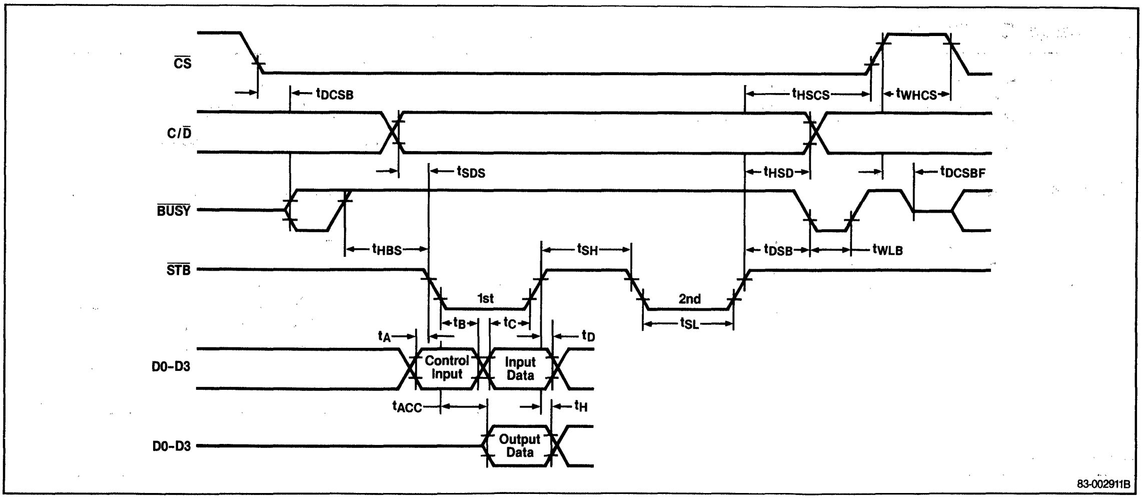


Serial Interface



8

Parallel Interface



Command Summary

Mnemonic	Operation	Instruction Code								Hex Code
SFF	Set frame frequency	0	0	0	1	0	F_2	F_1	F_0	10H-14H
SMM	Set multiplexing mode	0	0	0	1	1	M_2	M_1	M_0	18H-1FH
DISP OFF	Display off	0	0	0	0	1	0	0	0	08H
DISP ON	Display on	0	0	0	0	1	0	0	1	09H
LDPI	Load data pointer with immediate	1	D_6	D_5	D_4	D_3	D_2	D_1	D_0	80H-B1H, C0H-F1H
SRM	Set read mode	0	1	1	0	0	0	I_1	I_0	60H-63H
SWM	Set write mode	0	1	1	0	0	1	I_1	I_0	64H-67H
SORM	Set OR mode	0	1	1	0	1	0	I_1	I_0	68H-6BH
SANDM	Set AND mode	0	1	1	0	1	1	I_1	I_0	6CH-6FH
SCML	Set character mode with left entry	0	1	1	1	0	0	0	1	71H
SCMR	Set character mode with right entry	0	1	1	1	0	0	1	0	72H
BRESET	Bit reset	0	0	1	B_2	B_1	B_0	J_1	J_0	20H-3FH
BSET	Bit set	0	1	0	B_2	B_1	B_0	J_1	J_0	40H-5FH
CLCURS	Clear cursor	0	1	1	1	1	1	0	0	7CH
WRCURS	Write cursor	0	1	1	1	1	1	0	1	7DH
STOP	Set stop mode	0	0	0	0	0	0	0	1	01H

B_2-B_0 Specifies a data memory bit

D_6-D_0 Immediate data

F_2-F_0 Specifies frame frequency as a submultiple of clock frequency

I_1-I_0 Specifies modification of data pointer contents after byte data is processed

J_1-J_0 Specifies modification of data pointer contents after bit is set or reset

M_2-M_0 Specifies data memory bank, number of rows, functions of row/column drivers, and SYNC pin mode

NEC

Section 9
Development Tools

4-Bit; μPD7500 Series

4-Bit; μPD75000 Series

8-Bit; μPD7800 Series

8-Bit; μPD78K2 Series

8-Bit; μPD78K2 Series (cont)

8/16-Bit; μPD78K3 Series

NEC
NEC Electronics Inc.

EVAKIT-7500B for the μPD7500 Series

Description

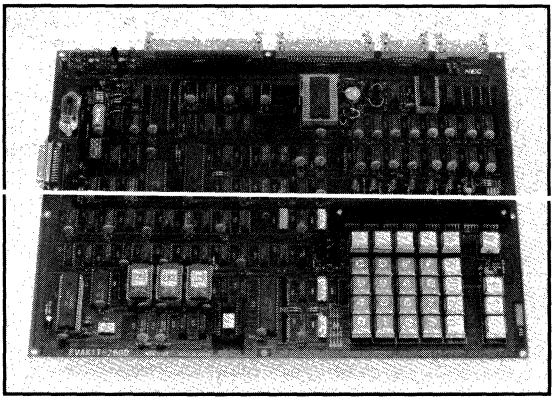
The EVAKIT-7500B is a stand-alone EVAKIT for NEC's μPD7500 series of four-bit, single-chip microcomputers. The EVAKIT-7500B provides complete hardware emulation and software debug capabilities for the μPD7507 and μPD7508 microcomputers. With the addition of device specific add-on boards, the EVAKIT-7500B is easily tailored to support the remaining members of the family.

Real-time and single-step emulation capability, together with a powerful on-board system monitor and real-time trace capability, create a powerful debug environment. The EVAKIT-7500B is controlled either from an on-board keypad or over a serial line from a terminal or host computer. User programs are downloaded through a serial line or read from a PROM. Existing programs can be modified or small programs can be created using the on-board hexadecimal keypad.

A host controller program for an IBM PC® series or compatible computer is provided with each EVAKIT-7500B. This program provides the following additional capabilities: complete EVAKIT-7500B control from the host console, program upload/download, line assembly, host system directory display and symbolic debugging.

IBM PC is a registered trademark of International Business Machines Corporation.

EVAKIT-7500B

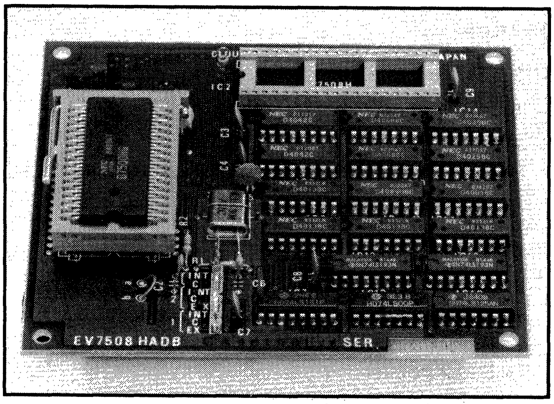


Features

- □ Real-time and single-step emulation capability
- □ 8K bytes of user program memory
- □ Powerful system monitor
  - — Display/modify/move program memory
  - — Display/modify data memory
  - — Load/verify/display PROM
  - — Examine/modify internal registers
  - — Full disassembler
- □ User-specified breakpoint conditions
  - — Program counter and number of passes
  - — Stack pointer
  - — Data address and value
- □ Real-time trace capability
  - — 2048 instruction cycle trace
  - — External trace probes
- □ Supports three operating modes
  - — On-board hexadecimal keypad controlled
  - — External terminal controlled
  - — Host computer system controlled
- □ Serial interface: RS-232C or TTL
- □ EPROM programming capability (2764 and 27128)
- □ Host Control Software for IBM PC Series or compatible

EVAKIT-7500B ADD-ON BOARDS

EV7508H

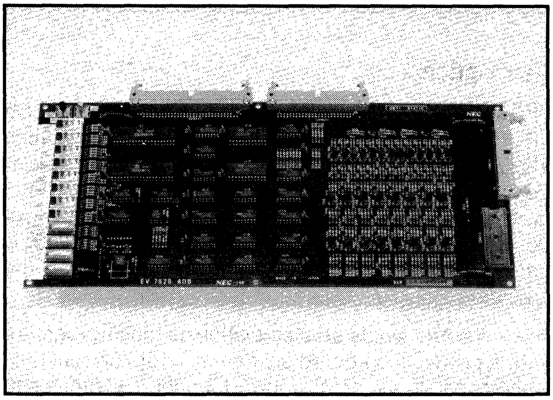


The EV7508H is an add-on board for the EVAKIT-7500B which is required for emulating the μPD7507H and the μPD7508H microcomputers. This board plugs directly into the μPD7500 socket on the EVAKIT-7500B, allowing the system to support these high speed versions of the μPD7500 series.

EV7514

The EV7514 is an add-on board for the EVAKIT-7500B required for emulating the μPD7502 and μPD7503 microcomputers. This board is mounted under the EVAKIT-7500B, adding LCD controller/driver capability to the EVAKIT.

EV7528



The EV7528 is an add-on board for the EVAKIT-7500B required for emulating the μPD7527A, μPD7528A, μPD7537A, and μPD7538A microcomputers. This board is mounted under the EVAKIT-7500B, allowing the EVAKIT to support the additional features of these parts: I/O ports with high dielectric strength, optional pull-down resistors, and zero voltage detection circuits.

EV7533

The EV7533 is an add-on board for the EVAKIT-7500B required for emulating the μPD7533 microcomputer. This board plugs directly into the μPD7500 socket, allowing the EVAKIT to emulate the μPD7533's four analog inputs and its 8-bit A/D converter.

EV7554A

The EV7554A is an add-on board for the EVAKIT-7500B required for emulating the μPD7554/54A, μPD7556/56A, μPD7564/64A, and μPD7566/66A microcomputers. This board mounts on top of EVAKIT-7500B, allowing the EVAKIT to emulate the additional features of these parts: optional pull-up/pull-down resistors for ports 0, 1, 10, and 11; comparator/CMOS inputs for port 1; high current/CMOS outputs for ports 8, 9, 10, and 11.

μPD7500 SERIES SYSTEM EVALUATION BOARDS

SE-7514A

The SE-7514A is the system evaluation board for the μPD7500 series microcomputers with LCD direct drive capabilities: μPD7502 and μPD7503. The SE-7514A is functionally equivalent to the ROM-based microcomputers. With the user's program housed in either an on-board μPD2764 or μPD27128, you can connect the SE-7514A to your prototype and evaluate total system performance.

SE-7554A

The SE-7554A is the system evaluation board for the μPD7500 series mini/microcomputers: μPD7554/54A, μPD7556/56A, μPD7564/64A, and μPD7566/66A. The SE-7554A is functionally equivalent to the ROM-based mini-microcomputer. It can be set up to emulate any of the available mask options. With your program residing in the lower 4K bytes of an on-board μPD2754, you can connect the SE-7554A to your prototype and evaluate total system performance.

NEC
NEC Electronics Inc.

ASM75
Absolute Assembler for the μPD7500 Series

Description

The μPD7500 series absolute assembler (ASM75) converts symbolic source code for the entire μPD7500 series microcomputer family into executable absolute address object code. The assembler verifies that each instruction assembled is valid for the target microcomputer specified at assembly time. An object code file is produced in ASCII hexadecimal format and may be down loaded to a PROM programmer or hardware debugger.

Features

- □ Absolute address object code output
- □ Macro definition capability
- □ Generic jump with optimization capability
- □ Conditional assembly options
  — Up to eight levels of nesting
- □ User-selectable and directable output files
- □ Runs under the MS-DOS® operating system

MS-DOS is a registered trademark of Microsoft Corporation.

Ordering Information

Part Number	System	Description
ASM75-D52	MS-DOS	5-1/4" double-density floppy diskette

NEC
NEC Electronics Inc.

RA75X Relocatable Assembler Package for the μPD75000 Series

Description

The RA75X relocatable assembler package converts symbolic source code for the μPD75000 series of microcomputers into executable absolute address object code. The package consists of six separate programs: assembler (RA75X), linker (LK75X), hexadecimal format object converter (OC75X), librarian (LB75X), list converter (LCNV75X), and macroprocessor (MP).

RA75X translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcomputer specified at assembly time.

LK75X combines relocatable object modules and absolute load modules and converts them into an absolute load module. OC75X converts an absolute object module or an absolute load module to an ASCII hexadecimal format object file. LB75X allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included as input to the linker, the linker extracts only those modules required to resolve external references from the library file and relocates and links them into the absolute load module.

LCNV75X allows relocatable list files to be converted into absolute list files. MP expands macros contained in a source program prior to assembling.

Features

- □ Absolute address object code output
- □ Generic branch capability and optimization
- □ User-selectable and directable output files
- □ Extensive error reporting
- □ Macro capabilities
- □ Runs under MS-DOS® and VAX®/VMS® operating systems

Ordering Information

Part Number	System	Description
RA75X-D52	MS-DOS	5-1/4" double-density floppy diskette
RA75X-VVT1	VAX/VMS	9-track 1600 BPI magnetic tape

MS-DOS is a registered trademark of Microsoft Corporation.
VAX and VMS are registered trademarks of Digital Equipment Corporation.
IBM PC, PC/XT, and PC/AT are registered trademarks of International Business Machines Corporation.

Program Syntax

An RA75X source module consists of a series of code and data segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, and comment.

The symbol field may contain a label, whose value is the instruction or data address, or a name, which represents an instruction address, data address, or a constant. The mnemonic field may contain an instruction or an assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, –, *, /, NOT, AND, OR, XOR, EQ or =, NE or < >, GT or >, GE or > =, LT or <, LE or < =, SHR, SHL, MOD, .(bit position), the + sign, and the – sign.

Assembler Directives

Assembler directives give instructions to the assembler. They are not translated into machine code during assembly. Basic assembler directives include: storage definition (DB, DW, DS, STKLN), symbol definition (EQU, SET), and program boundary definition (ORG, END). Program linkage directives are provided to NAME the module and to declare symbols as PUBLIC or external (EXTRN).

Segment definition directives define whether a segment is a code segment (CSEG) allocated to ROM, or a data segment (DSEG) allocated to RAM. The relocation attributes for each segment directive are specified in its operand. These attributes, which specify how the various segments are to be linked, include INBLOCK, XBLOCK, SENT, IENT, PAGE, and AT.

The VENTn directive defines the status of the memory bank enable flag (MBE), the register bank enable flag (RBE), and the code entry address for the interrupt vectors. The TCALL/TBR directives create a table that allows the CALL and BR instructions to function for the GETI instruction.

9

The μPD75000 series instruction set contains three branch instructions with varying legal address ranges. To avoid calculating which branch instruction to use, you can substitute the BR (Branch) directive for any BR $addr (one-byte branch), BRCB !caddr (two-byte branch), or BR !addr (three-byte branch) instruction in your source program. During assembly, a suitable branch instruction is chosen for each BR directive.

Assembler Controls

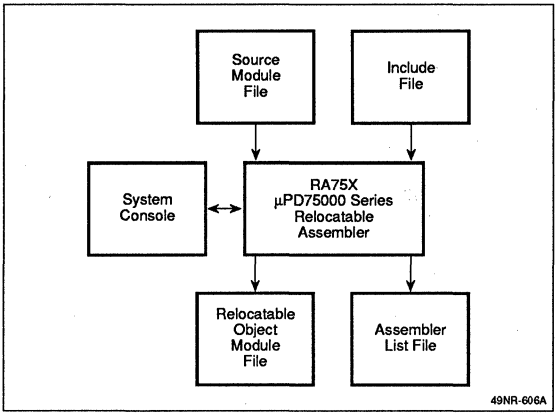
The RA75X assembler controls can be specified in a variety of ways. Depending upon the particular control, the controls can be specified directly in the assembler command line, in a parameter file invoked in the command line, at the beginning of the source module, or anywhere in the source program. The RA75X assembler controls include the following:

- Target microcomputer specification
- Output file selection and destination
- Listing format controls
- Date specification
- Generation/suppression of listing
- Title specification
- Inclusion of other source files (in source program only)
- Page eject (in source program only)

The listing file contains the complete assembly listing or only lines with errors, and a symbol table or cross reference table. The symbol table shows all defined symbols in alphabetical order, their types, attributes, and the values initially assigned to them. The cross-reference table contains all defined symbols and the numbers of all statements that refer to them.

The object file contains the relocatable object module. It is in a NEC proprietary relocatable object module format. The object file may also contain local symbol information for the symbolic debugger. Figure 1 is the relocatable assembler functional diagram.

Figure 1. RA75X Assembler Functional Diagram

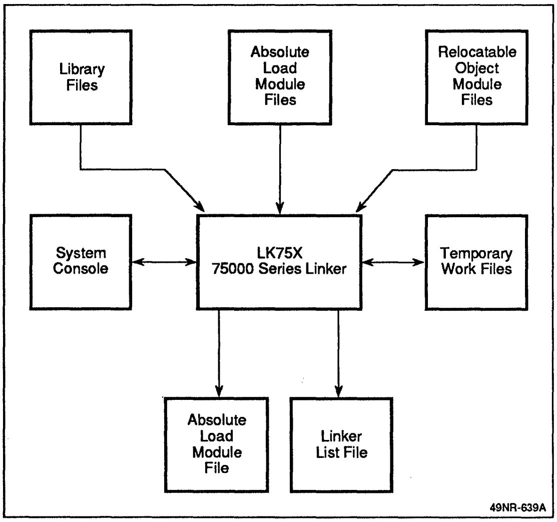

Source Module File
Include File
System Console
RA75X μPD75000 Series Relocatable Assembler
Relocatable Object Module File
Assembler List File
49NR-606A

Linker

The linker combines several relocatable object modules or absolute load modules, resolving PUBLIC/EXTRN references between modules, to create an absolute load module. This load module contains both absolute object code and symbol information. The linker can search library files for required modules to resolve external references.

The linker controls for LK75X can be specified in either the command line or a parameter file. The programmer can specify the date, the module name, the stack size and starting address, an inhibited area in ROM space, the starting address and order for relocatable code segments, and whether segments are linked sequentially as input or randomly in the most effective manner. The programmer can also specify that a list file containing a link map, a local symbol table, or a public symbol table be created. Figure 2 is the linker functional diagram.

Figure 2. LK75X Linker Functional Diagram


Library Files
Absolute Load Module Files
Relocatable Object Module Files
System Console
LK75X 75000 Series Linker
Temporary Work Files
Absolute Load Module File
Linker List File
49NR-639A

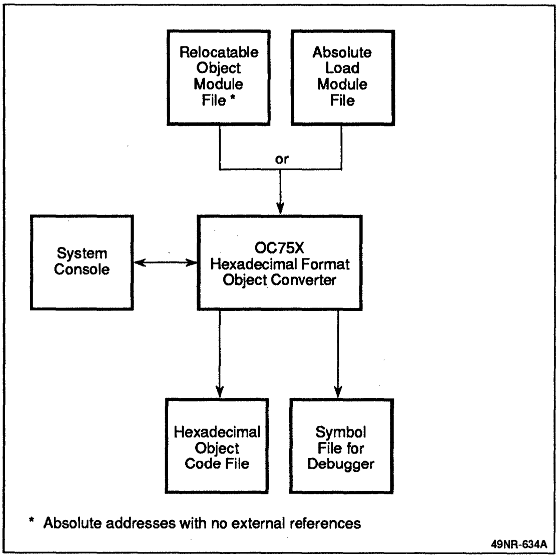
Hexadecimal Format Object Converter

The OC75X object converter outputs the object code file in ASCII hexadecimal format, which can be downloaded to a PROM programmer or hardware debugger. The object converter controls for OC75X can be specified in the command line or a parameter file. The programmer can specify whether or not to generate a symbol file for a hardware debugger and whether the addresses of the hex code should be sorted in numerical order or left as ordered in the source program. Figure 3 is the functional diagram of the hexadecimal format object code converter.

Librarian

The LB75X librarian creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by allowing several modules to be kept in a single file, and provides an easy wav to link freauentlv used modules into programs. Modules can be added to or deleted from a library file, or the contents of the library file can be listed.

Figure 3. OC75X Hexadecimal Format Object Code — Converter Functional Diagram


Relocatable Object Module File *
Absolute Load Module File
or
System Console
OC75X Hexadecimal Format Object Converter
Hexadecimal Object Code File
Symbol File for Debugger
49NR-634A

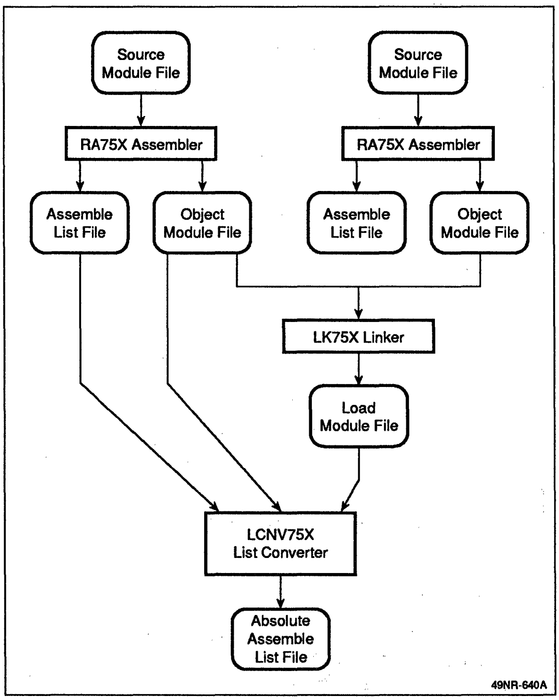
* Absolute addresses with no external references

List Converter

Normally, listing files produced by a relocatable assembler do not show the final absolute address for instructions, because their location is not decided until link time. The address shown in the listing is only the offset from the start of the code or data segment.

The LCNV75X list converter uses the assembly list and object module files from the assembler and the linkers load module file to create an absolute address assembly listing. This absolute listing shows the addresses of instructions as their final absolute address in memory. It is useful for debugging and documentating the assembled program. The programmer can specify the load module (–L), assembly list (–A), and output assembly (–O) file names. Figure 4 is the functional diagram of the list converter.

Figure 4. LCN75X List Converter Functional Diagram



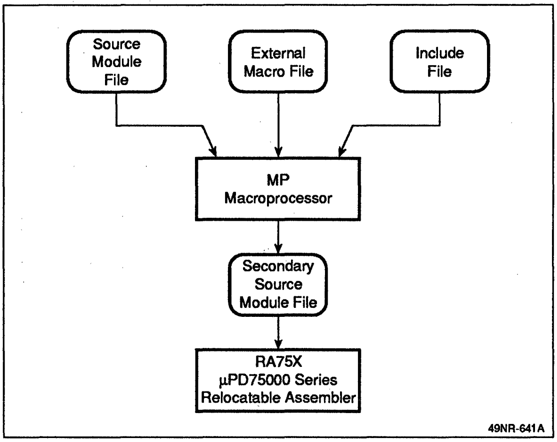
Macroprocessor

The macroprocessor interprets the macros described in a source program and expands them to create another source program. This can be input to the assembler. It has the following three main functions:

- Expands macros by defining and referencing them
- Reads and expands include files
- Selects assembler source based on a conditional macro instruction

Figure 5 is the functional diagram of the macroprocessor.

Figure 5. MP Macroprocessor Functional Diagram



Operating Environment

The RA75X package can run under a variety of operating systems. A version is available for an MS-DOS system with one or more disk drives and at least 128K of system memory. Another version is available to run on a Digital Equipment Corporation VAX Computer system under a VMS (Version 4.1 or later) operating system.

Emulator Controller Program

Absolute hex-format object module files produced by the RA75X relocatable assembler package can be debugged using an NEC EVAKIT-75X stand-alone emulator. The EVAKIT-75X controller program EC75X, allows the programmer to communicate with the emulator through an RS-232C serial line. EC75X is available to run on the IBM PC®, PC/XT®, and PC/AT® under MS-DOS, and is included with the MS-DOS version of the RA75X package at no extra charge.

The EC75X controller program provides the following features:

- Uploading/downloading of hexadecimal object files and symbol table
- Symbolic debugging
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory display
- Disk storage of debug session

License Agreement

RA75X is sold under terms of a license agreement, which is included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users.

Documentation

For more information on source program formats, assembler operation, and actual program examples NEC Electronics Inc. provides the following documentation:

- RA75X μPD75000 Series Relocatable Assembler Package, Language Manual (MS-DOS)
- RA75X μPD75000 Series Relocatable Assembler Package, Operation Manual (VMS)
- MP Macroprocessor, User's Manual

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

ST75X Structured Assembler Preprocessor for the µPD75000 Series

Description

The ST75X structured assembler preprocessor is a companion program to the RA75X relocatable assembler for the NEC µPD75000 series of microcomputers. ST75X converts a source code file containing structured assembly statements into a pure assembly language source file, which can then be assembled with RA75X.

ST75X converts a structured assembly statement into one or more µPD75000 assembly language instructions that perform the desired operation. Since ST75X converts only structured statements and does not convert µPD75000 assembly language instructions, a structured source program can include a combination of µPD75000 structured statements and assembly language.

ST75X enables the assembly language programmer to use some of the structures and syntax of higher-level languages, such as the C language. This improves program readability and reliability, and increases programmer productivity.

Features

- □ Control structures for conditions, looping, and switch-case
- □ Preprocessor directives for conditional code generation
- □ C-like representation of comparison operations
- □ C-like representation of assignment/arithmetic operations
- □ Increment and decrement operators
- □ Uses all µPD75000 mnemonics, registers, and features
- □ Runs under MS-DOS® and VAX®/VMS® operating systems

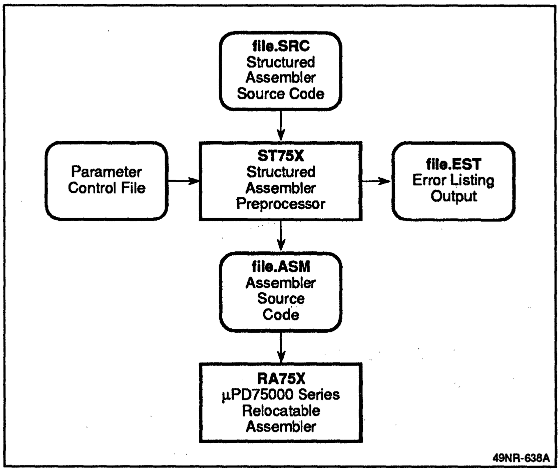
Ordering Information

The ST75X structured assembler preprocessor is provided in the following software package at no cost:

RA75X Relocatable Assembler Package for µPD75000 Series Microcomputers.

MS-DOS is a registered trademark of Microsoft Corporation
VAX and VMS are registered trademarks of Digital Equipment Corporation.

Structured Assembler Preprocessor Functional Diagram



A Summary of Structured Language

A line of source code for the ST75X contains either a structured assembly statement or a µPD75000 assembly language statement. µPD75000 assembly language statements (µPD75000 instructions, RA75X directives, or RA75X controls) pass through ST75X without change.

Structured assembly statements consist of preprocessor directives, assignment statements, and control statements. These statements are entered one per line, and are terminated at the end of a line. An optional comment may follow a semicolon at the end of the statement; all text following a semicolon is ignored by ST75X.

Preprocessor directives cause ST75X to include or omit portions of code. Assignment statements generate one or more µPD75000 assembly language instructions to alter the contents of a register or variable. Control statements aenerate the necessary instructions to test conditions and change control flow based on those conditions.

Preprocessor Directives

ST75X preprocessor directives set and test variables, allowing conditional processing of code; include external files; and map instructions to µPD75000 GETI table reference instructions. Table 1 lists the preprocessor directives and their functions.

Table 1. Preprocessor Directives and Functions

Directive	Function
#define NAME value	Defines the variable NAME, set to the supplied value.
#ifdef ABC <statements> #else <statements> #endif	If ABC is defined as above, or on the command line with the –D option, the first set of statements is processed and the second set is ignored; if ABC was not defined, or defined as zero, the first set of statements is ignored and the second set is processed.
#include "filename"	The named file is read from disk and processed as if included in the source.
#defgeti getiname <instructions> #endgeti	The listed instructions are assigned to getiname. When those instructions are found in the source, they are replaced by a "GETI getiname" instruction.

Assignment, Increment, and Decrement Statements

ST75X provides the ability to represent an assignment, or an assignment with an arithmetic operation, in the C language syntax:

destination <assign-op> source

The assignment operators allow either simple assignment, or the combination of an assignment with an arithmetic operation on the source and destination.

Examples:

```
A = B ; Move contents of B register to A

A += @HL ; Add contents of memory at HL to A,
         ; store in A
```

Where an assignment requires an intermediate register to hold the value being assigned, the register is designated by naming it in parentheses following the assignment operation.

Examples:

```
DATA1 = B (A) ; Store contents of B into memory at
              ; DATA1, using A as temporary
              ; storage

BC &= HL (XA) ; AND BC with HL, store in BC, use
              ; XA as temp
```

The increment and decrement operators (+ + and – –) operate on a single operand.

Table 2 lists the assignment operators with examples and functions.

Table 2. Assignment Operators with Examples and Functions

Operator	Example	Function
=	A = B	A ← B
<->	A <-> B	Contents of A and B are exchanged
+=	A += B	A ← A + B
–=	A –= B	A ← A – B
&=	A &= B	A ← A & B (logical AND)
|=	A |= B	A ← A | B (logical OR)
^=	A ^= B	A ← A ^ B (logical XOR)
++	A++	A ← A + 1
– –	A – –	A ← A – 1

Control Statements

Control statements allow conditions to be tested. Based on the results of the test, blocks of code can be executed or skipped. Reserved words in the control statement define the start and end of blocks of code, and expressions to be evaluated.

Example:

```
if ( A = = @HL )      The condition is tested.
  PORT5 = B (A)       If A equals the content of
  A = @HL             memory at HL, this code is
                      executed.
else
  A + = @HL           Otherwise, this code is
  A – = B             executed.
  PORT5 = A
endif
```

Table 3 shows the directives used within the control statements.

Table 3. Control Statement Directives

Directive	Function
if – elseif – else – endif	Test variable expressions
if_bit – elseif_bit – else – endif	Test bit expressions
switch – case – default – ends	Select based on variable
for – next	Loop, test variable
while – endw	
repeat – until	
while_bit – endw	Loop, test bit
repeat – until_bit	
break	Exit control block
continue	Skip to top of block
goto LABEL	Branch to label

Variable and Bit Expressions

Variable expressions for tests consist of a single value, a comparison between two variables, or a logical combination of comparisons. Bit expressions test individual bits. Table 4 shows examples of comparisons.

Table 4. Examples of Variable Expression Comparisons

Comparison	Meaning
if (A)	True if A is non-zero
if (A < B)	True if A is less than B
if ((A < B) && (A > C))	True if A is less than B and greater than C
if_bit (PORT1.2)	True if bit 2 of PORT1 is 1

The allowable expressions using variables are shown in table 5.

Table 5. Expressions and Examples

Expression	Example
Primary	(A)
Term	(A < = B)
Term && Term	((A<B) && (A>C))
Term || Term	((A= =C) || (A= =B))

A primary value for a variable expression is a register name or defined symbol. A term consists of two primary values compared with a binary operator. Table 6 lists the supported binary operators and their meanings.

Table 6. Binary Operators

Binary Operator	Meaning
= =	Equals
!=	Not equal
>	Greater than
← > =	Greater than or equal to
<	Less than
< =	Less than or equal to

Bit expressions test individual bits of registers, ports, or memory locations. Table 7 shows the allowable forms of bit expressions.

Table 7. Bit Expressions

Bit Expression	Example
Bit_primary	(PORT0.2)
!Bit_primary	(!CY)
Bit_primary && Bit_primary	(A.0 && CY)
Bit_primary || Bit_primary	(PORT0.2 || CY)

A Bit_primary can be either a reserved word bit identifier, such as a bit of a register or port (PORT0.1, CY), or a bit definition symbol (SB0 EQU PORT0.2).

ST75X Operation and Controls

ST75X is invoked by specifying the name of the source file, followed by optional controls.

Example:

```
  C>ST75X ABC.SRC -DXYZ=3
```

ST75X reads the specified source file and produces an output assembly language file, which can be input to RA75X. The output file contains all lines provided in the input source file, plus those generated by ST75X. Lines containing no statements for the structured assembler are passed through unchanged. Lines with structured assembly statements are placed in the output preceded by a semicolon. RA75X treats these lines as comments. These lines are then followed by the code generated by ST75X.

The controls for ST75X are specified in the preprocessor command line or in a parameter file invoked in the command line. Table 8 lists the ST75X preprocessor controls and functions.

Table 8. ST75X Preprocessor Controls

Control	Function
–Ofilename	Specify name of output assembly source file
–Ffilename	Specify name of parameter file to be read
–Efilename	Specify name of error listing file
–Dsymbol[=value]	Define symbol (like #define in code)
–I[d:][directory]	Define path for include file
–WTn1,n2,n3	Define TAB settings for generated code

9

The –O option allows the name of the output file to be specified. If not specified, the output file name defaults to the name of the input source file with the extension .ASM.

The –F option allows a parameter file to be specified, which will be read by ST75X. This parameter file can contain a list of controls to be given to ST75X, instead of or in addition to those specified on the command line.

The –E option specifies the name of the error listing file. The error file contains the file name, error number, description of error, and the line containing the error. If the –E option is not specified, the error file name defaults to the name of the input source file with the extension .EST.

The –D control allows a symbol to be defined on the command line, with an optional value provided. If a symbol is defined but no value specified, the value defaults to 1. If the source file contains a #define directive, which specifies a variable with the same name as the –D control, the value on the command line will override the value in the #define directive.

The –I specifies a drive or directory other than the current drive and directory to search for include files.

The –WT control specifies the number of TAB characters to insert before labels, instruction mnemonics, and instruction operands generated by ST75X. This allows clear separation of assembly language instructions coded in the source file from those generated by ST75X.

NEC

NEC Electronics Inc.

DDK-78C10 Evaluation Board for the μPD78CXX Series

Description

The DDK-78C10 is an evaluation board for the NEC μPD78CXX series of 8-bit single-chip microcomputers. The DDK-78C10 is designed to provide maximum flexibility when evaluating and designing with the μPD78CXX series. Prominent features of the DDK-78C10 are 8K bytes of ROM, 8K bytes of RAM, an RS-232C communication port, and a powerful monitor program. The DDK-78C10 board is supplied on an IBM PC® compatible card and includes a playpen area for building your application specific hardware.

A copy of RA87, the μPD7800 series relocatable assembler for use on an IBM PC, PC/XT®, PC AT®, or compatible host computer, is shipped with each DDK-78C10 to allow code to be developed for evaluation purposes. Also included with the DDK-78C10 is an emulator controller program for the IBM PC, the source code for the monitor, and a complete set of documentation. This total package provides you with a fast, efficient way of evaluating the capabilities of the μPD78CXX series for your application.

Features

□ μPD78CXX series evaluation board with power supply

□ On-board memory
  — ROM: 8K bytes
  — RAM: 8K bytes

□ Powerful on-board debug monitor
  — Real-time operation
  — Display/change/fill/move memory
  — One software breakpoint
  — User program download capability
  — Input from ports A, C, and port B (bits 2-7)
  — Output to ports A, C, and port B (bits 2-7)
  — Repeat the previous command

□ RS-232C serial interface for terminal or host computer

□ Playpen area for user circuitry

□ IBM PC card form factor

□ RA87 μPD7800 series relocatable assembler package

□ Host control software for IBM PC, PC/XT, PC/AT, or compatibles

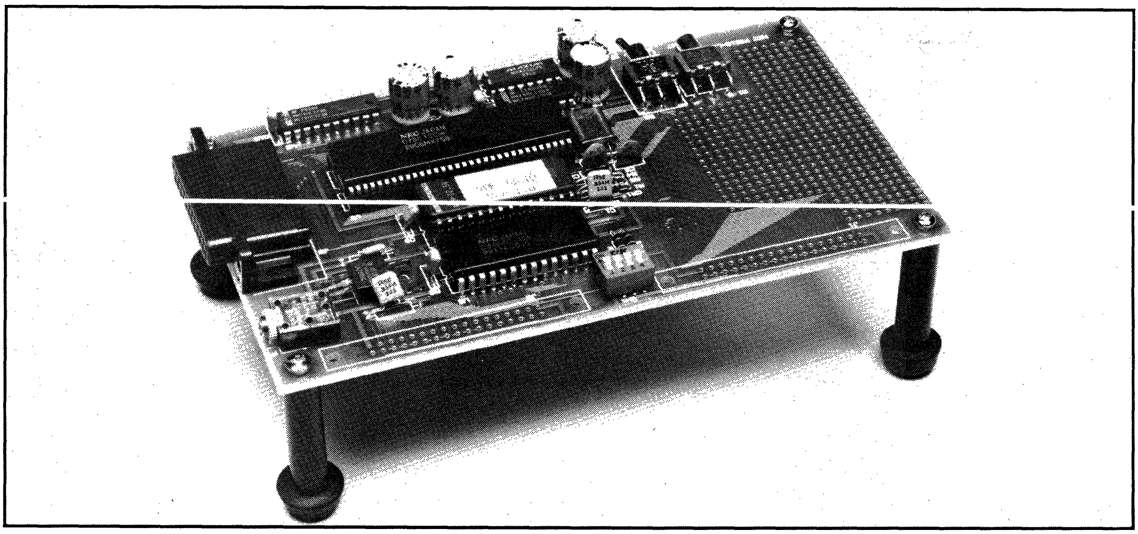
□ Source code for DDK-78C10 monitor included

Ordering Information

Part Number	Description
DDK-78C10	Evaluation board for the μPD78CXX series

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

DDK-78C10 Evaluation Board



Hardware Description

The DDK-78C10 features a μPD78C10 with 16K bytes of on-board memory. The first 8K bytes are dedicated to ROM and contain a powerful monitor program. The second 8K bytes are dedicated to RAM and can be used for user program storage. The internal RAM area of the μPD78C10 (addresses 0FF00H to 0FFFFH) is used for the monitor stack and data area.

The serial port of the μPD78C10 is connected through an RS-232C driver/receiver to an DB25 pin connector. A reset switch is provided to return the DDK-78C10 to the power-up state without losing the contents of the external RAM.

An AC/DC converter is provided to power the DDK-78C10 in the stand-alone mode. The DDK-78C10 can also receive its power directly from the IBM PC bus.

The DDK-78C10 block diagram is shown below.

Software Description

The DDK-78C10 comes with a powerful interactive monitor to facilitate software design with the μPD78CXX series. A user program can be downloaded into user RAM and executed in real-time with or without a breakpoint.

The DDK-78C10 supports one address breakpoint that can be specified in the Go command line. The monitor sets a breakpoint by substituting a software interrupt instruction (opcode 72H) for an instruction in the user program.

Additional commands are available to:

- Display, fill, change, or move memory
- Display the command list
- Input data from ports A, C, and port B (bits 2-7)
- Output data to ports A, C, and port B (bits 2-7)
- Repeat the previous command

Block Diagram

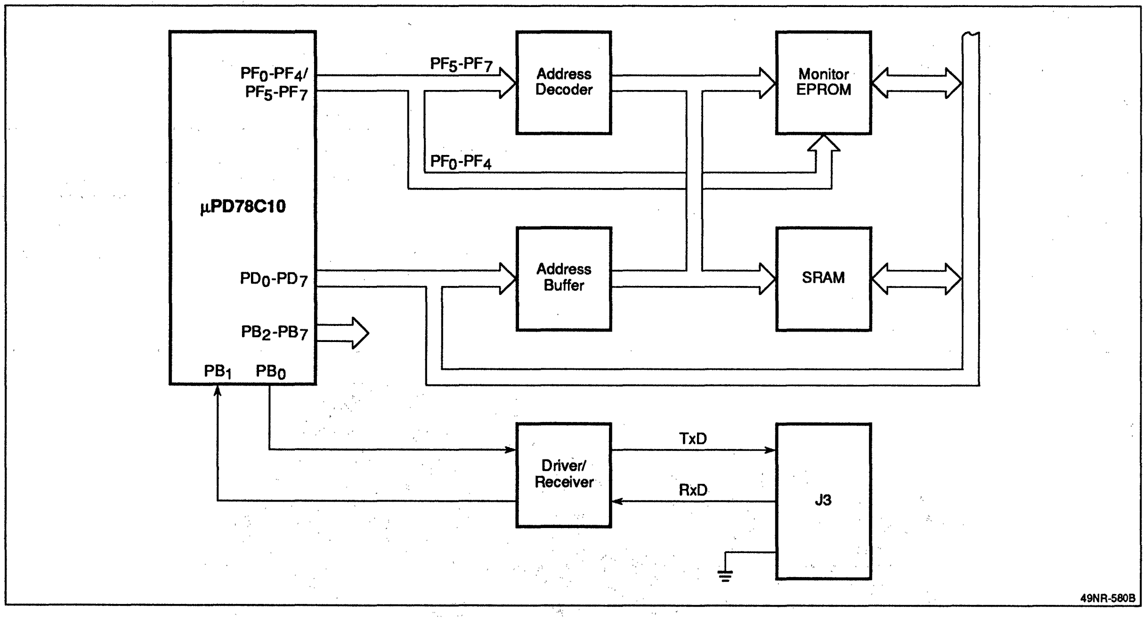


Table 1 contains a complete list of the DDK-78C10 monitor commands and their syntax.

Table 1. Command List

Command	Function	Syntax
C	Change memory byte	C[addr]
D	Display memory	D[saddr][,eaddr]
F	Fill memory	F[saddr],[eaddr],dd
G	Go (to breakpoint)	G[saddr][,baddr]
I	Input from port A,C, and port B (bits 2-7)	I[p]
H	Show this menu of commands	H
L	Load a HEX file on to the DDK-78C10	L[saddr]
M	Move a block of memory	M[saddr][,eaddr][,addr]
O	Output to port A, C, and port B (bits 2-7)	O[p]
R	Repeats the previous command	R

Notes:

(1) addr = 16-bit address in hexadecimal format

(2) dd = 8-bit value in hexadecimal format

(3) saddr = 16-bit start address in hexadecimal format

(4) eaddr = 16-bit end address in hexadecimal format

(5) p = ports A,B,C

RA87 Relocatable Assembler Package

The RA87 relocatable assembler package converts symbolic source code for the μPD7800 series 8-bit single-chip microcomputers into executable absolute address object code. A copy of RA87 is included with the DDK-78C10 for use on an IBM PC, PC/XT, PC AT, or compatible. Using this software, you can easily write evaluation programs for the μPD78CXX family.

Emulator Controller Program

Absolute address object files produced by the RA87 relocatable assembler package can be downloaded to the DDK-78C10 using the NEC emulator controller program which is supplied with the DDK-78C10. This controller program allows you to download files from your IBM PC or compatible to the DDK-78C10 board. In addition to downloading files, the NEC emulator controller program provides you the following additional capabilities:

- Complete DDK-78C10 control from host console
- On-line help facilities
- Host system directory and file display
- Storage of debug session on disk

License Agreement

RA87 is provided under the terms of a license agreement which is included with the DDK-78C10 board. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided to registered users.

Documentation

For further information on the DDK-78C10 evaluation board, NEC Electronics Inc. provides the following documentation:

- DDK-78C10 User's Manual

This manual is provided with the board. Additional copies may be obtained from NEC Electronics Inc.

9

NEC
NEC Electronics Inc.

IE-78C11 In-Circuit Emulator

Description

The IE-78C11 is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78C10, μPD78C11, μPD78C14, and μPD78CP14 eight-bit single-chip microcomputers. Real-time and single-step emulation, coupled with sophisticated memory mapping features, breakpoints and trace capabilities, create a powerful debugging environment. A line assembler and disassembler, full register and memory control, and complete upload/download capabilities simplify the task of debugging hardware and software. The IE-78C11 is designed to operate as a stand-alone, in-circuit emulator controlled from either a user terminal or a host computer system.

Features

□ Real-time and single-step emulation capability

□ User-specified breakpoints
  — Logical OR of up to four sets of break conditions:
    Opcode fetch count
    External sense clips condition
    Emulation time
    Logical AND of addresses, data values, CPU controls, and number of loops

□ Sophisticated trace capabilities
  — Instruction or machine cycle display
  — 1,024 trace frames
  — Address, control, data, and port trace

□ Powerful memory mapping
  — 64K bytes of RAM mappable in 256-byte blocks

□ Line assembler/disassembler

□ Operating state LED indicators

□ Latch-up warning for CMOS protection

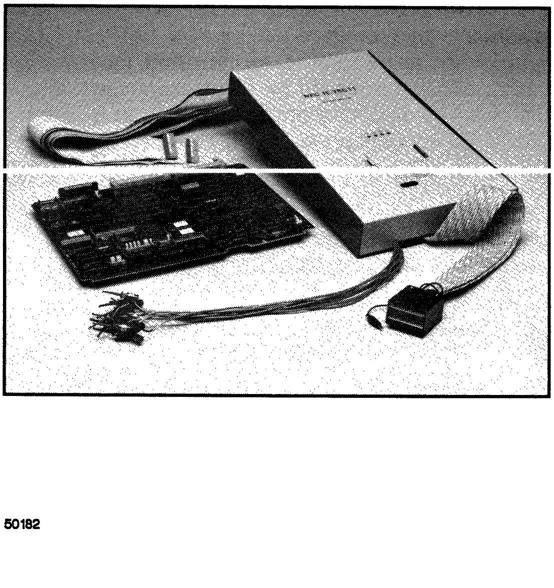
□ Eight external sense probes

□ Self-diagnostic command

□ Stand-alone configuration
  — User terminal controlled
  — Host computer system controlled

□ IE78C11 controller program for IBM PC®, PC/XT®, PC AT®, or compatibles
  — Symbolic debugging
  — Autoexecution of commands
  — On-line help facility
  — Debug session logging

IE-78C11 In-Circuit Emulator



Ordering Information

Part Number	Description
IE-78C11-M	In-circuit emulator for μPD78C10/C11/C14/CP14
EP-7811HGQ	Emulator probe for 64-pin QUIP package (shipped with IE-78C11)
EV-9001-64	Optional emulator probe adapter for 64-pin shrink DIP package (used with EP-7811HGQ)

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

9

Hardware Description

The IE-78C11 hardware consists of a controller module, driver module, interface probe, external sensing clips, and the interconnecting cables. The controller module, responsible for real-time trace and control of the driver module, houses the host CPU, two RS-232C serial ports, and an IEEE-796 bus connection. The driver module containing the emulation chip and associated control logic is connected to the controller module by two 50-pin flat cables. The driver module interfaces to the prototype system through the 64-pin emulation probe and eight external sensing clips used for monitoring user-selected signals in the prototype hardware.

Memory Mapping

The IE-78C11 incorporates a sophisticated memory mapping scheme which allows access to up to 64K bytes of memory mapped in 256-byte units. The map command allocates memory space of the emulation CPU either to the user system or to the IE system. Even if development of the target system is not complete, software debugging is possible by using this internal RAM in place of the target system RAM. When memory is mapped as internal ROM of the IE-78C11, write-protect becomes operative.

Emulation

Following termination of real-time emulation or during single-step emulation, the registers, stack pointer, program status word, and program counter are displayed. Following a real-time emulation break, the IE-78C11 automatically enters the single-step mode. Each time the space bar is pressed during single-step emulation, one instruction is executed and the trace data, disassembly list, and the register contents are displayed.

Breakpoint Capabilities

The following three conditions cause a break in real-time emulation:

- Entering the ESC (escape) key on the user terminal
- Attempting access to a non-mapped area
- Satisfying a user-designated breakpoint

Four user-designated breakpoints may be selected from a combination of address registers, data registers, or control signals. A break can also be set in the following ways: by a loop counter, by an instruction count, by a timer function set in the range of 1 to 65,535 ms, and by matching user-specified conditions for the eight external sense signals.

Trace Capabilities

The IE-78C11 has a 1K x 56-bit trace RAM for storing emulation data for each machine cycle. For the range specified by the user, a trace can be performed on the address, data, and control signals, including $\overline{RD}$, $\overline{WR}$, OP and IO/M as well as ports A and B and the signals from the eight external sense clips, for up to 1,023 machine cycles. In machine cycle display mode, the trace display includes the address, data, cycle, port A and port B. In the instruction cycle trace mode, the trace display includes the address, object, label, mnemonic, port A and port B.

Self-Diagnostics

A self-diagnostic command monitors the IE-78C11 for error-free operation. It checks internal memory, ports A, B, C, D, and F, the analog inputs, pins MODE0 and MODE1, and the serial I/O lines.

Utilities

The upload/download commands provide easy loading and saving of hex files to and from a disk. The on-board assembler/disassembler allows the user to avoid programming in machine code. Display/change register/ memory commands give the user full data manipulation capability. Initialize commands allow the user to choose a clock source and a base number, and to define memory locations.

Operating States and CMOS Protection

Four LED indicators HALT, SOFT STOP, HARD STOP, and LATCH-UP are provided on the top panel of the driver module to indicate the IE-78C11 operating state. HALT, and SOFT STOP will light when executing a HLT or STOP instruction. The HARD STOP LED lights when a low level is input on the STOP pin. The LATCH-UP LED lights when any CMOS IC in the driver module is in danger of being damaged by improper voltage levels on the pins. A protection circuit is activated to isolate the CMOS ICs from the power supply.

Emulation Accuracy

Software can designate ports D and F as either input or output ports if they are being used to communicate with a peripheral device with a bidirectional data bus. However, when using the IE-78C11, the user must specify the port direction upon power-up, and only use the port in that configuration. The low level output voltage of $\overline{RD}$, $\overline{WR}$, ALE, PD_0-PD_7, or PF_0-PF_7 for the µPD78C10/78C11/78C14 is typically .45 V. Depending on the conditions, the emulator may deviate up to ± 10% of this rating.

IE78C11 Controller Program

The IE-78C11 can be connected to an IBM PC, PC/XT, PC AT or compatible by an RS-232C port and operated in system mode. By using the accompanying control software, the debugging capabilities of the IE-78C11 are greatly increased. Macro command file capability allows the user to execute a defined set of commands automatically. The on-line help facility, the history command, and the ability to store the console display on disk ease debugging tasks. The uploading and downloading capability can be utilized to upload and download both object code and symbol information. Other advantages are an alter symbol command and a terminate command for exiting to the operating system.

Table 1 lists commands available for both the stand-alone and IBM PC controlled configurations of the IE-78C11. Commands listed in table 2 supplement table 1, but can be used only with the IBM PC based controller program, IE78C11.

Table 1. Stand-Alone and IBM PC Based Controller Program

Command	Function
ASM	Line assemble command
BR?	Changes or displays the breakpoint register used for stopping real-time emulation
CLK	Clock command (internal or external)
DAS	Disassemble command
DIG	Self-diagnostic command
LOD	Loads hex format file into program memory
MAP	Memory mapping (64K bytes are accessible)
MAT	Calculator function
MDR	Displays or modifies the mode registers of emulator CPU
MEM	Memory manipulation command
MOV	Moves memory content to different mapping area
REG	Displays or modifies the registers of emulator CPU
RES	Resets IE-78C11 and emulator CPU
RUN	Commences execution of emulator CPU in real-time with options for break conditions
SAV	Saves contents of hex memory onto disk
SPR	Displays or modifies the special registers of emulator CPU
SUF	Base number specification (hex, octal, binary, decimal)
TR?	Changes or displays the trace conditions for both real-time or single-step emulation

Table 2. IBM PC Based Controller Program

Command	Function
DIR	Displays filenames
EXT	Terminates IE-78C11 operation
HLP	Displays command format
LOD	Loads hex format and symbol files
LST	Stores console display on disk
MOD	Inputs local symbols of specified modules
PAG	Displays and changes V register value
SAV	Saves object code and symbol table onto disk
STR	Automatically executes macro command file
SYM	Clears, displays, or changes a symbol

Equipment Supplied

The IE-78C11-M package consists of the following:

- IE-78C11 controller module
- IE78C11 controller program (IBM PC based)
- IE-78C11 driver module with 64-pin QUIP emulation probe and eight external sense probes
- Power supply connector
- Serial communication cable for RS-232C to RS-232C
- Serial communication cable for TTL to RS-232C
- IE-78C11 user's manual
- Standoffs and associated hardware
- Warranty policy and registration card

Basic Specifications

Control module:

- Weight: 560 g
- External dimensions: length, 230 mm; width, 305 mm
- Power consumption: 6.5 A (+5 V max), 0.5 A (+12 V max), 0.5 A (-12 V max)

Driver module:

- Weight: 2,600 g
- External dimensions: length, 400 mm; width, 230 mm; height, 48 mm

Environmental Characteristics

- Operating temperature range: 0 to +45°C
- Storage temperature range: –10 to +55°C
- Ambient humidity range: 30 to 85% relative humidity

Documentation

For further information on IE-78C11 operation, NEC Electronics Inc. provides the in-circuit emulator together with the following manuals:

- IE-78C11-M In-Circuit Emulator Stand-Alone User's Manual
- IE78XX Controller Program User's Manual (IBM PC Based)
- IE-78C11 Sample Session User's Manual (IBM PC Based)

Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

CC87 Micro-Series™ C Compiler Package for the μPD7800 Series

Description

The CC87 Micro-Series™ C compiler package for the NEC μPD7800 series of microcomputers consists of an ANSI C cross compiler, relocatable macro assembler, linker, library manager, loader, and converter. Developed by IAR systems in Sweden for NEC, the Micro-Series C compiler package is available for use on an MS-DOS®, VAX/VMS®, or VAX/UNIX™4.2BSD or ULTRIX® system with a free-standing system as target (embedded system). The target microcomputers supported by this package are: μPD7807/09, μPD7810/10H, μPD7811/11H, μPD78PG11/PG11H, and the μPD78C10/C11/C14.

Ordering Information

Part Number	System	Description
CCMSD-I5DD-87	MS-DOS	5-1/4" double-density floppy diskette
CCVMS-OT16-87	VAX/VMS	9-track 1600 BPI magnetic tape
CCUNX-OT16-87	VAX/UNIX4.2BSD or ULTRIX	9-track 1600 BPI magnetic tape

C CROSS COMPILER (ICC7800)

Description

The C cross compiler which is the ICC7800 program, converts standard C source code into relocatable object modules in the IAR systems proprietary universal binary relocatable object format (UBROF). This format is used for all relocatable object files in the micro series development system, whether generated by an assembler or compiler.

Features

- □ ANSI standard C
  - — Const, volatile, signed, void, enum keywords
  - — Function prototyping
  - — Hex strina constants
  - — Structure and union assignments
- □ UNIX LINT functions (legal C code verification) integrated into the compiler
- □ Interface checking between modules performed by the linker XLINK
- □ Library interface checking
- □ Generation of list and full cross reference files
- □ Built in help facility
- □ Simple diagnostics

C Library Functions

The CC87 Micro-Series C compiler package includes most of the important C library functions that apply to PROM-based embedded systems. All library functions reside in the supplied library files. Header files that declare the set of library functions are also included.

The following library functions are available:

CHARACTER HANDLING <ctype.h>
isalnum isalpha iscntrl isdigit islower
isprint ispunct
isspace isupper tolower toupper

NON-LOGICAL JUMPS <setjmp.h>
longjmp setjmp

FORMATTED INPUT/OUTPUT <stdio.h>
getchar printf putchar sprintf _formatted_write

GENERAL UTILITIES <stdlib.h>
calloc exit free malloc ralloc

STRING HANDLING <string.h>
strcat strcmp strcpy strlen strncat strncmp strncpy

MATHEMATICS <math.h>
atan atan2 cos exp log log10 modf pow sin sqrt tan

Memory Models

There are two memory models, static and reentrant, which differ only in allocation of auto variables. In the reentrant mode, all local auto variables are allocated and deallocated dynamically; the auto variables reside on the stack, which is necessary if recursive or reentrant functions are needed. This option sometimes generates more code and slower code than the static mode. In the static mode, all function level variables are put into static memory, with the exception of function arguments which are always placed on the stack.

Micro-Series is a trademark of IAR Systems AB.
MS-DOS is a registered trademark of Microsoft Corporation.
VAX and VMS are registered trademarks of Digital Equipment Corporation.
UNIX is a trademark of AT&T.
ULTRIX is a registered trademark of Digital Equipment Corporation.

9

RELOCATABLE MACRO ASSEMBLER (A7800)

Description

The relocatable macro assembler (A7800), translates symbolic source code for the NEC μPD7800 series of microcomputers into relocatable object modules in the IAR systems proprietary UBROF format.

Features

The relocatable macro assembler features are as follows:

- □ Absolute or relocatable address object code output
- □ Directives
  - — List formatting
  - — Conditional assembly, separate assembly
  - — Memory allocation
  - — Macro definition and value assignments to symbol directives
- □ Generation of list files
- □ Generation of cross reference and symbol tables
- □ Ability to include files in another source

Directives

Assembler directives give instructions to the program but are not translated into machine code during assembly. Basic directives include those for storage definition and memory allocation (DB, DD, DW, DS); symbol control and usability (PUBLIC, EXTERN, LOCSYM); and value assignments to symbols (SET, EQU, =, DEFINE).

Program control directives include those for module definition (NAME, MODULE, ENDMOD); segment definition and control (ASEG, RSEG, STACK, COMMON, ORG); conditional assembly (IF, ELSE, ENDIF); macro processing (MACRO, ENDMAC); and listings control (LSTOUT, LSTCND, LSTCOD, LSTEXP, LSTMAC, LSTWID, LSTFOR, LSTPAG, PAGSIZ, PAGE, TITL, STITL, PTITL, PSTITL, LSTXRF).

LINKER (XLINK)

The universal linker, XLINK, combines relocatable object modules and absolute load modules and produces one absolute load module. The controls for XLINK may be specified either on the command line or in a parameter file. In addition to being able to generate several types of absolute load module formats, it is also possible to generate cross reference lists with an index list; define segment allocation; force load and conditional load of files; bank segments; and define a symbol on a command line. The absolute load module can contain symbol information as well as absolute object code.

LOADER (RC7800) AND CONVERTER (CONVERT)

Linker output is usually fed to the target system RAM/PROM or emulator using the RC7800 loader or other user program. Previously written assembler programs, coded by the NEC μPD7800 family assembler, may be converted to the A7800 assembler format using the CONVERT program.

LIBRARIAN (XLIB)

The XLIB librarian creates and maintains files containing relocatable object modules. With XLIB, the user can merge object files from different assemblies/compilations in order to create libraries; delete individual modules, change the order of modules and check the CRC in a module; and rename modules, segments, externals or entries. In addition, XLIB can change the properties of a module to be conditionally or unconditionally loaded. Use of XLIB reduces the number of files that need to be linked together by allowing several modules to be kept in a single file, providing an easy way to link frequently used modules into programs.

LICENSE AGREEMENT

CC87 Micro-Series C Compiler package is sold under terms of a license agreement, which is included with the purchased copies of the compiler. The accompanying card must be completed and sent to NEC Electronics Inc. to register the license. Software updates are provided free to registered users.

DOCUMENTATION

For further information on source program format, compiler operation, assembler operation, linker, librarian, and converter programs, and actual program examples, NEC Electronics Inc. provides the following documentation:

- Micro-Series ANSI C Cross Compiler for Microprocessor Development
- ICC7800 Micro-Series ANSI X Cross-Compiler Appendix for the 7800 Microprocessor Family
- Micro-Series Assemblers, Linker, and Librarian for Microprocessor Development
- A7800 Micro-Series 7800 Family Assembler Reference Manual
- Micro-Series RC7800 and Converter Manual

This documentation is provided with purchased copies of the Micro-Series C Compiler package.

NEC
NEC Electronics Inc.

RA87 Relocatable Assembler Package for the μPD7800 Series

Description

The RA87 relocatable assembler package converts symbolic source code for the μPD7800 series of microcomputers into executable absolute address object code. The μPD7800 series relocatable assembler package consists of six separate programs: assembler (RA87), linker (LK87), hexadecimal format object converter (OC87), librarian (LB87), list converter (LCNV87), and macroprocessor (MP).

RA87 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcomputer specified at assembly time.

LK87 combines relocatable object modules and absolute load modules and converts them into an absolute load module. OC87 converts an absolute object module or an absolute load module into an ASCII hexadecimal format object file.

LB87 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input to the linker, the linker extracts only those modules required to resolve external references from the file and relocates and links them into the absolute load module.

LCNV87 allows relocatable list files to be converted into absolute list files. MP expands macros contained in a source program prior to assembling.

Features

□ Absolute address object code output

□ Generic jump capability

□ User-selectable and directable output files

□ Extensive error reporting

□ Macro Capabilities

□ Runs under MS-DOS® and VAX®/VMS® operating systems

MS-DOS is a registered trademark of Microsoft Corporation.
VAX and VMS are registered trademarks of Digital Equipment Corporation.

Ordering Information

Part Number	System	Description
RA87-D52	MS-DOS	5-1/4 inch double-density floppy diskette
RA87-VVT1	VAX/VMS	9-track 1600 BPI magnetic tape

Program Syntax

An RA87 source module consists of a series of code, byte-oriented data, or bit-oriented data segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, comment.

The symbol field may contain a label whose value is the instruction or data address or a name which represents an instruction address, data address or a constant. The mnemonic field may contain an instruction or an assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, NOT, AND, OR, XOR, EQ, NE, GT, GE, LT, LE, SHR, SHL, HIGH byte, LOW byte, MOD, and the - sign.

Assembler Directives

Assembler directives give instructions to the assembler but are not translated into machine code during assembly. Basic assembler directives include: storage definition (DB, DW, DS, DBIT); symbol definition (EQU, SET, CODE, DATA, BIT); and program boundary definition (ORG, END). Program linkage directives are provided to NAME the module and to declare symbols as PUBLIC or external (EXTRN).

Segment definition directives define whether a segment is a code segment (CSEG), allocated to ROM; a data segment (DSEG) or a bit segment (BSEG), allocated to RAM; or a working register segment (VREG). The address boundary conditions for each segment directive are specified in its operand. These include UNIT, PAGE, INPAGE, FIXEDAREA, BYTE, CALLTABLE, AT, BITADDRESSABLE. The combination types of PUBLIC, COMMON and COMPLETE, specified in the operand, define how to link segments with the same name and segment definition.

The μPD7800 series instruction set contains three jump instructions with varying legal address ranges. To avoid calculating which jump instruction to use, the programmer can substitute the generic jump (GJMP) directive for any relative jump (JR), any extended relative jump (JRE), or any long jump (JMP) instruction in the source program. During assembly a suitable jump instruction is chosen for each GJMP directive.

Assembler Controls

The RA87 assembler (figure 1) has two types of controls. The primary controls, which are specified in the assembler command line, a parameter file, or at the beginning of the source module, are as follows:

- Target microcomputer specification
- Output file selection and destination
- Listing format controls
- Date specification

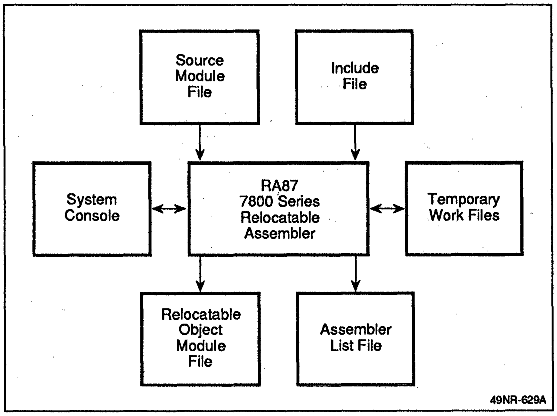
The general controls, specified in the assembler command line, a parameter file, or at any place in the source program, are as follows:

- Generation/suppression of listing
- Listing titles
- Inclusion of other source files (in source program only)
- Page eject (in source program only)

The listing file may contain the complete assembly listing or only lines with errors, and a symbol table or a cross reference table. The symbol table shows all defined symbols in alphabetical order, their types, attributes, and the values initially assigned to them. The cross-reference table contains all defined symbols and the numbers of all statements that refer to them.

The object file contains the relocatable object module. The format of this module is a NEC proprietary relocatable object module format. This object file may also contain local symbol information for the symbolic debugger.

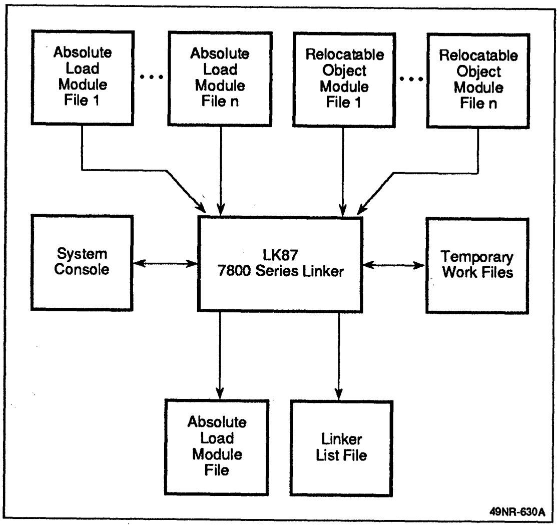
Figure 1. Relocatable Assembler Functional Diagram



Linker

The LK87 linker (figure 2) combines several relocatable object modules or absolute load modules, resolving PUBLIC/EXTRN references between modules, to create an absolute load module. This load module contains both absolute object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK87 can be specified in either the command line or in a parameter file. The programmer can specify the date, module name, stack size and starting address, ROM/RAM segment allocation, starting address and order for code/data/bit relocatable segments, and the page address for the working register group. The programmer may also specify that a list file containing a link map, a local symbol table, or a public symbol table be created.

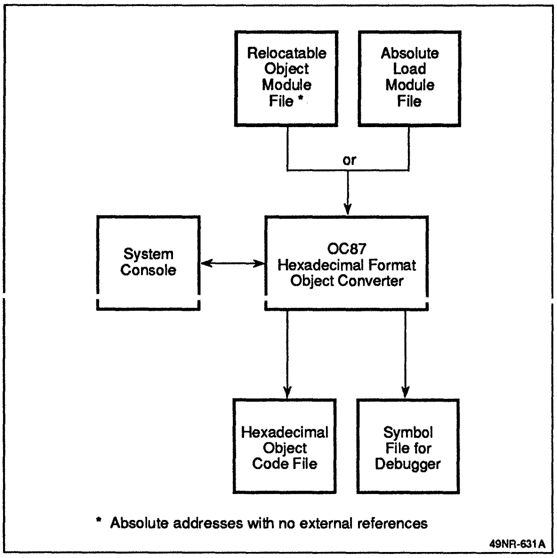
Figure 2. Linker Functional Diagram


Absolute Load Module File 1
Absolute Load Module File n
Relocatable Object Module File 1
Relocatable Object Module File n
System Console
LK87 7800 Series Linker
Temporary Work Files
Absolute Load Module File
Linker List File
49NR-630A

Hexadecimal Format Object Converter

The OC87 object converter (figure 3) outputs the object code file in ASCII hexadecimal format, which can be downloaded to a prom programmer or hardware debugger. The programmer can specify whether or not to generate a symbol file for a hardware debugger.

Figure 3. Hexadecimal Format Object Code Converter Functional Diagram


Relocatable Object Module File *
Absolute Load Module File
or
System Console
OC87 Hexadecimal Format Object Converter
Hexadecimal Object Code File
Symbol File for Debugger
* Absolute addresses with no external references
49NR-631A

Librarian

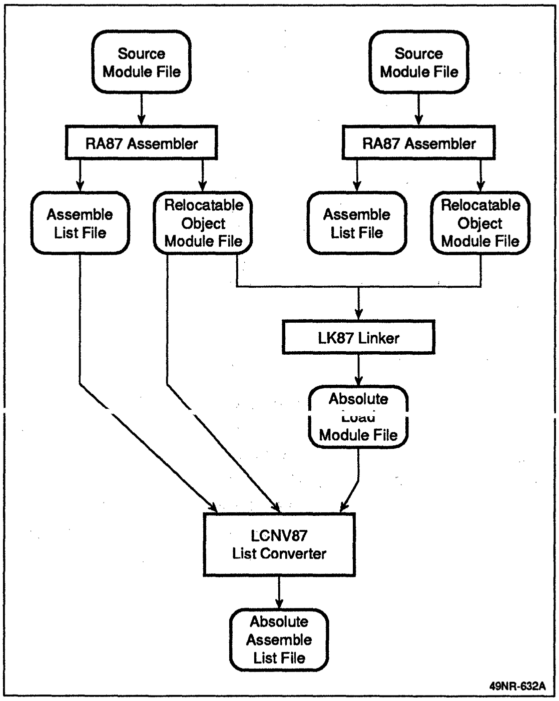
The LB87 librarian creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by allowing several modules to be stored in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to, deleted from, or the contents of the library file can be listed.

List Converter

Normally, listing files produced by a relocatable assembler do not show the final absolute address for instructions, as their location is not decided until link time. The address shown in the listing is only the offset from the start of the code or data segment.

The LCNV87 list converter (figure 4) uses the assembly list and object module files from the assembler and the load module file from the linker, to create an absolute address assembly listing. This absolute listing shows the addresses of instructions as their final absolute address in memory, and is useful in debugging or program documentation. The programmer can specify the load module (-L), assembly list (-A), and output assembly (-O) file names.

Figure 4. List Converter Functional Diagram

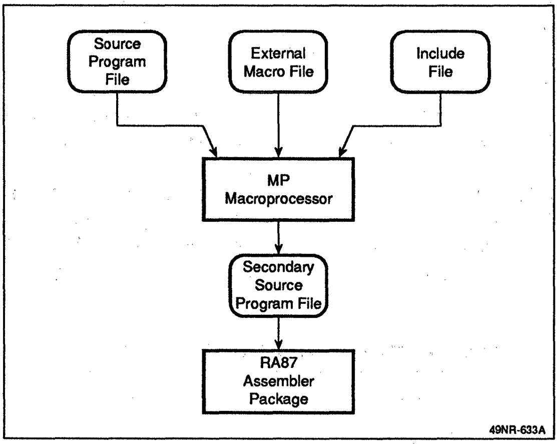


Macroprocessor

The MP macroprocessor (figure 5) interprets the macros described in a source program and expands them to create another source program, which can then be input to the assembler. It has the following three main functions:

- Expands macros by defining and referencing them
- Reads and expands include files
- Selects an assembler source based on a conditional macro instruction

Figure 5. Macroprocessor Functional Diagram



Operating Environment

The NEC RA87 package can run under a variety of operating systems. A version is available to run on a MS-DOS system with one or more disk drives and at least 128K of system memory. Another version is available to run on a Digital Equipment Corporation VAX computer under the VMS (Version 4.1 or later) operating system.

Emulator Controller Program

Absolute object files produced by the RA87 relocatable assembler package can be debugged using the appropriate NEC stand-alone in-circuit emulator. NEC emulator controller programs allow communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, and PC/AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provides the following features:

- Uploading and downloading of object and symbol files
- Symbolic debugging
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session

License Agreement

RA87 is sold under terms of a license agreement which is included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users.

Documentation

For further information on source program formats, assembler operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- RA87 μPD7800 Series Relocatable Assembler Package User's Manual
- MP Macroprocessor User's Manual

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

IBM PC, PC/XT, and PC/AT are registered trademarks of International Business Machines Corporation.

DK-78K2
µPD782XX Designer Kits

Description

The DK-78K2 designer kits are powerful development toolboxes for the 78K2 family of eight-bit microcomputers. Each kit includes all the hardware and software to design and implement elaborate embedded control applications for the µPD7821X, µPD7822X, or µPD7823X. To enhance the development process, each kit includes NEC's new ST78K2 structured assembler preprocessor, which provides high-level language constructs without code inefficiency.

The DK-78K2 features an EB-782XX emulation board with either a µPD78213, µPD78220, or µPD78230 microcomputer, probe connector, on-board monitor, and serial interface for an IBM PC®, PC/XT®, PC/AT®, or compatible computer. The EB-782XX emulation board can be used without a target system or can be directly connected to a user's system with the enclosed emulation probe.

The on-board monitor facilitates access to RAM, ROM, I/O, and special function registers in a real-time environment. Programs can be downloaded to the on-board 32K byte memory for evaluation and debugging. A line assembler and disassembler provide easy code debugging and modification. An NEC emulator controller program on disk makes it possible to download code from an IBM PC and provides complete control of the EB-782XX from the console of the PC.

The kit includes both the RA78K2 relocatable assembler package and the ST78K2 structured assembler preprocessor for software development. Source modules consist of a combination of structured and pure assembly language which reduces development time and effort. A complete set of documentation is provided for the EB-782XX, its two software packages, the target µPD782XX microcomputer, and other NEC support products for the 78K2 family.

Features

□ EB-782XX emulation board

□ 32K bytes of static RAM

□ Resident monitor

□ Emulation probe

□ Power supply

□ RA78K2 relocatable assembler package

□ ST78K2 structured assembler preprocessor

□ Emulator controller program for IBM PC, PC/XT, PC AT, or compatibles

□ Full documentation package

Ordering Information

Part Number	Description
DK-78K2-21XCW	µPD7821X designer kit for shrink DIP package
DK-78K2-21XGJ	µPD7821X designer kit for QFP package
DK-78K2-21XGQ	µPD7821X designer kit for QUIP package
DK-78K2-21XL	µPD7821X designer kit for PLCC package
DK-78K2-22XGJ	µPD7822X designer kit for QFP package
DK-78K2-22XL	µPD7822X designer kit for PLCC package
DK-78K2-23XGC	µPD7823X designer kit for 80-pin QFP package
DK-78K2-23XGJ	µPD7823X designer kit for 94-pin QFP package
DK-78K2-23XL	µPD7823X designer kit for PLCC package

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

NEC

NEC Electronics Inc.

EK-78K2
μPD782XX Evaluation Kits

Description

The EK-78K2 are powerful evaluation kits for the 78K2 family of eight-bit microcomputers. The EK-78K2 allows full evaluation of the μPD7821X, μPD7822X, or μPD7823X in either a stand-alone or an application environment. Each kit includes all of the tools to write and test application software, and to experiment with the 78K2 hardware. Also included for user evaluation is a copy of NEC's new ST78K2 structured assembler preprocessor, which provides high-level language constructs without code ineffiency.

The EK-78K2 features a DDB-78K2 evaluation board with either the μPD78213, μPD78220, or μPD78233 microcomputer and a serial interface that will convert an IBM PC®, PC/XT®, PC AT®, or compatible into a 78K2 design center. The on-board monitor facilitates the hex object code downloading from a PC to the resident RAM on the DDB board, where it then can be executed in real-time. An NEC emulator controller program on disk allows code to be downloaded from an IBM PC and provides complete control of the DDB-78K2 from the PC console.

The kit includes both the RA78K2 relocatable assembler package and the ST78K2 structured assembler preprocessor for writing evaluation programs. A complete set of documentation is provided for the DDB-78K2, its two software packages, the target μPD782XX microcomputer, and other NEC support products for the 78K2 family.

Features

- □ DDB-78K2-2XX evaluation board
- □ 32K bytes of static RAM
- □ Resident monitor ROM
- □ Expansion ROM socket for extended data memory
- □ Nine square inches of user prototype area
- □ Power supply
- □ RA78K2 relocatable assembler package
- □ ST78K2 structured assembler preprocessor
- □ Emulator controller program for IBM PC, PC/XT, PC AT, or compatibles
- □ Full documentation package

Ordering Information

Part Number	Description
EK-78K2-21X	μPD7821X evaluation kit (IBM PC Based)
EK-78K2-22X	μPD7822X evaluation kit (IBM PC Based)
EK-78K2-23X	μPD7823X evaluation kit (IBM PC Based)

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

NEC
NEC Electronics Inc.

IK-78K2
μPD782XX In-Circuit Emulator Kits

Description

The IK-78K2 in-circuit emulator kits are the ultimate debugging tools for the 78K2 family of eight-bit microcomputers. Each kit includes all of the hardware and software to design and implement elaborate embedded control applications for the μPD7821X, μPD7822X, or μPD7823X. To enhance the user's development process, each kit includes NEC's new ST78K2 structured assembler preprocessor, which provides high-level language constructs without code inefficiency.

The IK-78K2 features an IE-782XX in-circuit emulator and emulator probe. The IE-782XX can be used without a target system or can be connected directly to a user's system with an enclosed emulator probe. This full feature emulator for the μPD7821X, μPD7822X, or μPD7823X microcomputers provides upload/download capabilities from an IBM PC®, PC/XT®, PC AT® or compatible computer using the NEC emulator controller program on disk. This controller program allows the in-circuit emulator (IE) to be controlled directly from a PC console and enhances the IE with an added HELP facility, STRING command file capability, and HISTORY command.

Real-time and single-step emulation capability together with extremely sophisticated breakpoint and trace capabilities create a powerful, real-time debugging environment. All memory can be written to or read from, displayed using the disassembler, altered by the line assembler and traced without restrictions. All special function registers can also be displayed and altered. Up to 32K bytes of internal high-speed memory can be mapped for internal ROM emulation.

The kit includes both the RA78K2 relocatable assembler package and the ST78K2 structured assembler preprocessor for software development. Source modules can consist of a combination of structured and pure assembly language which greatly reduces development time and effort. A complete set of documentation is provided for the IE-782XX, its two software packages, the target μPD782XX microcomputer, and other NEC support products for the 78K2 family.

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

Features

- □ IE-782XX in-circuit emulator
- □ Emulation probe
- □ RA78K2 relocatable assembler package
- □ ST78K2 structured assembler preprocessor
- □ Emulator controller program for IBM PC, PC/XT, PC AT, or compatibles
- □ Full documentation package

Ordering Information

Part Number	Description
IK-78K2-21XCW	μPD7821X in-circuit emulator kit for shrink DIP package
IK-78K2-21XGJ	μPD7821X in-circuit emulator kit for QFP package
IK-78K2-21XGQ	μPD7821X in-circuit emulator kit for QUIP package
IK-78K2-21XL	μPD7821X in-circuit emulator kit for PLCC package
IK-78K2-22XGJ	μPD7822X in-circuit emulator kit for QFP package
IK-78K2-22XL	μPD7822X in-circuit emulator kit for PLCC package
IK-78K2-23XGC	μPD7823X in-circuit emulator kit for 80-pin QFP package
IK-78K2-23XGJ	μPD7823X in-circuit emulator kit for 94-pin QFP package
IK-78K2-23XL	μPD7823X in-circuit emulator kit for PLCC package

NEC
NEC Electronics Inc.

DDB-78K2 Evaluation Boards for the μPD782XX Series

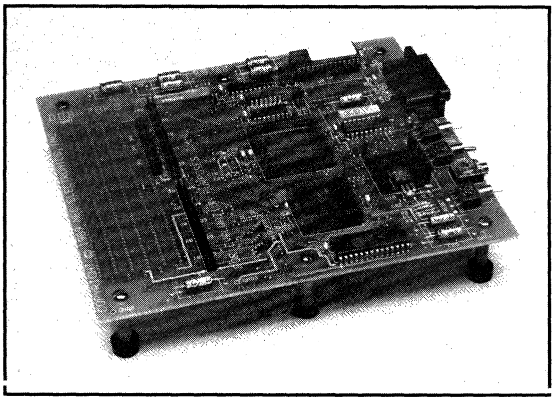
Description

The DDB-78K2 are evaluation boards for the NEC μPD782XX eight-bit, single-chip microcomputers. The DDB-78K2 provides maximum flexibility when evaluating and designing with the μPD782XX family of microcomputers. Every DDB-78K2 features a μPD78213, μPD78220 or μPD78233 microcomputer, 32K bytes of ROM, 32K bytes of RAM, μPD27C512 footprint for 64K bytes of optional extended data memory, RS-232C communication port, and a powerful monitor program. A playpen area is included for evaluating the μPD782XX with application specific hardware.

Features

- □ μPD78213, μPD78220, or μPD78233 evaluation board
  - — Convertible by changing microcomputer and firmware
- □ On-board memory
  - — ROM: 32K bytes
  - — RAM: 32K bytes

DDB-78K2 Evaluation Board



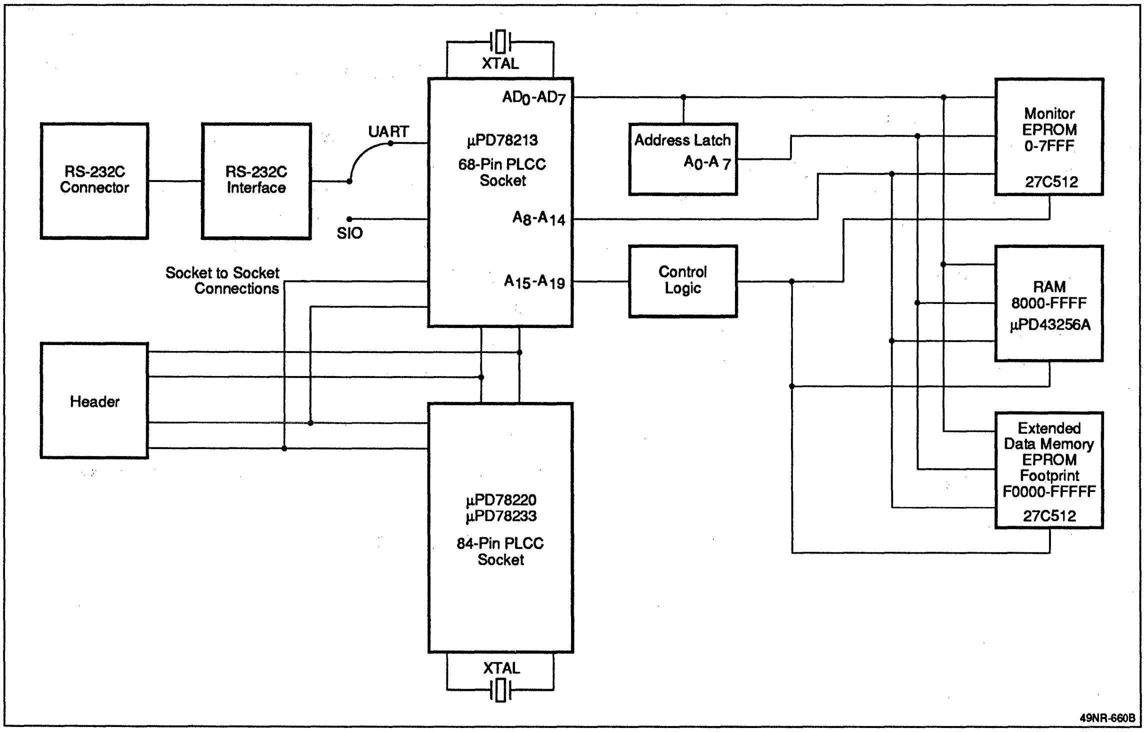
- □ μPD27C512 footprint for 64K bytes of extended data memory
- □ Powerful on-board debug monitor
  - — Real-time and single-step operation
  - — Display/change memory and internal registers
  - — Multiple software breakpoints
  - — User program download capability
- □ RS-232C serial interface for terminal or host computer
- □ Playpen area for user circuitry
- □ Includes AC/DC converter

Ordering Information

The DDB-78K2 evaluation boards are sold only as part of the following:

- ● EK-78K2 evaluation kits

DDB-78K2 Block Diagram



Hardware Description

The DDB-78K2 features 64K bytes of on-board memory. The first 32K bytes are dedicated to ROM and include a powerful monitor program. The second 32K bytes are dedicated to RAM and can include a user area for user program downloading (28K bytes), interrupt and CALLT re-vector area (256 bytes), monitor work area (3.8K bytes), and the internal RAM and register area of the μPD78213, μPD78220, or μPD78233 (768 bytes). The DDB-78K2 contains a footprint for a user installed μPD27C512 EPROM. This provides access to 64K bytes of extended data memory space (0F0000H to 0FFFFFH).

The microcomputer UART is connected to a DB25 pin connector through an RS-232C driver/receiver. If the capabilities of the UART need to be evaluated, a jumper selectable option allows the clock-synchronized serial interface (SIO) to be used in place of the UART for communicating to a terminal or host computer.

All the microcomputer pins are connected to wirewrap headers. This provides a convenient place for attaching oscilloscope probes for performing detailed signal analysis or for connecting application specific hardware.

A reset switch allows the DDB-78K2 to return to the power-up state without losing the contents of the external RAM. An NMI switch returns control from a user program to the monitor while saving the user's state. An AC/DC converter provides power to the DDB-78K2.

Software Description

Every DDB-78K2 has a powerful interactive monitor to facilitate software design for the μPD782XX microcomputer. A user program can be downloaded into user RAM and executed in real-time with or without breakpoints or executed one instruction at a time. During single-stepping, the registers, program counter, and the next instruction to be executed are displayed.

The DDB-78K2 has nine address breakpoints. The user can set up to eight of these prior to program execution. The ninth breakpoint is reserved for use in the GO command line. The monitor sets a breakpoint by substituting a software break instruction (opcode 5EH) for an instruction in the user's program.

Additional commands are available to:

- Display, fill, change, or move memory
- Display or change the general and special function registers
- Disassemble memory
- Display the command list
- Initialize the interrupt and call table re-vector areas
- Set the monitor's environment

Table 1 contains a complete list of the DDB-78K2 monitor commands and their syntax.

Table 1. Command List

Command	Function	Syntax
?/H	Print this summary of commands	? or H
B	Show or set breakpoints	B{bp}{,addr}
C	Change memory bytes	C{{b:}addr}{,val}
D	Display memory bytes	D{{b:}addr}{,addr}
E	Show or set environment variables	E{var,val}
F	Fill memory bytes with value	F{b:}addr,addr,val
G	Go (execute to breakpoint)	G{addr}{,addr}
I	Initialize interrupt & CALLT vectors	I
K	Kill breakpoint(s)	K{bp}
L	Load HEX file into memory	L{{b:}addr}
M	Move a block of memory	M{b:}addr,addr,{b:}addr
R	Display/change registers	R{s,}{reg{,val}}
S	Display/change special function registers	S{sfr{,val}}
T	Trace execution (trace mode)	T{addr}
U	Unassemble a block of memory	U{addr}{,addr}

Notes:

(1) addr = 16-bit address in hexadecimal format.

(2) b = four-bit bank number in hexadecimal format (0 - FH).

(3) bp = breakpoint number (0 - 7).

(4) reg = general purpose register mnemonic.

(5) s = register bank selector (0 - 3).

(6) sfr = special function register mnemonic.

(7) val = eight-bit value in hexadecimal notation.

(8) var = environment variable mnemonic.

(9) { } = optional parameter.

Documentation

For further information on the DDB-78K2 Evaluation Board, NEC Electronics, Inc. provides the following manual:

- DDB-78K2 μPD782XX Evaluation Board User's Manual

This manual is provided with the board. Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

EB-78210 Evaluation Board for the µPD78213

Description

The EB-78210 is an evaluation board for the NEC µPD78213 eight-bit, single-chip microcomputer. The EB-78210 provides a simple way to evaluate the capabilities of the µPD78213 in an application without having to build a prototype. If it is necessary to connect the EB-78210 directly to a target system, the IE-78210 emulator probes can be purchased separately.

The EB-78210 features 32K bytes of static RAM for evaluation programs, an RS-232C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using the line assembler. Programs can be executed in real-time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general or special function registers, and to disassemble code.

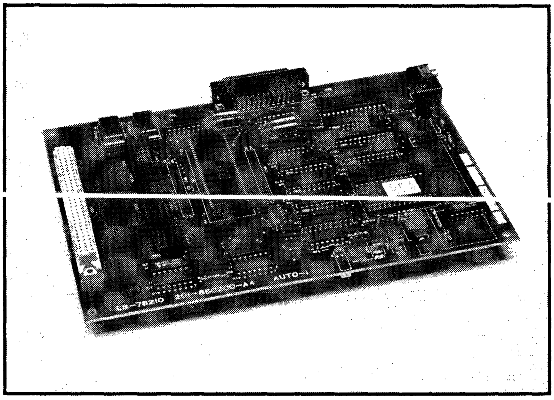
A controller program controls the EB-78210 directly from the console of an IBM PC®, PC/XT®, PC AT®, or compatible host computer using an RS-232C serial interface.

Features

- □ µPD78213 evaluation board
- □ 32K bytes of static RAM
- □ Real-time and single-step execution
- □ Four parallel or sequential breakpoints
- □ Display/change memory and general registers
- □ Display/change special function registers
- □ User program upload/download capability
- □ Symbolic debugging support
- □ Line assembler and disassembler
- □ RS-232C serial interface for host computer
- □ Host control software for IBM PC, PC/XT, PC AT, or compatibles
- □ Connection to a target system using in-circuit emulator probes

IBM PC, PC/XT and PC AT are registered trademarks of International Business Machines Corporation.

EB-78210

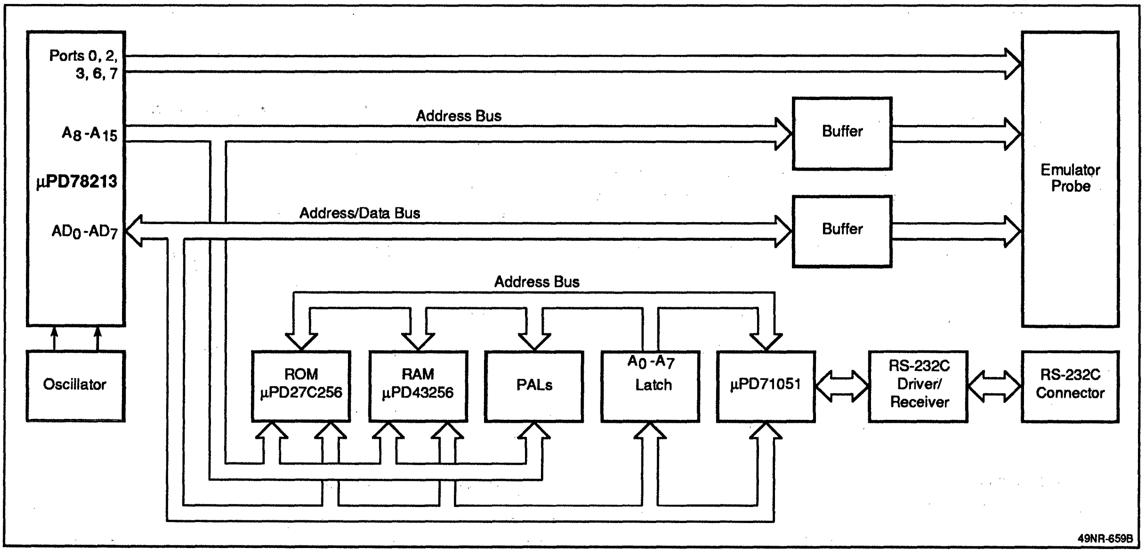


Ordering Information

Part Number	Description
EB-78210-PC	µD78213 evaluation board (IBM PC Based)
EP-78210CW-R	Emulator probe for 64-pin shrink DIP package (optional)
EP-78210GJ-R	Emulator probe for 74-pin QFP (optional)
EP-78210GQ-R	Emulator probe for 64-pin QUIP package (optional)
EP-78210L-R	Emulator probe for 68-pin PLCC package (optional)

9

Block Diagram



Hardware Description

The EB-78210 features 32K bytes of on-board static RAM. It can be used without a target system or can be directly connected to a target system using one of the IE-78210 emulation probes. When the EB-78210 is used without a target system, 28K bytes of RAM are available for downloading programs; the on-board monitor uses the remaining 4K bytes as a work area. When the EB-78210 is connected to a target system, 52K bytes of the µPD78213's 64K-byte code space are mapped to the target system. The extended µPD78213 data memory space (10000H to 0FFFFFH) is also mapped to the target system. A memory extension command specifies the high-order four bits of the address of the external extended data memory for use in the memory display commands.

The serial port for the host computer connection consists of a µPD71051 USART, an RS-232C driver/receiver, and a DB25 pin connector. A reset switch returns the EB-78210 to the power-up state. An AC/DC converter is shipped with each EB-78210 board for convenience. The EB-78210 can also be powered from batteries using the enclosed battery holder.

Emulation

The EB-78210 allows the following methods of program emulation: real-time program execution with or without breakpoints; real-time program execution for a specified number of instructions; single-step emulation for a specified number of instructions or until a register condition is satisfied. The registers, stack pointer, program status word, and program counter are displayed following termination of real-time emulation or during single-step emulation. The EB-78210 enters the single-step mode following a real-time emulation break. When the enter key is pressed during single-step emulation, the next instruction is executed and the executed address, instruction mnemonic and above data are displayed.

Emulation Accuracy

When the emulation probe is connected to a target system, ports 0, 2, 3, 6, 7, and the A/D converter related signals are identical to the device. However, all other signals differ from the actual device because of buffering and control gating.

Breakpoint Capabilities

The EB-78210 has four parallel instruction address breakpoints or up to a four-level sequential instruction address breakpoint. If any one of the four parallel breakpoints is satisfied, a break in emulation occurs. For a sequential breakpoint, each address must be encountered in the specified order before a break in emulation can occur. These breakpoints are set by substituting a software break instruction for an instruction in the user's program.

Software Description

The EB-78210 is controlled from the console of an IBM PC, PC/XT, PC AT, or compatible computer with an RS-232C interface using the enclosed emulator controller program. This program provides commands for downloading and uploading object code and symbol files to and from the EB-78210. A line assembler and disassembler avoid debugging in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation capability is available with the change register/memory commands. Initialization commands allow the user to choose a base number, register mnemonics, and extended data memory segment.

The EB-78210 program also has macro command file capability, so the user can execute a defined set of commands automatically. The on-line help facility, history command, and ability to store the console display on disk or send it to a printer ease debugging tasks.

Table 1 lists the available EB-78210 commands. These are a subset of the IE-78210 commands.

Table 1. Command List

Command	Function
ASM	Line assemble command
BRS	Sets instruction address breakpoints
COM	Creates command file
DAS	Disassemble command
DIR	Displays disk directory
EXP	Changes/displays high-order four bits of an address of the externally extended data memory
EXT	Terminates EB-78210 controller program operation
HIS	Displays last twenty commands
HLP	Displays format of commands
LOD	Loads object code and symbol files
LST	Sends console display to disk or printer
MAP	Dislays memory map
MAT	Evaluates arithmetic expression
MDR	Displays/modifies µPD78213 mode registers
MEM	Memory manipulation command
REG	Displays/modifies µPD78213 registers

Table 1. Command List (cont)

Command	Function
RES	Resets only the μPD78213
RGM	Changes/displays the implied or general register mode for the display of registers in a disassembler list
RUN	Executes programs in single-step mode or in real-time with options for break conditions
SAV	Saves contents of memory onto disk
SPR	Displays/modifies μPD78213 special function registers
STR	Automatically executes command string file
SUF	Base number specification (hex, octal, binary, decimal)
SYM	Adds/deletes/displays/changes/loads/saves symbols
VRY	Compares contents of an object file with memory

Equipment Supplied

The EB-78210-PC package consists of the following:

- EB-78210 evaluation board
- EB-78210 user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware
- Warranty policy and registration card

Documentation

For further information on EB-78210 operation, NEC Electronics Inc. provides the following manual with the board:

- EB-78210 μPD78213 Evaluation Board User's Manual

Additional copies may be obtained from NEC Electronics Inc.

NEC

NEC Electronics Inc.

EB-78220 Evaluation Board for the μPD78220

Description

The EB-78220 is an evaluation board for the NEC μPD78220 eight-bit single-chip microcomputer. The EB-78220 provides a simple way to evaluate the capabilities of the μPD78220 in an application without having to build a prototype. If it is necessary to connect the EB-78220 directly to a target system, the IE-78220 emulator probes can be purchased separately.

The EB-78220 features 32K bytes of static RAM for evaluation programs, an RS-232C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using the line assembler. Programs can be executed in real-time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general or special function registers, and to disassemble your code.

A controller program controls the EB-78220 directly from the console of an IBM PC®, PC/XT®, PC AT® or compatible host computer using an RS-232C serial interface.

Features

□ μPD78220 evaluation board

□ 32K bytes of static RAM

□ Real-time and single-step execution

□ Four parallel or sequential breakpoints

□ Display/change memory and general registers

□ Display/change special function registers

□ User program upload/download capability

□ Symbolic debugging support

□ Line assembler and disassembler

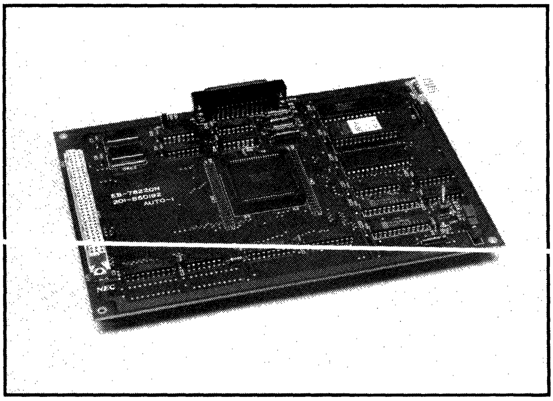
□ RS-232C serial interface for host computer

□ Host control software for IBM PC, PC/XT, PC AT, or compatibles

□ Connection to a target system using in-circuit emulator probes

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

EB-78220

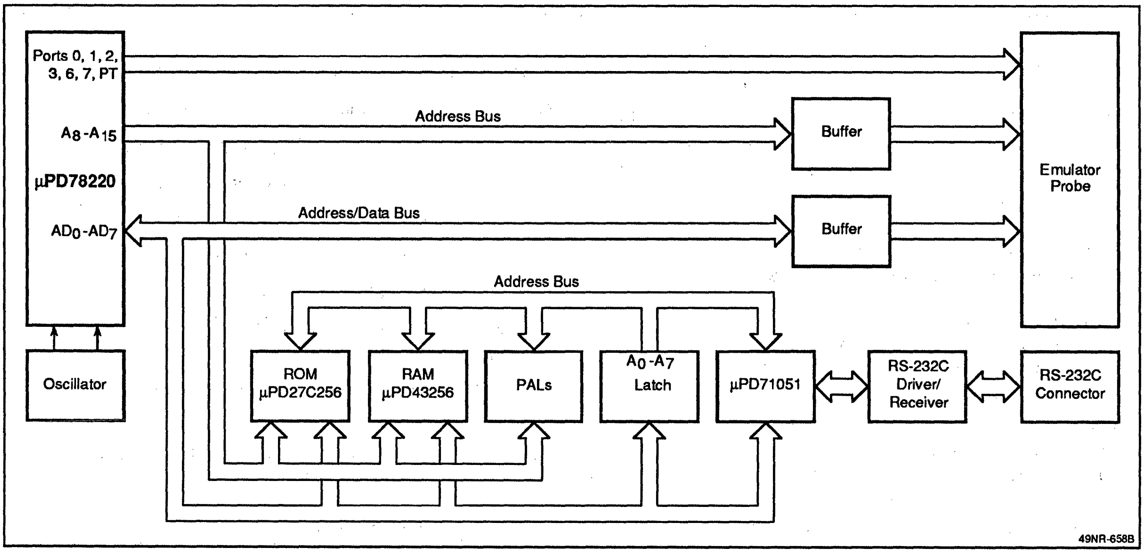


Ordering Information

Part Number	Description
EB-78220-PC	μPD78220 evaluation board (IBM PC Based)
EP-78220GJ-R	Emulator probe for 94-pin QFP (optional)
EP-78220L-R	Emulator probe for 84-pin PLCC package (optional)

9

Block Diagram



Hardware Description

The EB-78220 features 32K bytes of on-board static RAM. It can be used without a target system or can be directly connected to a target system using one of the IE-78220 emulation probes. When the EB-78220 is used without a target system, 28K bytes of RAM are available for downloading programs; the on-board monitor uses the remaining 4K bytes as a work area. When the EB-78220 is connected to a target system, 52K bytes of the μPD78220's 64K-byte code space are mapped to the target system. The extended μPD78220 data memory space (10000H to 0FFFFFH) is also mapped to the target system. A memory extension command specifies the high-order four bits of the address of the external extended data memory for use in the memory display commands.

The serial port for the host computer connection consists of a μPD71051 USART, an RS-232C driver/receiver, and a DB25 pin connector. A reset switch returns the EB-78220 to the power-up state. An AC/DC converter is shipped with each EB-78220 board for convenience. The EB-78220 can also be powered from batteries using the enclosed battery holder.

Emulation

The EB-78220 allows the following methods of program emulation: real-time program execution with or without breakpoints; real-time program execution for a specified number of instructions; single-step emulation for a specified number of instructions or until a register condition is satisfied. The registers, stack pointer, program status word, and program counter are displayed following termination of real-time emulation or during single-step emulation. The EB-78220 enters the single-step mode following a real-time emulation break. When the enter key is pressed during single-step emulation, the next instruction is executed and the executed address, instruction mnemonic and above data are displayed.

Emulation Accuracy

When the emulation probe is connected to a target system, ports 0, 1, 2, 3, 6, 7 and the analog comparators are identical to the device. However, all other signals differ from the actual device because of buffering and control gating.

Breakpoint Capabilities

The EB-78220 has four parallel instruction address breakpoints or up to a four-level sequential instruction address breakpoint. If any one of the four parallel breakpoints is satisfied, a break in emulation occurs. For a sequential breakpoint, each address must be encountered in the specified order before a break in emulation can occur. These breakpoints are set by substituting a software break instruction for an instruction in the user's program.

Software Description

The EB-78220 is controlled from the console of an IBM PC, PC/XT, PC AT, or compatible computer with an RS-232C interface using the enclosed emulator controller program. This program provides commands for downloading and uploading object code and symbol files to and from the EB-78220. A line assembler and disassembler avoid debuggining in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation capability is available with the change register/memory commands. Initialization commands allow the user to choose a base number, register mnemonics, and extended data memory segment.

The EB-78220 program also has macro command file capability, so the user can execute a defined set of commands automatically. The on-line help facility, history command, and ability to store the console display on disk or send it to a printer ease debugging tasks.

Table 1 lists the available EB-78220 commands. These are a subset of the IE-78220 commands.

Table 1. Command List

Command	Function
ASM	Line assemble command
BRS	Sets instruction address breakpoints
COM	Creates command file
DAS	Disassemble command
DIR	Displays disk directory
EXP	Changes/displays high-order four bits of an address of the externally extended data memory
EXT	Terminates EB-78220 controller program operation
HIS	Displays last twenty commands
HLP	Displays format of commands
LOD	Loads object code and symbol files
LST	Sends console display to disk or printer
MAP	Dislays memory map
MAT	Evaluates arithmetic expression
MDR	Displays/modifies μPD78220 mode registers
MEM	Memory manipulation command
REG	Displays/modifies μPD78220 registers
RES	Resets only the μPD78220

Table 1. Command List (cont)

Command	Function
RGM	Changes/displays the implied or general register mode for the display of registers in a disassembler list
RUN	Executes programs in single-step mode or in real-time with options for break conditions
SAV	Saves contents of memory onto disk
SPR	Displays/modifies µPD78220 special function registers
STR	Automatically executes command string file
SUF	Base number specification (hex, octal, binary, decimal)
SYM	Adds/deletes/displays/changes/loads/saves symbols
VRY	Compares contents of an object file with memory

Equipment Supplied

The EB-78220-PC package consists of the following:

- EB-78220 evaluation board
- EB-78220 user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware
- Warranty policy and registration card

Documentation

For further information on EB-78220 operation, NEC Electronics Inc. provides the following manual with the board:

- EB-78220 µPD78220 Evaluation Board User's Manual

Additional copies may be obtained from NEC Electronics Inc.

NEC

NEC Electronics Inc.

IE-78210 In-Circuit Emulator

Description

The IE-78210 is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78213 and μPD78214 single-chip microcomputers. Real-time and single-step emulation, inconjunction with sophisticated memory mapping features, breakpoints, and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

Features

□ Real-time and single-step emulation capability

□ User-specified breakpoints; logical OR of up to four sets of break conditions
- Opcode fetch count
- External sense clip condition
- Parallel or sequential fetch address break
- Logical AND of addresses, data values, CPU controls, and loop count

□ Sophisticated trace capabilities
- Traces program fetch or data access
- 2K x 44-bit trace buffer
- Address, control, data, and external signal trace features
- Instruction or frame display
- Trace search capability
- Trace display before or after specified break

□ Powerful memory mapping feature
- 64K bytes of RAM mappable in 128-byte blocks
- Up to 16K bytes of high-speed internal RAM for μPD78214 ROM emulation

□ Line assembler/disassembler

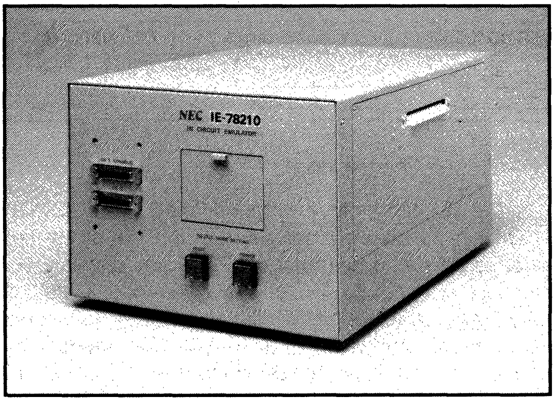
□ Symbolic debugging
- 2,000 symbols available
- IEEE-796 bus memory expansion slot for 32K additional symbols

□ CMOS latch-up warning and protection

□ Eight external sense clips on emulator probe

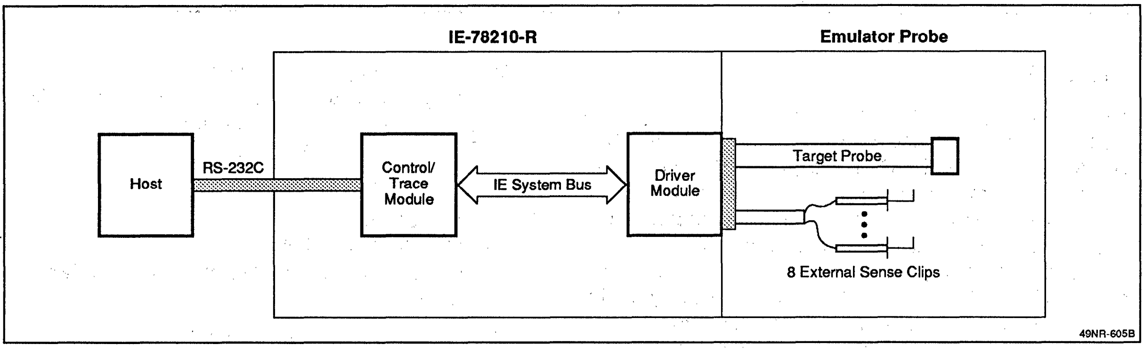
□ Stand-alone mode or system mode with host control program

IE-78210



IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

Block Diagram



Ordering Information

Part Number	Description
IE-78210-R	In-circuit emulator for µPD7821X
EP-78210CW-R	Emulator probe for 64-pin shrink DIP (optional)
EP-78210GJ-R	Emulator probe for 74-pin QFP (optional)
EP-78210GQ-R	Emulator probe for 64-pin QUIP (optional)
EP-78210L-R	Emulator probe for 68-pin PLCC (optional)

Hardware Description

As the IE-78210 block diagram shows, the IE-78210 hardware consists of a control/trace module, driver module, target probe, external sensing clips, and interconnecting system bus. The control/trace module includes the trace control unit, emulation memory unit, break control unit, and latch-up alarm unit. This module also houses the emulation CPU, which directly connects to the target emulation probe. The driver module houses the serial interface circuit, control CPU, trace RAM, and system memory.

Memory Mapping

The IE-78210 has a sophisticated memory mapping scheme which allows access of up to 64K bytes of internal memory, mappable in 128-byte units. The map command allocates the first 64K bytes of memory space of the emulation CPU either to the user system or to the IE system. Even if development of the target system is not complete, software debugging is still possible by using this internal RAM in place of the target system RAM. In addition to this emulation memory, the IE-78210 has an alternate high-speed memory for real-time emulation of the µPD78214 internal ROM. 4K, 8K, 12K, or 16K bytes of the high speed memory can be selected as internal ROM.

The extended data memory space of the emulation CPU (10000H to 0FFFFFH) is always mapped to the user system. A memory extension command is available to specify the high order four bits of the address of the external extended data memory for use in the memory display and break setting commands.

Emulation

The IE-78210 allows the following methods of program emulation: real-time program execution with or without breakpoints; real-time program execution for a specified number of instructions; and single-step emulation for a specified number of instructions or until a register condition is satisfied. Following termination of real-time emulation or during single-step emulation, the registers, stack pointer, program status word, and program counter are displayed. Following a real-time emulation break, the IE-78210 enters the single-step mode. Each time the enter key is pressed during single-step emulation, the next smallest group of instructions is executed and the above data is displayed.

Emulation Accuracy

Once a breakpoint is reached during emulation, the next few instructions are executed before breaking actually occurs. This is known as slip. The exact number of instructions slipped depends on the instructions in the prefetch queue and whether the emulation chip is accessing internal ROM or external memory. Ports 4 through 6 and the A/D converter related signals are identical to the devices. However, other signals differ from the actual device due to buffering and control gating.

Breakpoint Capabilities

The break function can be divided into two types: break register (physical and logical) breaks and fail-safe breaks. The user sets physical break registers to cause emulation breaks upon address, data, status or loop count; instruction count; parallel or sequential fetch addresses or matching a condition on an external sense clip. Combinations of these physical registers can then be set to the logical break registers and executed when running a break command. Fail-safe break conditions occur unconditionally and include manual break (ESC key or STP command in RUN N mode), non-mapped memory break, write-protected memory break, and SFR illegal access break.

Trace Capabilities

The IE-78210 has a 2K x 44-bit trace RAM for storing emulation data from each machine cycle. All fetch-related or data access-related addresses, data, CPU status signals and the eight external sense clips can be traced for up to 2,047 machine cycles. There are two types of trace displays: frame mode and instruction mode. In the frame mode display, the frame number and type, address and data information and external sense clip status are displayed for each frame in the order in which they are traced. In instruction mode, the executed instructions are displayed with their frame number, instruction address, mnemonics and operands. A number of trace display options are available. These include the display of all trace data, the display of only the frames meeting trace data search conditions, the display of five lines before or after frame meeting trace data search condition and the display of a specified number of lines following detection of the specified break register condition.

During real-time program emulation without breakpoints, the break condition can be used to stop the tracer a specified number of frames after the break condition is satisfied. At tracer stop time, the trace buffer can be viewed, new trace conditions set and the trace restarted while the program continues to execute in real-time.

CMOS Protection

The latch-up alarm circuit is activated when any CMOS IC in the driver module is in danger of being damaged by improper voltage levels on the pins. A protection circuit isolates the power supply to CMOS ICs and the message "emulation CPU latchup!" is displayed.

Utilities

The upload/download commands provide easy loading and saving of hex files to and from a host computer. The on-board assembler/disassembler allows the user to avoid programming in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation capability is available for the user with the change register/memory commands. Initialization commands allow the user to choose a clock source and a base number, and to define the system memory map.

System Mode

The IE-78210 can be connected to an IBM PC®, PC/XT®, PC AT®, or MD-086FD-10 by an RS232C port and operated in system mode. By using the accompanying control software, the debugging capabilities of the IE-78210 are greatly increased. It has a macro command file capability, allowing the user to execute a defined set of commands automatically. The on-line help facility, the history command, and the ability to store the console display on disk ease debugging tasks. The uploading/downloading capability can be utilized to upload and download both object code and symbol information. Other advantages are a verify command that compares memory to hex files, an alter symbol command, and a termination command for exiting to the operating system.

Table 1 lists commands available for both the stand-alone and system modes of the IE-78210. Commands listed in table 2 supplement table 1, but can only be used in the system mode.

Table 1. Stand-Alone and System Mode Commands

Command	Function
ASM	Line assemble command
BR?	Changes/displays breakpoint register used for stopping real-time emulation
CLK	Clock command (internal or external)
DAS	Disassemble command
DLY	Changes/displays trace frame count after trace trigger has been detected
EXP	Changes/displays high-order 4 bits of an address of the externally extended data memory
LOD	Loads hex format file into program memory
MAP	Memory mapping (64K bytes are accessible)
MAT	Performs arithmetic operation on an expression
MDR	Displays/modifies mode registers of emulator CPU
MEM	Memory manipulation command

Table 1. Stand-Alone and System Mode Commands (cont)

Command	Function
MOD	Sets channel two mode setting
MOV	Moves memory content to different mapping area
PGM	Performs PG series programmer from IE
REG	Displays/modifies registers of emulator CPU
RES	Resets IE-78210 and/or emulator CPU
RGM	Changes/displays the implied or general register mode for the display of registers in a disassembler list
RUN	Commences execution of emulator CPU in real-time with options for break conditions
SAV	Saves contents of hex memory onto disk
SPR	Displays/modifies special registers of emulator CPU
STP	Stops emulation CPU during normal emulation
SUF	Base number specification (hex, octal, binary, decimal)
SYM	Clears, displays, or changes a symbol
TRG	Starts real-time tracer during normal emulation
TR?	Changes/displays trace conditions for either real-time or single-step emulation
VRY	Compares contents of an object file with memory contents

Table 2. System Mode Only Commands

Command	Function
COM	Creates command file
DIR	Displays filenames
EXT	Terminates IE-78210 operation
HIS	Displays last twenty commands
HLP	Displays command format
LST	Stores console display on disk
STR	Automatically executes macro command file
SYM	Loads and saves symbol file

Equipment Supplied

The IE-78210-R package consists of the following:

- IE-78210 housing
- IE-78210-R user's manuals
- System disk for MD-086 series
- System disk for IBM PC
- AC power cable
- AC ground adapter
- Ground cable
- Spare fuse
- RS-232C interface cable
- Two 16-pin component carriers
- Warranty policy and registration card

Basic Specifications

- Weight: 10.5 kg
- External dimensions: length, 395 mm; width, 291 mm; height, 217 mm
- Power consumption: 100 V AC, 50/60 Hz, 5 A

Environmental Characteristics

- Operating temperature range: 10 to +40°C
- Storage temperature range: -20 to +45°C
- Ambient humidity range: 10 to 90% relative humidity

Documentation

For further information on IE-78210 operation, NEC Electronics Inc. provides the following manuals with the in-circuit emulator:

- IE-78210 μPD7821X In-Circuit Emulator Hardware Manual
- IE-78210 μPD7821X In-Circuit Emulator Software Manual
- IE78210 Controller Program User's Manual (IBM PC Based)

Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

IE-78220 In-Circuit Emulator

Description

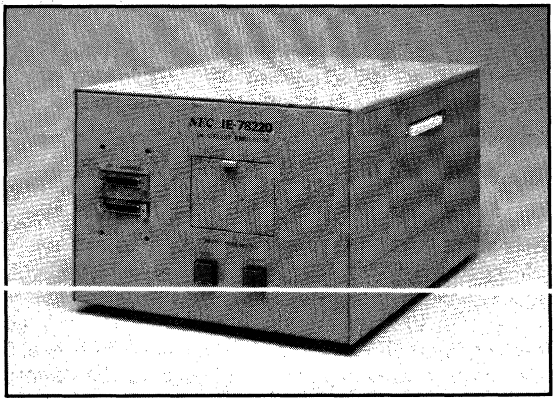
The IE-78220 is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78220 and μPD78224 single-chip microcomputers. Real-time and single-step emulation, in conjunction with sophisticated memory mapping features, breakpoints and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

Features

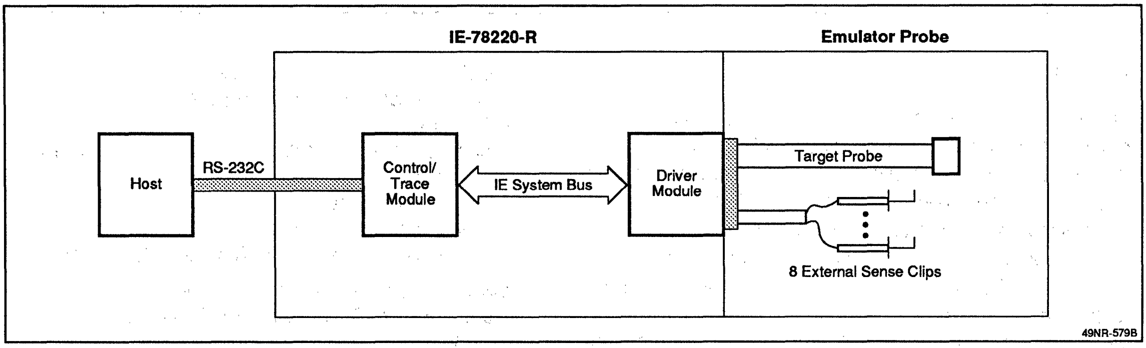
- Real-time and single-step emulation capability
- User-specified breakpoints; logical OR of up to four sets of break conditions
  - Opcode fetch count
  - External sense clip condition
  - Parallel or sequential fetch address break
  - Logical AND of addresses, data values, CPU controls, and loop count
- Sophisticated trace capabilities
  - Traces program fetch or data access
  - 2K x 44-bit trace buffer
  - Address, control, data, and external signal trace features
  - Instruction or frame display
  - Trace search capability
  - Trace display before or after specified break
- Powerful memory mapping feature
  - 64K bytes of RAM mappable in 128-byte blocks
  - Up to 16K bytes of high-speed internal RAM for μPD78224 ROM emulation
- Line assembler/disassembler
- Symbolic debugging
  - 2,000 symbols available
  - IEEE-796 bus memory expansion slot for 32K additional symbols
- CMOS latch-up warning and protection
- Eight external sense clips on emulator probe
- Stand-alone mode or system mode with host control program

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

IE-78220



Block Diagram



Ordering Information

Part Number	Description
IE-78220-R	In-circuit emulator for μPD78220/78224
EP-78220GJ-R	Emulator probe for 94-pin QFP (optional)
EP-78220L-R	Emulator probe for 84-pin PLCC (optional)

Hardware Description

As the IE-78220 block diagram shows, the IE-78220 hardware consists of a control/trace module, driver module, target probe, external sensing clips, and interconnecting system bus. The control/trace module includes the trace control unit, emulation memory unit, break control unit, and latch-up alarm unit. This module also houses the emulation CPU, which directly connects to the target emulation probe. The driver module houses the serial interface circuit, control CPU, trace RAM, and system memory.

Memory Mapping

The IE-78220 has a sophisticated memory mapping scheme which allows access to up to 64K bytes of internal memory, mappable in 128-byte units. The map command allocates the first 64K bytes of memory space of the emulation CPU either to the user system or to the IE system. Even if development of the target system is not complete, software debugging is still possible by using this internal RAM in place of the target system RAM. In addition to this emulation memory, the IE-78220 has an alternate high-speed memory for real-time emulation of the μPD78224 internal ROM. 4K, 8K, 12K, or 16K bytes of the high speed memory may be selected as internal ROM.

The extended data memory space of the emulation CPU (10000H to 0FFFFFH) is always mapped to the user system. A memory extension command is available to specify the high order four bits of the address of the external extended data memory for use in the memory display and break setting commands.

Emulation

The IE-78220 allows the following methods of program emulation: real-time program execution with or without breakpoints; real-time program execution for a specified number of instructions; and single-step emulation for a specified number of instructions or until a register condition is satisfied. Following termination of real-time emulation or during single-step emulation, the registers, stack pointer, program status word, and program counter are displayed. Following a real-time emulation break, the IE-78220 enters the single-step mode. Each time the enter key is pressed during single-step emulation, the next smallest group of instructions is executed and the above data is displayed.

Emulation Accuracy

Once a breakpoint is reached, during emulation, the next several instructions are executed before breaking actually occurs. This is known as slip. The exact number of instructions slipped depends on the instructions in the prefetch queue and whether the emulation chip is accessing internal ROM or external memory. Ports 4, 5, 6, and the T related signals are identical to the devices. However, other signals differ from the actual device due to buffering and control gating.

Breakpoint Capabilities

The break function can be divided into two types; break register (physical and logical) breaks, and fail-safe breaks. The user can set physical break registers to cause emulation breaks upon address, data, status, or loop count; instruction count; and parallel or sequential fetch addresses or matching a condition on an external sense clips. Combinations of these physical registers can then be set to the logical break registers and executed when running a break command. Fail-safe break conditions occur unconditionally and include manual break (ESC key or STP command in RUN N mode), non-mapped memory break, write-protected memory break, and SFR illegal access break.

Trace Capabilities

The IE-78220 has a 2K x 44-bit trace RAM for storing emulation data from each machine cycle. All fetch-related or data access-related addresses, data, CPU status signals and the eight external sense clips can be traced for up to 2,047 machine cycles. There are two types of trace displays: frame mode and instruction mode. In the frame mode display, the frame number and type, address and data information and external sense clip status are displayed for each frame in the order in which they are traced. In instruction mode, the executed instructions are displayed with their frame number, instruction address, mnemonics and operands. A number of trace display options are available. These include the displaying of all trace data, the displaying of only frames meeting trace data search conditions, the displaying of five lines before or after frame meeting trace data search condition, and the displaying of a specified number of lines following detection of the specified break register condition.

During real-time program emulation without breakpoints, the break condition can be used to stop the tracer a specified number of frames after the break condition is satisfied. At tracer stop time, the trace buffer can be viewed, new trace conditions set, and the trace restarted while the program continues to execute in real-time.

CMOS Protection

The latch-up alarm circuit is activated when any CMOS IC in the driver module is in danger of being damaged by improper voltage levels on the pins. A protection circuit isolates the power supply to CMOS ICs and the message "emulation CPU latchup !" is displayed.

Utilities

The upload/download commands provide easy loading and saving of hex files to and from a host computer. The on-board assembler/disassembler allows the user to avoid programming in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation capability is available for the user with the change register/memory commands. Initialization commands allow the user to choose a clock source and a base number, and to define the system memory map.

System Mode

The IE-78220 can be connected to an IBM PC®, PC/XT®, PC AT®, or MD-086FD-10 via an RS232C port and operated in system mode. By using the accompanying control software, the debugging capabilities of the IE-78220-R are greatly increased. It has a macro command file capability, allowing the user to execute a defined set of commands automatically. The on-line help facility, the history command, and the ability to store the console display on disk ease debugging tasks. The uploading/downloading capability can be utilized to upload and download both object code and symbol information. Other advantages are a verify command that compares memory to hex files, an alter symbol command, and a termination command for exiting to the operating system.

Table 1 lists commands available for both the stand-alone and system modes of the IE-78220. Commands listed in table 2 supplement table 1, but can be used only in the system mode.

Table 1. Stand-Alone and System Mode Commands

Command	Function
ASM	Line assemble command
BR?	Changes/displays breakpoint register used for stopping real-time emulation
CLK	Clock command (internal or external)
DAS	Disassemble command
DLY	Changes/displays trace frame count after trace trigger has been detected
EXP	Changes/displays high-order 4 bits of an address of the externally extended data memory
LOD	Loads hex format file into program memory
MAP	Memory mapping (64K bytes are accessible)
MAT	Performs arithmetic operation on an expression
MDR	Displays/modifies mode registers of emulator CPU
MEM	Memory manipulation command

Table 1. Stand-Alone and System Mode Commands (cont)

Command	Function
MOD	Sets channel two mode setting
MOV	Moves memory content to different mapping area
PGM	Performs PG series programmer from IE
REG	Displays/modifies registers of emulator CPU
RES	Resets IE-78220 and/or emulator CPU
RGM	Changes/displays the implied or general register mode for the display of registers in a disassembler list
RUN	Commences execution of emulator CPU in real-time with options for break conditions
SAV	Saves contents of hex memory onto disk
SPR	Displays/modifies special registers of emulator CPU
STP	Stops emulation CPU during normal emulation
SUF	Base number specification (hex, octal, binary, decimal)
SYM	Clears, displays, or changes a symbol
TRG	Starts real-time tracer during normal emulation
TR?	Changes/displays trace conditions for either real-time or single-step emulation
VRY	Compares the contents of an object file with memory contents

Table 2. System Mode Only Commands

Command	Function
COM	Creates command file
DIR	Displays file names
EXT	Terminates IE-78220 operation
HIS	Displays last twenty commands
HLP	Displays command format
LST	Stores console display on disk
STR	Automatically executes macro command file
SYM	Loads and saves symbol file

Equipment Supplied

The IE-78220-R package consists of the following:

- IE-78220 housing
- IE-78220-R user's manuals
- System disk for MD-086 series
- System disk for IBM PC
- AC power cable
- AC ground adapter
- Ground cable
- Spare fuse
- RS-232C interface cable
- Two 16-pin component carriers
- Warranty policy and registration card

Basic Specifications

- Weight: 10.5 kg
- External dimensions: length, 395 mm; width, 291 mm; height, 217 mm
- Power consumption: 100 V AC, 50/60 Hz, 5 A

Environmental Characteristics

- Operating temperature range: 10 to +40°C
- Storage temperature range: –20 to +45°C
- Ambient humidity range: 10 to 90% relative humidity

Documentation

For further information on the IE-78220 operation, NEC Electronics Inc. provides the following manuals with the in-circuit emulator:

- IE-78220 μPD7822X In-Circuit Emulator Hardware Manual
- IE-78220 μPD7822X In-Circuit Emulator Software Manual
- IE78220 Controller Program User's Manual (IBM PC Based)

Additional copies may be obtained from NEC Electronics Inc.

NEC

NEC Electronics Inc.

CC782XX
C Compiler Package for the μPD782XX Series

Description

The CC782XX C compiler package for the NEC μPD782XX microcomputers consists of an Kernighan and Ritchie compatible C cross compiler (CC210), relocatable assembler (RA210), linker (LK210), librarian (LB210), locater (LC210), and an emulator controller program. The CC782XX C compiler package is available for use on an MS-DOS® system with a free-standing system as the target (embedded system).

Features

- □ Kernighan and Ritchie standard C
  - — unsigned, enum, typedef, interrupt keywords
  - — extern, auto, static, register keywords
- □ Legal C code verification integrated into the compiler
- □ User-selectable and directable output files, list and full cross reference files
- □ Macro definitions
- □ Branch optimization
- □ Conditional assembly
- □ Simple diagnostics
- □ Powerful librarian

Ordering Information

Part Number	System	Description
CCMSD-I5DD-782XX	MS-DOS	5-1/4 inch double-density floppy diskette

C CROSS COMPILER (CC210)

Description

The CC210 C cross compiler converts standard C source code into relocatable object modules. The same relocatable object format is used for all relocatable object files in the C compiler package, regardless of how it is generated (by an assembler or compiler).

Compiler Options

The CC210 C compiler supports the following options during compilation:

- ● Integer size control
- ● Include file control
- ● Defining/undefining constants
- ● Prologue/epilogue control
- ● Forced stack checking before each C function
- ● Packed data allocation
- ● Special relocatable data segment
- ● Microprocessor type

C Library Functions

The CC210 C Compiler library includes most of the important C library functions that apply to PROM-based embedded systems. All library functions reside in the supplied library files. Header files that declare the set of library functions are also included.

The following character operation macros are available:

CHARACTER HANDLING <ctype.h>

Classification macros:

isalnum isalpha isascii iscntrl isdigit isgraph
islower isprint ispunct isspace isupper isxdigit

Conversion macros:

toascii tolower toupper

The following library functions are available:

NON-LOGICAL JUMPS <setjmp.h>

longjmp setjmp

FORMATTED INPUT/OUTPUT <stdio.h>

sscanf sprintf

GENERAL UTILITIES

crt0 (startup for C programs)
cipt (interrupt system support)
chkstk (check for stack overflow)

MS-DOS is a registered trademark of Microsoft Corporation.

STRING HANDLING <string.h>

strcat strchr strcmp strcpy strcspn strlen strncat strncmp strncpy strpbrk strrchr strspn strtok

MATHEMATICS

abs atoi atol

Memory Models

CC210 supports only the small memory model, since the μPD782XX series can only address a maximum of 64K bytes of program memory.

RELOCATABLE ASSEMBLER (RA210)

Description

RA210 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction is valid for the target μPD782XX microcomputer and produces a listing file and a relocatable object module.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Capability

RA210 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence differs from a subroutine call because the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler but are not translated into machine code during assembly. Basic assembler directives include: storage definition and allocation directives (DB,DW, DS, DBIT); symbol directives (EQU, SET); location counter control directive (ORG). Program control directives include: segment directives (CSEG, CSEG FIXED, CSEG CALLT, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); macro directives (MACRO, LOCAL, REPT, IRP, ENDM, EXITM); automatic BR instruction selection directive (BR) and assembly termination directive (END).

Assembler Controls

There are two types of assembler controls available for RA210. Primary controls specified in the assembler command line or at the beginning of the source module, are as follows:

- Processor selection
- Output object file selection
- Output list file selection
- Listing format controls
- Date specification
- Optimization selection
- Workfile drive selection
- Symbol letter case selection

General controls, specified in the source program, are as follows:

- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing subtitles
- Conditional assembly controls

LINKER (LK210)

LK210 combines multiple relocatable object modules and library modules and converts them into a single relocatable object module. The linker resolves PUBLIC/EXTRN references between modules, creating a relocatable output module that contains both relocatable object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK210 can be specified in either the command line or in a parameter file. Linker options include specifying the date and the absolute load module name, specifying the creation of a list file containing a link map, and specifying the letter case for symbols.

LOCATER (LC210)

LC210 converts a relocatable object module with no external references into an ASCII hexadecimal format absolute object code file. The locater outputs two files: an absolute load file in an expanded seven-bit ASCII hexadecimal format, which can be downloaded to a PROM programmer and a symbol file for the symbolic debugger. Locater options include specifying the starting address and order for code/data/stack segments, specifying areas of memory to be protected from being assigned, and specifying the creation of a map file with symbol tables.

LIBRARIAN (LB210)

LB210 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

The librarian creates and maintains library files containing relocatable object modules. Modules can be added to or deleted from a library file, or the contents of the library file can be listed.

EMULATOR CONTROLLER PROGRAM

Absolute object files produced by the CC782XX C compiler package can be debugged using the appropriate NEC stand-alone in-circuit emulator. NEC emulator controller programs allow you to communicate with the emulator through an RS-232C serial line. An emulator controller program is available to run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS. The emulator controller program provides the following features:

- Uploading/downloading of object/symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session
- Storage of last 20 commands for recall

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

LICENSE AGREEMENT

CC782XX is sold under terms of a license agreement, which is included with purchased copies of the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

DOCUMENTATION

For further information on source program formats, C compiler and assembler operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- CC78XXX C Compiler μPD78XXX C Compiler User's Manual
- CC78XXX C Compiler μPD78XXX Relocatable Assembler User's Manual

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

RA78K2
Relocatable Assembler Package for the µPD782XX Series

Description

The RA78K2 relocatable assembler package converts symbolic source code for the µPD782XX eight-bit single-chip microcomputers into executable absolute address object code. The RA78K2 relocatable assembler package consists of four separate programs: assembler (RA78K2), linker (LK78K2), locater (LC78K2), and librarian (LB78K2).

RA78K2 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcomputer specified at assembly time and produces a listing file and a relocatable object module.

LK78K2 combines multiple relocatable object modules and library modules and converts them into a single relocatable object module. LC78K2 converts a relocatable object module with no external references into an ASCII hexadecimal format absolute object code file.

LB78K2 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

Features

□ Absolute address object code output
□ User selectable and directable output files
□ Macro definitions
□ Branch optimization
□ Conditional assembly
□ Extensive error reporting
□ Powerful librarian
□ Runs under MS-DOS® and VAX®/VMS® operating systems

MS-DOS is a registered trademark of Microsoft Corporation. VAX and VMS are registered trademarks of Digital Equipment Corporation.

Ordering Information

Part Number	System	Description
RA78K2-D52	MS-DOS	5-1/4 inch double-density floppy diskette
RA78K2-VVT1	VAX/VMS	9-track 1600 BPI magnetic tape

Program Syntax

An RA78K2 source module consists of a series of code, data, or bit segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, and comment.

The symbol field may contain a label whose value is the instruction or data address or a name which represents an instruction address, data address, or constant. The mnemonic field may contain an instruction or assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Definition

RA78K2 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence is different than a subroutine call in that the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler but are not translated into machine code during assembly. Basic assembler directives include: storage definition and allocation directives (DB, DW, DS, DBIT); symbol directives (EQU, SET); and the location counter control directive ORG. Program control directives include: segment directives (CSEG, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); macro directives (MACRO, LOCAL, REPT, IRP, EXITM, ENDM); automatic BR instruction directive (BR); and assembly termination directive (END).

Assembler Controls

TheRA78K2 assembler (figure 1) has two types of controls . The primary controls, which are specified in the assembler command line or at the beginning of the source module, are as follows:

- Processor selection
- Output object creation selection
- Output list file selection
- Listing format controls
- Optimization selection
- Work file drive specification

The general controls, specified in the source program, are as follows:

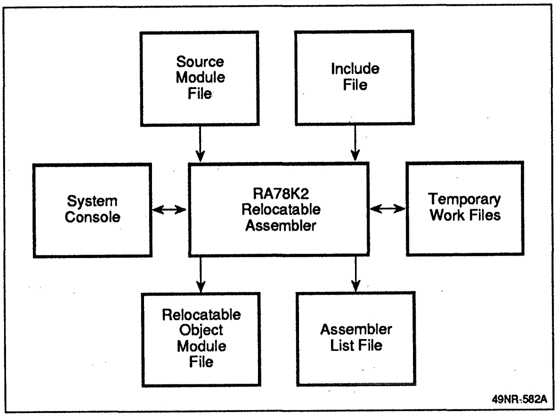
- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing titles
- Conditional assembly controls

The listing file may contain the complete assembly listing or only lines with errors, and a symbol or cross-reference table. The symbol table shows all defined symbols in alphabetical order, their types, attributes, and the values initially assigned to them.

The cross-reference table contains all defined symbols and the numbers of all statements that refer to them. The object file contains the relocatable object module. The format of this module is an NEC proprietary relocatable object module format.

If the optimization option is chosen, the assembler will generate the most efficient code by converting, wherever possible, three-byte absolute branches into two-byte relative branches.

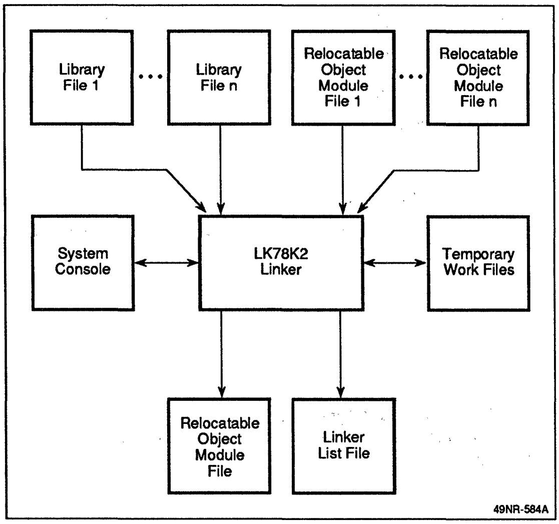
Figure 1. Relocatable Assembler Functional Diagram



Linker

The LK78K2 linker (figure 2) combines several relocatable object modules, resolving PUBLIC/EXTRN references between modules, to create a relocatable output module. This output module contains both relocatable object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK78K2 can be specified in either the command line or in a parameter file. The programmer can specify the date, the absolute load module name, and control the creation of a list file containing a link map.

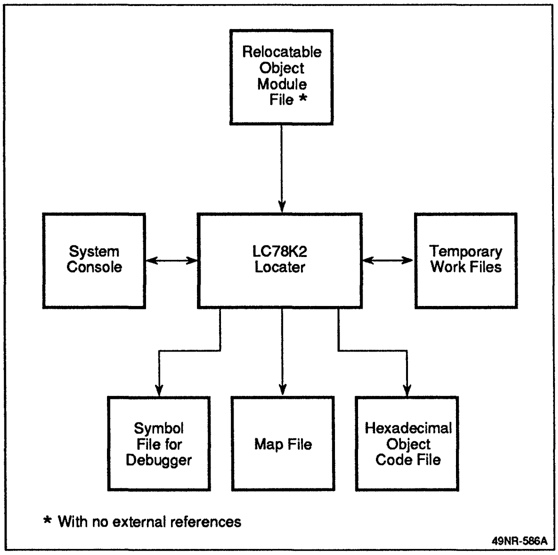
Figure 2. Linker Functional Diagram



Locater

The LC78K2 locater (figure 3), outputs two files: an absolute load file in an expanded hexadecimal format seven-bit ASCII, which can be downloaded to a PROM programmer; and a symbol file for the symbolic debugger. The programmer can specify the starting address and order for code/data/stack segments, and protect areas of memory from being assigned. The programmer can specify that a map file with symbol tables be created.

Figure 3. Locater Functional Diagram



* With no external references

Librarian

The LB78K2 librarian creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by storing several modules in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to, deleted from, or replaced within a library file, or the contents of the library file can be listed.

Operating Environment

The NEC RA78K2 package can run under a variety of operating systems. A version is available to run on a MS-DOS system with one or more disk drives and at least 128K of system memory. Another version is available to run on a Digital Equipment Corporation VAX computer under the VMS (Version 4.1 or later) operating system.

Emulator Controller Program

Absolute object files produced by the RA78K2 relocatable assembler package can be debugged by using the appropriate NEC stand-alone in-circuit emulator. NEC emulator controller programs allows communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provide the following features:

- Uploading and downloading of object and symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session
- Storage of last 20 commands for recall

License Agreement

RA78K2 is sold under terms of a license agreement, which is included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users.

Documentation

For further information on source program formats, assembler operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- RA78K2 μPD782XX Relocatable Assembler Package Language Manual
- RA78K2 μPD782XX Relocatable Assembler Package Operation Manual (MS-DOS)
- RA78K2 μPD782XX Relocatable Assembler Package Operation Manual (VMS)

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

NEC
NEC Electronics Inc.

ST78K2
Structured Assembler Preprocessor for the μPD782XX Series

Description

The ST78K2 structured assembler preprocessor is a companion program to the RA78K2 relocatable assembler for the NEC μPD782XX series of microcomputers. ST78K2 converts a source code file containing structured assembly statements into a pure assembly language source file, which can then be assembled with RA78K2.

ST78K2 converts a structured assembly statement into one or more μPD782XX assembly language instructions which perform the desired operation. Since ST78K2 only converts the structured assembly statements and does not convert μPD782XX assembly language instructions, a structured source program can include a combination of μPD782XX structured assembly statements and assembly language.

ST78K2 enables the assembly language programmer to use some of the structures and syntax of higher-level languages such as the C language. This improves program readability and reliability, and increases programmer productivity.

Features

□ Control structures for conditions, looping, and switch-case

□ Preprocessor directives for conditional code generation

□ C-like representation of comparison operations

□ C-like representation of assignment/arithmetic operations

□ Increment and decrement operators

□ Allow use of all μPD782XX mnemonics, registers, and features

□ Runs under MS-DOS® and VAX®/VMS® operating systems

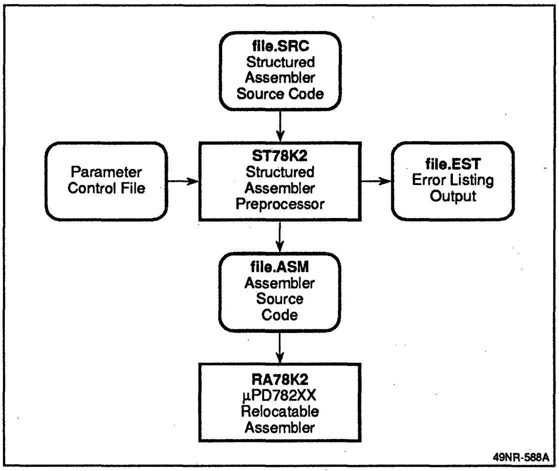
Ordering Information

The ST78K2 structured assembler preprocessor is included in the following software package at no cost:

- RA78K2 μPD782XX Relocatable Assembler Package

MS-DOS is a registered trademark of Microsoft Corporation.
VAX and VMS are registered trademarks of Digital Equipment Corporation.

Structured Assembler Preprocessor Functional Diagram



Summary Of Structured Language

A line of source code for ST78K2 contains either a structured assembly statement or a μPD782XX assembly language statement. μPD782XX assembly language statements (μPD782XX instructions, RA78K2 directives, or RA78K2 controls) pass through ST78K2 without change.

Structured assembly statements consist of preprocessor directives, assignment statements, and control statements. These statements are entered one per line, and are terminated by a line feed character. An optional comment may follow a semicolon at the end of the statement; all text following a semicolon is ignored by ST78K2.

Preprocessor directives cause ST78K2 to include or omit portions of code. Assignment statements cause ST78K2 to generate one or more μPD782XX assembly language instructions to alter the contents of a register or variable. Control statements cause ST78K2 to generate the necessary instructions to test conditions and change control flow based on those conditions.

Preprocessor Directives

ST78K2 preprocessor directives set and test variables, allowing conditional processing of code; include external files; and map instructions to μPD782XX CALT table reference instructions. Table 1 lists the preprocessor directives and their functions.

Table 1. Preprocessor Directives and Functions

Directive	Function
#define NAME value	Defines the variable NAME, set to the supplied value.
#ifdef ABC <statements> #else <statements> #endif	If ABC has been defined as above, or on the command line with the -D option, the first set of statements is processed and the second set ignored; if ABC has not been defined, or defined as zero, the first set of statements is ignored and the second set is processed.
#include "filename"	The named file is read from disk and processed as if included in the source.
#defcallt @LABEL CALL !label #endcallt	Whenever the instruction "CALL !label" is encountered in the source program, it is replaced by "CALLT [@LABEL]". The label must be defined in the CALLT table.

Assignment, Increment, And Decrement Statements

ST78K2 provides the ability to represent an assignment, or an assignment with an arithmetic operation, in C language syntax:

destination <assign-op> source

The assignment operators allow either simple assignment, or the combination of an assignment with an arithmetic operation on the source and destination.

Examples:

```
A = B       ;Move contents of B register to A
A + = [HL]  ;Add contents of memory at HL to A,
            ;store in A
```

Where an assignment requires an intermediate register to hold the value being assigned, the register is designated by naming it in parentheses following the assignment operation.

Examples:

```
DATA1 = B (A)     ;Store contents of B into memory at
                  ;DATA1, using A as temporary
                  ;storage
BC & = HL (XA)    ;and BC with HL, store in BC,
                  ;use XA as temp
```

The increment and decrement operators (+ + and --) operate on a single operand.

Table 2 lists the assignment operators with examples and functions.

Table 2. Assignment Operators with Examples and Functions

Operator	Example	Function
=	A = B	A ← B
< - >	A < - >B	Contents of A and B are exchanged
+ =	A + = B	A ← A + B
-=	A -= B	A ← A - B
*=	AX *= B	AX ← AX * B
/=	AX /= C	AX ← AX / C
&=	A &= B	A ← A & B (logical AND)
|=	A |= B	A ← A | B (logical OR)
^=	A ^= B	A ← A ^ B (logical XOR)
> > =	A> > =B	(CY←A_0,A_{n-1}←A_n,...,A_{max}←0) x B times
< < =	A< < =B	(CY←A_{max},A_{n+1}←A_n,...A_0←0) x B times
+ +	A+ +	A ← A + 1
--	A--	A ← A - 1

Control Statements

Control statements allow conditions to be tested. Based on the results of the test, blocks of code are allowed to be executed or skipped. Reserved words in the control statement define the start and end of blocks of code, and expressions to be evaluated.

Example:

```
if (A = = [HL])     ;The condition is tested
  P5 = B (A)        ;If A equals the content of memory
  A = [HL]          ;at HL, this code is executed
else
  A + = [HL]        ;Otherwise this code is executed
  A − = B
  P5 = A
endif
```

Table 3 shows the control statements and their functions.

Table 3. Control Statement Directives

Control Statement	Function
if - elseif - else - endif	Test variable expressions
if_bit - elseif_bit - else - endif	Test bit expressions
switch - case - default - ends	Select based on variable
for - next	Loop, test variable
while - endw	
repeat - until	
while_bit - endw	Loop, test bit
repeat - until_bit	
break	Exit control block
continue	Skip to top of block
goto LABEL	Branch to label

Variable And Bit Expressions

Variable expressions for tests consist of a single value, comparison between two variables, or a logical combination of comparisons. Bit expressions test individual bits. Table 4 shows examples of comparisons.

Table 4. Examples of Variable Expression Comparisons

Comparison	Meaning
if (A)	True if A is non-zero
if (A < B)	True if A is less than B
if ((A < B) && (A > C))	True if A is less than B and greater than C
if_bit (P3.2)	True if bit 2 of P3 is 1
if_bit (!P3.2)	True if bit 2 of P3 is 0

The allowable expressions using variables are shown in table 5.

Table 5. Expressions and Examples

Expression	Example
Primary	(A)
Term	(A < = B)
Term && Term	((A<B) && (A>C)) (logical AND)
Term || Term	((A= =C) || (A= =B)) (logical OR)

A primary value for a variable expression is a register name or defined symbol. A term consists of two primary values compared with a binary operator. Table 6 lists the supported binary operators and their meanings.

Table 6. Binary Operators

Binary Operator	Meaning
= =	Equals
!=	Not Equal
>	Greater Than
> =	Greater Than or Equal To
<	Less Than
< =	Less Than or Equal To

Bit expressions test individual bits of registers, ports, or memory locations. Table 7 shows the acceptable forms of bit expressions.

Table 7. Bit Expressions and Examples

Bit Expression	Example
Bit_primary	(P2.1)
!Bit_primary	(!CY)
Bit_primary && Bit_primary	(A.0 && CY)
Bit_primary || Bit_primary	(P2.2 || CY)

A Bit_primary can be either a reserved word bit identifier, such as a bit of a register or port (P2.1, CY), or a bit definition symbol (SB0 EQU P2.2).

ST78K2 Operation And Controls

ST78K2 is invoked by specifying the name of the source file, followed by optional controls.

Example:

```
C>ST78K2 ABC.SRC -DXYZ = 3
```

ST78K2 reads the specified source file and produces an output assembly language file, which can be input to RA78K2. The output file contains all lines provided in the input source file, plus those generated by ST78K2. Lines containing no statements for the structured assembler are passed through unchanged. Lines with structured assembly statements are placed in the output preceded by a semicolon. RA78K2 treats these lines as comments. These commented lines are then followed by the code generated by ST78K2.

The controls for ST78K2 are specified in the preprocessor command line or in a parameter file invoked in the command line. Table 8 lists the ST78K2 preprocessor controls and functions.

Table 8. ST78K2 Preprocessor Controls

Control	Function
-Ofilename	Specify name of output assembly source file
-Ffilename	Specify name of parameter file to be read
-Efilename	Specify name of error listing file
-Dsymbol[=value]	Define a symbol (like #define in code)
-I[d:][directory]	Define path for include file
-WTn1,n2,n3	Define TAB settings for generated code
-SCcharacter	Defines word symbol last character

The -O option allows the name of the output file to be specified. If not specified, the output file name defaults to the name of the input source file with the extension .ASM.

The -F option allows a parameter file to be specified, which will be read by ST78K2. This parameter file can contain a list of controls to be given to ST78K2, instead of or in addition to those specified on the command line.

The -E option specifies the name of the error listing file. The error file contains the file name, error number, description of error and the line containing the error. If the -E option is not specified, the error file name defaults to the name of the input source file with the extension .EST.

The -D control allows a symbol to be defined on the command line, with an optional value provided. If a symbol is defined but no value specified, the value defaults to 1. If the source file contains a #define directive which specifies a variable with the same name as the -D control, the value on the command line will override the value in the #define directive.

The -I control specifies a drive or directory other than the current drive and directory to search for include files.

The -WT control specifies the number of TAB characters to insert before labels, instruction mnemonics, and instruction operands generated by ST78K2. This allows clear separation of assembly language instructions coded in the source file from those generated by ST78K2.

The -SCcharacter control specifies the character used as the last character in a word symbol. The character must be a letter of the alphabet or the @, _ or ?. This allows ST78K2 to distingush between word and byte operations. Symbols which end in this character are treated as word symbols and will generate a word operation (ie. MOVW). If the -SC option is not specified, ST78K2 assumes that a symbol ending with the character "P" or "p" is a word symbol.

Documentation

For further information on source program formats, preprocessor operation, and actual program examples, NEC Electronics Inc. provides the following documentation.

- St78K2/ST78K3 μPD782xx/μPD783xx Structured Assembler Preprocessor User's Manual.

This documentation is provided with purchased copies of the RA78K2 μPD782xx relocatable assembler package. Additional copies can be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

DDK-78310A Evaluation Board for the µPD78310A

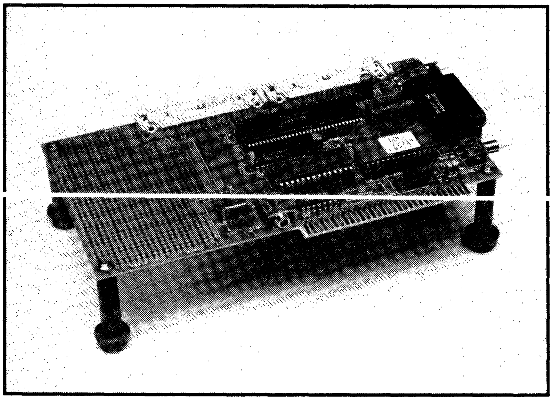
Description

The DDK-78310A is an evaluation board for the NEC µPD78310A eight/sixteen-bit, single-chip microcomputer. The DDK-78310A provides maximum flexibility when evaluating and designing with the µPD78310A. The DDK-78310A features 32K bytes of ROM, 32K bytes of RAM, RS-232C communication port, and a powerful monitor program. The DDK-78310A board is provided on an IBM PC® compatible card and includes a playpen area for building application specific hardware.

A copy of RA78K3, the µPD7831X/µPD7832X relocatable assembler for use on an IBM PC®, PC/XT®, PC AT®, or compatible host computer, is shipped with each DDK-78310A to allow development of code for evaluation purposes. Also included with the DDK-78310A is a emulator controller program for the IBM PC, a small demonstration program in ROM, the source code for the monitor, and a complete set of documentation. This total package provides a fast, efficient means for evaluating the capabilities of the µPD78310A for the user's application.

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

DDK-78310A Evaluation Board



Features

- □ µPD78310A evaluation board with power supply
- □ On-board memory:
  - — ROM: 32K-byte
  - — RAM: 32K-byte
- □ Powerful on-board debug monitor:
  - — Real-time and single-step operation
  - — Display/change memory and internal registers
  - — Disassembler
  - — Multiple software breakpoints
  - — User program download capability
- □ RS-232C serial interface for terminal or host computer
- □ Playpen area for user circuitry
- □ IBM PC card form factor
- □ RA78K3 µPD7831X/µPD7832X relocatable assembler package
- □ Host control software for IBM PC, PC/XT, PC/AT, or compatibles
- □ Demonstration program in ROM
- □ Source code for DDK-78310A monitor included

Ordering Information

Part Number	Description
DDK-78310A	µPD78310A evaluation board

Hardware Description

The DDK-78310A features 64K bytes of on-board memory. The lower 32K bytes are dedicated to ROM and include a powerful monitor program and user area. The upper 32K bytes are dedicated to RAM and include a user area for program downloading (7DFFH bytes), a monitor work area (7CFH bytes), and the internal RAM area of the μPD78310A (1FFH bytes).

The μPD78310A serial port is connected by an RS-232C driver/receiver to a DB25 pin connector. A reset switch allows the DDK-78310A to return to the power-up state without losing the contents of the external RAM. An NMI switch returns control from a user program to the monitor while saving the user's state.

An AC/DC converter provides power for the DDK-78310A in the stand-alone mode. The DDK-78310A can also receive power directly from the IBM PC bus.

Software Description

The DDK-78310A has a powerful interactive monitor to facilitate software design using the μPD78310A. A user program can be downloaded into user RAM and executed either in real-time with or without breakpoints or executed one instruction at a time. During single-stepping, the registers, program counter, and the next instruction to be executed are displayed.

The DDK-78310A has eight address breakpoints. The user can set up to seven of these prior to program execution. The eighth breakpoint is reserved for use in the GO command line. The monitor sets a breakpoint by substituting a software break instruction (opcode 5EH) for an instruction in the user's program.

Additional commands are available to:

- Display, fill, change, or move memory
- Display or change registers
- Disassemble memory
- Display the command list
- Place the interrupt vector and call table areas at 0H or 8000H

Table 1 contains a complete list of the DDK-78310A monitor commands and their syntax.

Block Diagram

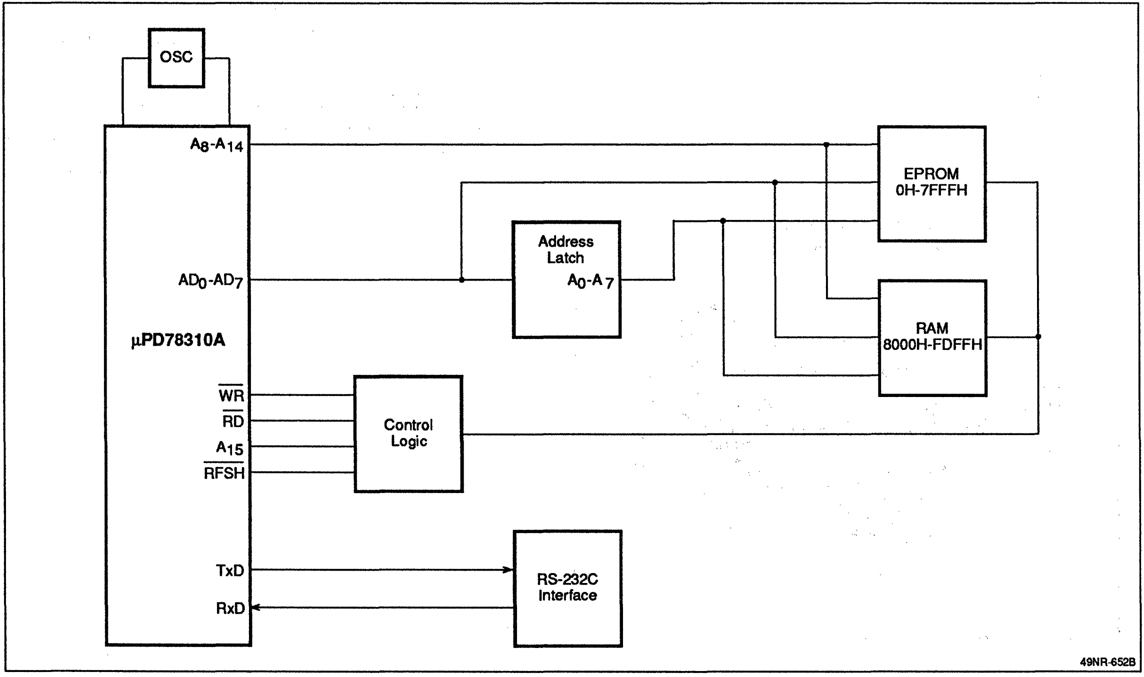


Table 1. Command List

Command	Function	Syntax
?/H	Show this menu of commands	?
B	Show or set breakpoints	B[bp,addr]
C	Change memory byte	C[addr][,val]
D	Display memory	D[addr][,addr]
F	Fill memory	Faddr,addr,val
G	Go (to breakpoint)	G[addr][,addr]
I	Move interrupt vectors to/from 8000H	I
K	Kill breakpoint(s)	K[bp]
L	Load a HEX file on to the DDK-78310A	L[addr]
M	Move a block of memory	Maddr,addr,addr
R	Display/change registers	R[reg]
T	Trace execution	T[addr]
U	Unassemble a block of memory	U[addr][,addr]

Notes:

(1) addr = 16-bit address in hexadecimal format.

(2) bp = breakpoint number, 1-7.

(3) reg = general purpose or control register mnemonic.

(4) val = eight-bit value in hexadecimal notation.

(5) [ ] = optional parameter.

RA78K3 Relocatable Assembler Package

The RA78K3 relocatable assembler package converts symbolic source code for the μPD7831X and μPD7832X eight/sixteen-bit, single-chip microcomputers into executable absolute address object code. A copy of RA78K3 is included with the DDK-78310A to use with an IBM PC, PC/XT, PC AT, or compatible. Evaluation programs for the μPD78310A can be written easily with this software.

Emulator Controller Program

Absolute address object files produced by the RA78K3 relocatable assembler package can be downloaded to the DDK-78310A using the NEC emulator controller program, supplied with the DDK-78310A. This controller program allows files to be downloaded from an IBM PC or compatible to the DDK-78310A board. In addition to downloading files, the NEC emulator controller program provides these additional capabilities:

- Complete DDK-78310A control from host console
- On-line help facilities
- Host system directory and file display
- Storage of debug session on disk

License Agreement

RA78K3 is provided under the terms of a license agreement included with the DDK-78310A board. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided to registered users.

Documentation

For further information on the DDK-78310A evaluation board, NEC Electronics Inc. provides the following manual:

- DDK-78310A μPD78310A Evaluation Board User's Manual

This manual is provided with the board. Additional copies can be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

EB-78320 Evaluation Board for the μPD78320

Description

The EB-78320 is an evaluation board for the NEC μPD78320 eight-bit, single-chip microcomputer. The EB-78320 provides a simple way to evaluate the capabilities of the μPD78320 in an application without having to build a prototype. If it is necessary to connect the EB-78320 directly to a target system, the IE-78320 emulator probes can be purchased separately.

The EB-78320 features 32K bytes of static RAM for evaluation programs, an RS-232C communication port, and a powerful on-board monitor. Evaluation programs can be downloaded from a host computer or created directly on the board using the line assembler. Programs can be executed in real-time with or without breakpoints or one instruction at a time. Commands are available to display or change memory, general or special function registers, and to disassemble code.

A controller program controls the EB-78320 directly from the console of an IBM PC®, PC/XT®, PC AT®, or compatible host computer using an RS-232C serial interface.

Features

□ μPD78320 evaluation board

□ 32K bytes of static RAM

□ Real-time and single-step execution

□ Four parallel or sequential breakpoints

□ Display/change memory and general registers

□ Display/change special function registers

□ User program upload/download capability

□ Symbolic debugging support

□ Line assembler and disassembler

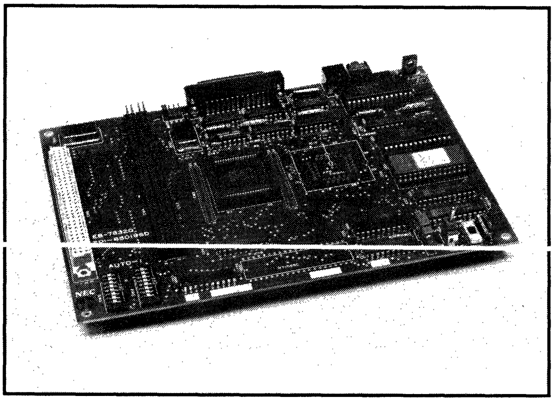
□ RS-232C serial interface for host computer

□ Host control software for IBM PC, PC/XT, PC AT, or compatibles

□ Connection to a target system using in-circuit emulator probes

IBM PC, PC/XT and PC AT are registered trademarks of International Business Machines Corporation.

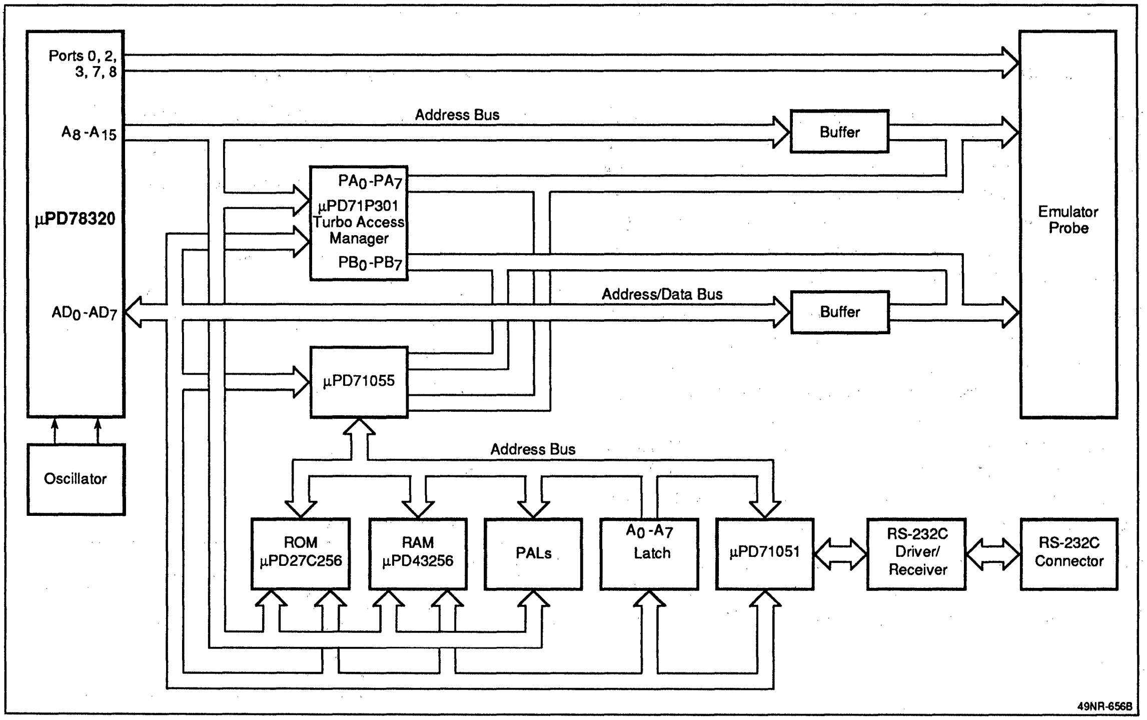
EB-78320



Ordering Information

Part Number	Description
EB-78320-PC	μPD78320 evaluation board (IBM PC Based)
EP-78320GJ-R	Emulator probe for 74-pin QFP (optional)
EP-78320L-R	Emulator probe for 68-pin PLCC package (optional)

Block Diagram



Hardware Description

The EB-78320 features 32K bytes of on-board static RAM. It can be used without a target system or can be directly connected to a target system using one of the IE-78320 emulation probes. When the EB-78320 is used without a target system, 28K bytes of RAM are available for downloading programs; the on-board monitor uses the remaining 4K bytes as a work area. When the EB-78320 is connected to a target system, 52K bytes of the μPD78320's 64K-byte code space are mapped to the target system.

The EB-78320 can be used to evaluate the instruction execution speed of the μPD78322's internal ROM by installing a μPD71P301 turbo access manager in the footprint on the board. When using the μPD71P301, the evaluation program is placed in the EPROM of the μPD71P301; the emulation function of the EB-78320 board is not available.

The serial port for the host computer connection consists of a μPD71051 USART, an RS-232C driver/receiver and a DB25 pin connector. A reset switch returns the EB-78320 to the power-up state. An ac/dc converter is shipped with each EB-78320 board for convenience. The EB-78320 can also be powered from batteries using the enclosed battery holder.

Emulation

The EB-78320 allows the following methods of program emulation: real-time program execution with or without breakpoints; real-time program execution for a specified number of instructions; single-step emulation for a specified number of instructions or until a register condition is satisfied. The registers, stack pointer, program status word, and program counter are displayed following termination of real-time emulation or during single-step emulation. The EB-78320 enters the single-step mode following a real-time emulation break. When the enter key is pressed during single-step emulation, the next instruction is executed, and the executed address, instruction mnemonic and above data are displayed.

Emulation Accuracy

When the emulation probe is connected to a target system, ports 0, 2, 3, 7, and 8, the watchdog timer output and the A/D converter related signals are identical to the device. However, all other signals differ from the actual device because of buffering and control gating.

Breakpoint Capabilities

The EB-78320 has four parallel instruction address breakpoints or up to a four-level sequential instruction address breakpoint. If any one of the four parallel breakpoints is satisfied, a break in emulation occurs. For a sequential breakpoint, each address must be encountered in the specified order before a break in emulation can occur. These breakpoints are set by substituting a software break instruction for an instruction in the user's program.

Software Description

The EB-78320 is controlled from the console of an IBM PC, PC/XT, PC AT, or compatible computer with an RS-232C interface using the enclosed emulator controller program. This program provides commands for downloading and uploading object code and symbol files to and from the EB-78320. A line assembler and disassembler avoid debugging in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation capability is available with the change register/memory commands. Initialization commands choose a base number and register mnemonics.

The EB-78320 program also has macro command file capability, so the user can execute a defined set of commands automatically. The on-line help facility, history command, and ability to store the console display on disk or send it to a printer ease debugging tasks.

Table 1 lists the available EB-78320 commands. These are a subset of the IE-78320 commands.

Table 1. Command List

Command	Function
ASM	Line assemble command
BRS	Sets instruction address breakpoints
COM	Creates command file
DAS	Disassemble command
DIR	Displays disk directory
EXT	Terminates EB-78320 controller program operation
HIS	Displays last twenty commands
HLP	Displays format of commands
LOD	Loads object code and symbol files
LST	Sends console display to disk or printer
MAP	Dislays memory map
MAT	Evaluates arithmetic expression
MDR	Displays/modifies μPD78320 mode registers
MEM	Memory manipulation command
REG	Displays/modifies μPD78320 registers
RES	Resets only the μPD78320

Table 1. Command List (cont)

Command	Function
RUN	Executes programs in single-step mode or in real-time with options for break conditions
SAV	Saves contents of memory onto disk
SPR	Displays/modifies μPD78320 special function registers
STR	Automatically executes command string file
SUF	Base number specification (hex, octal, binary, decimal)
SYM	Adds/deletes/displays/changes/loads/saves symbols
VRY	Compares contents of an object file with memory

Equipment Supplied

The EB-78320-PC package consists of the following:

- EB-78320 evaluation board
- EB-78320 user's manual
- System disk for IBM PC
- AC/DC converter power supply
- Battery holder and mounting hardware
- Warranty policy and registration card

Documentation

For further information on EB-78320 operation, NEC Electronics Inc. provides the following manual with the board:

- EB-78320 μPD78320 Evaluation Board User's Manual

Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

IE-78310A
In-Circuit Emulator

Description

The IE-78310A is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78310A and μPD78312A single-chip microcomputers. Real-time and single-step emulation, in conjunction with sophisticated memory mapping features, breakpoints, and trace capabilities, create a powerful debugging environment. A line assembler/disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

Features

□ Real-time and single-step emulation capability

□ User-specified breakpoints
— Logical OR of up to four sets of break conditions
Opcode fetch count
External sense clips condition
Emulation time
Logical AND of addresses, data values, CPU controls, and loop count

□ Sophisticated trace capabilities
— Instruction, frame, or macro service display
— 2K x 44-bit trace buffer
— Address, control, data, and port trace features

□ Powerful memory mapping feature
— 64K bytes of RAM mappable in 256-byte blocks
— Up to 16K bytes of high-speed internal RAM for μPD78312A ROM emulation

□ Line assembler/disassembler

□ Symbolic debugging
— 2,000 symbols available
— IEEE-796 bus memory expansion slot for 32K additional symbols

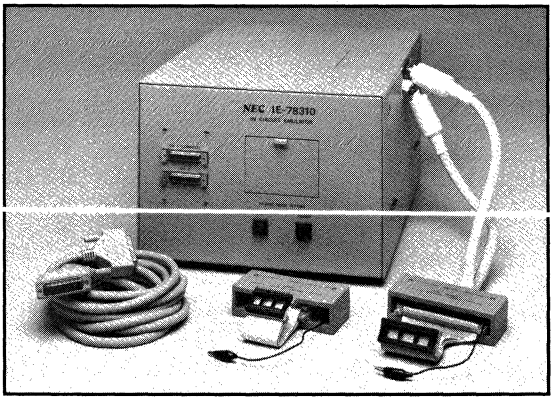
□ CMOS latch-up warning and protection

□ Eight external sense clips

□ Self-diagnostic command

□ Stand-alone mode or system mode with host control program

IE-78310A with Emulator Probe



Ordering Information

Part Number	Description
IE-78310A-R	In-circuit emulator for μPD78310A/μPD78312A
EP-78310CW	Emulator probe for 64-pin shrink DIP package(shipped with IE-78310A)
EP-78310GQ	Emulator probe for 64-pin QUIP package(shipped with IE-78310A)
EP-78310L	Emulator probe for 68-pin PLCC package (optional)
EP-78310GF	Emulator probe for 64-pin QFP package (optional)

Hardware Description

As the IE-78310A block diagram shows, the IE-78310A hardware consists of a control/trace module, driver module, target probe, external sense clips, and interconnecting system bus. The control/trace module includes the trace control unit, emulation memory unit, self-diagnostic unit, break control unit, and latch-up alarm unit. This module also houses the emulation CPU, which is directly connected to the target emulation probe. The driver module houses the serial interface circuit, control CPU, trace RAM, and system memory.

Memory Mapping

The IE-78310A has a sophisticated memory mapping scheme which allows access of up to 64K bytes of internal memory, mappable in 256-byte units. The map command allocates the memory space of the emulation CPU either to the user system or to the IE system. Even if development of the target system is not complete, software debugging is still possible by using this internal RAM in place of the target system RAM. In addition to this emulation memory, the IE-78310A has an alternate high-speed memory for real-time emulation of the μPD78312A internal ROM. 0, 4K, 8K, or 16K bytes of the high speed memory can be selected as internal ROM.

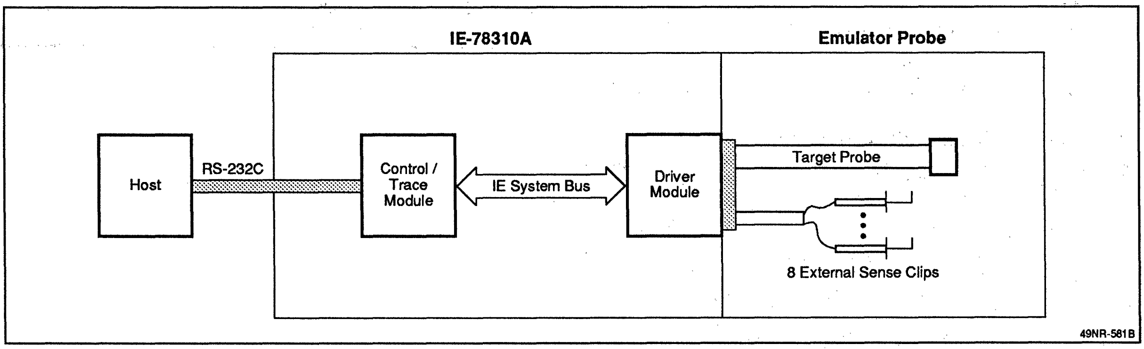
Emulation

The IE-78310A allows the following methods of program emulation: real-time program execution with or without breakpoints; real-time program execution for a specified number of instructions; and single-step emulation for a specified number of instructions or until a register condition is satisfied. Following termination of real-time emulation or during single-step emulation, the registers, stack pointer, program status word, and program counter are displayed. Following a real-time emulation break, the IE-78310A enters the single-step mode. Each time the space bar is pressed during single-step emulation, the next smallest group of instructions is executed and the above data is displayed.

Emulation Accuracy

Once a breakpoint is reached during emulation, the next few instructions are executed before breaking actually occurs. This is known as slip. The exact number of instructions slipped depends on the instructions in the prefetch queue and whether the emulation chip is accessing internal ROM or external memory. Ports 0, 2, 3, the A/D, and the refresh signals are identical to the μPD78310A/μPD78312A. However, other signals differ from the actual device due to buffering and control gating.

Block Diagram



Self-Diagnostics

A self-diagnostic command monitors the IE-78310A for error-free operation. It checks alternate RAM, user RAM, address/data bus, the 64-pin probe, the emulation chip (including all port lines), and both the EA/Vpp and reset lines.

Breakpoint Capabilities

The break function can be divided into three types: break register (physical and logical) breaks, command breaks, and fail-safe breaks. The user sets physical break registers to cause emulation breaks upon address, data, status or loop count; instruction count; timer (1 to 65,535 ms range) or matching a set of conditions on the eight external sense clips. Combinations of these physical registers can then be set to the logical break registers and executed when running a break command. Command breaks are set in the emulation command and can cause emulation breaks after a specified number of steps are executed or a register condition is satisfied. Fail-safe break conditions occur unconditionally and include manual break (ESC key), non-mapped memory break, write-protected memory break, and reset break.

Trace Capabilities

The IE-78310A has a 2K x 44-bit trace RAM for storing emulation data from each machine cycle. Given a user-specified range, trace can be performed upon address, data, frame status signals ($\overline{RD}$, $\overline{WR}$, MSRD, MSWR, OP, M1), ports P0 to P5, and the eight external sense clips for up to 2,047 machine cycles. There are three types of trace displays: frame mode, macro service mode, and instruction mode. In frame mode display, the frame number and type, address and data information and port and external sense clip status are displayed for each frame in the order in which they are traced. In instruction mode, the executed instructions are displayed with their frame number, instruction address, mnemonics and operands. In macro service mode, reads/writes of the macro service routines are added into the instruction mode display.

CMOS Protection

The latch-up alarm circuit is activated when any CMOS IC in the driver module is in danger of being damaged by improper voltage levels on the pins. A protection circuit isolates the power supply to CMOS ICs and the message "emulation CPU latchup !" is displayed.

Utilities

The upload/download commands provide easy loading and saving of hex files to and from a host computer. The on-board assembler/disassembler allows the user to avoid programming in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation capability is available for the user with the change register/memory commands. Initialization commands allow the user to choose a clock source and a base number, and to define the system memory map.

System Mode

The IE-78310A can be connected to an IBM PC®, PC/XT®, PC AT®, or MD-086FD-10 by an RS232C port and operated in system mode. By using the accompanying control software, the debugging capabilities of the IE-78310A-R are greatly increased. It has a macro command file capability, allowing the user to execute a defined set of commands automatically. The on-line help facility, the history command, and the ability to store the console display on disk ease debugging tasks. The uploading/downloading capability can be utilized to upload and download both object code and symbol information. Other advantages are a verify command that compares memory to hex files, an alter symbol command, and a termination command for exiting to the operating system.

Table 1 lists commands available for both the stand-alone and system modes of the IE-78310A. Commands listed in table 2 supplement table 1, but can only be used in the system mode.

IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

Table 1. Stand-Alone and System Mode Commands

Command	Function
ASM	Line assemble command
BR?	Changes/displays breakpoint register used for stopping real-time emulation
CLK	Clock command (internal or external)
DAS	Disassemble command
DIG	Self-diagnostic command
LOD	Loads hex format file into program memory
MAP	Memory mapping (64K bytes are accessible)
MDR	Displays/modifies mode registers of emulator CPU
MEM	Memory manipulation command
MOD	Sets channel two mode setting
MOV	Moves memory content to different mapping area
REG	Displays/modifies registers of emulator CPU
RES	Resets IE-78310A and/or emulator CPU
RUN	Commences execution of emulator CPU in real-time with options for break conditions
SAV	Saves contents of hex memory onto disk
SPR	Displays/modifies special registers of emulator CPU
SUF	Base number specification (hex, octal, binary, decimal)
SYM	Clears, displays, or changes a symbol
TR?	Changes/displays trace conditions for either real-time or single-step emulation
VRY	Compares memory and hex files

Table 2. System Mode Only Commands

Command	Function
COM	Creates command file
DIR	Displays filenames
EXT	Terminates IE-78310A controller program operation
HIS	Displays last twenty commands
HLP	Displays command format
LST	Stores console display on disk
STR	Automatically executes macro command file
SYM	Loads and saves symbol file

Equipment Supplied

The IE-78310A-R package consists of the following:

- IE-78310A housing
- Target probe cable
- Target probe unit for 64-pin shrink DIP socket (EP-78310CW)
- Target probe unit for 64-pin QUIP socket (EP-78310GQ)
- External sense clips
- IE-78310A-R user's manuals
- System disk for MD-086 series
- System disk for IBM PC
- AC power cable
- AC ground adapter
- Ground cable
- Spare fuse
- RS-232C interface cable
- Two 16-pin component carriers
- Warranty policy and registration card

Basic Specifications

- Weight: 10.5 kg
- External dimensions: length, 395 mm; width, 291 mm; height, 217 mm
- Power consumption: 100 V AC, 50/60 Hz, 5 A

Environmental Characteristics

- Operating temperature range: 10 to +40°C
- Storage temperature range: –20 to +45°C
- Ambient humidity range: 10 to 90% relative humidity

Documentation

For further information on IE-78310A operation, NEC Electronics Inc. provides the following manuals with the in-circuit emulator:

- IE-78310A μPD7831XA In-Circuit Emulator Hardware User's Manual
- IE-78310A μPD7831XA In-Circuit Emulator Software User's Manual
- IE78310A Controller Manual (IBM PC Based)
- IE-78310A Sample Session (IBM PC Based)

Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

IE-78320 In-Circuit Emulator

Description

The IE-78320 is an in-circuit emulator providing both hardware emulation and software debugging capabilities for the NEC μPD78320 and μPD78322 single-chip microcomputers. Real-time and single-step emulation, combined with sophisticated memory mapping features, breakpoints and trace capabilities, create a powerful debugging environment. A line assembler and disassembler, full register and memory control, symbolic debugging, and complete upload/download capabilities simplify the task of debugging hardware and software.

Features

□ Real-time and non-real-time emulation

□ User-specified breakpoints
— Logical OR of up to four sets of break conditions
— Executed instruction count
— External sense clip number one condition
— Parallel or sequential instruction address break
— Logical AND of addresses, data values, CPU status, loop count, and external sense clip data for either the main or internal CPU bus

□ Sophisticated trace capabilities
— Traces main and internal CPU bus activity or main bus and external sense clip activity
— 2K x 44-bit trace buffer
— Instruction, instruction with macro service, or frame display
— Trace search capability
— Trace display before or after specified break

□ Powerful memory mapping
— Up to 56K bytes of RAM for internal ROM, turbo access manager memory, or off-chip memory emulation
— Mappable in 8K-byte blocks

□ Emulation timer and instruction counter

□ Line assembler/disassembler

□ Symbolic debugging
— 7,000 symbols available

□ CMOS latch-up warning and protection

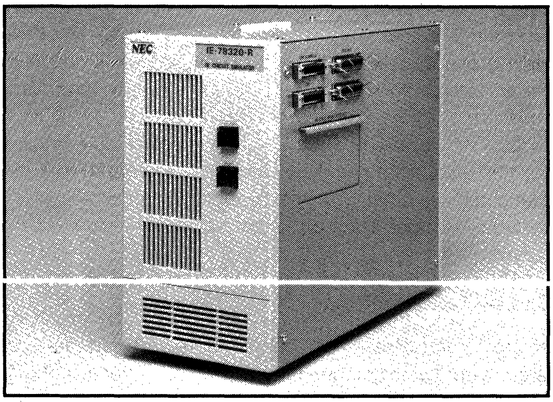
□ Eight external sense clips on emulator probe

□ Stand-alone or system mode with host control program

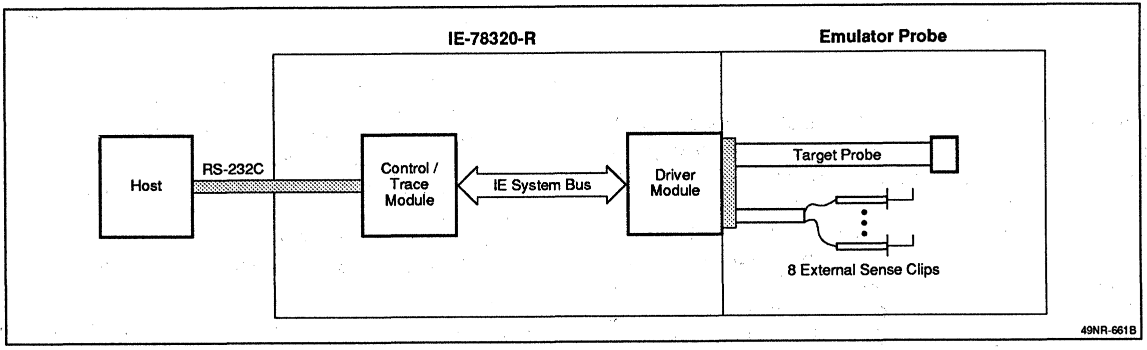
□ Centronics parallel interface for optional high-speed download

MS-DOS is a registered trademark of Microsoft Corporation.
IBM PC, PC/XT, and PC AT are registered trademarks of International Business Machines Corporation.

IE-78320



Block Diagram



Ordering Information

Part Number	Description
IE-78320-R	In-circuit emulator for µPD78320 and µPD78322
EP-78320GJ-R	Emulator probe for 74-pin QFP (optional)
EP-78320L-R	Emulator probe for 68-pin PLCC package (optional)

Hardware Description

The IE-78320 hardware consists of a control/trace module, driver module, target probe, external sense clips, and the interconnecting system bus. The control/trace module includes the trace control unit, emulation memory unit, break control unit, and the latch-up alarm unit. The control/trace module also houses the emulation CPU, which is connected directly to the target emulation probe. The driver module houses the serial and parallel interface circuits, trace RAM, control CPU, and system memory.

Memory Mapping

The IE-78320 incorporates a sophisticated memory mapping scheme which allows the 64K bytes of microcomputer memory space to be mapped to internal or external memory in 8K-byte units. Even if development of the target system is not complete, software debugging is possible by using internal RAM in place of the target system RAM or ROM.

The first 56K bytes of memory space can be emulated in the in-circuit emulator as internal on-chip ROM, turbo access manager (µPD71P301) memory, off-chip memory (RAM) or write-protected off-chip memory (ROM); it can be mapped to the user system; it can be left unmapped. The remaining 8K bytes of memory space excluding the on-chip internal RAM and special function register area can be mapped to the user system or be left unmapped.

Emulation

The IE-78320 allows the following methods of program emulation: real-time program execution with or without breakpoints; non-real-time program execution for a specified number of instructions or until a register condition is satisfied. During non-real-time program execution, the display and trace of procedures at a nesting level deeper than the routine from which execution was started is optional. During non-real-time emulation, each executed instruction is displayed with its frame number and bus cycle status, instruction address, data, label, mnemonic, and operands. Display of the registers is optional and can be specified by the user.

Following termination of real-time program emulation, the elapsed emulation time, number of instructions executed, and the registers (general registers, stack pointer, program counter, and program status word) are displayed and the IE-78320 enters single-step emulation mode. Following termination of non-real-time program emulation, the IE-78320 enters the single-step emulation mode. Each time the enter key is pressed during single-step emulation, the next instruction is executed and its frame number and bus cycle status, instruction address, data, label, mnemonic, operands, and registers are displayed.

Emulation Accuracy

All port-related and A/D converter related signals are taken directly from the emulation chip. These signals function identically to the devices. To improve signal quality a 100 Ω resistor is inserted in series on each port-related line. Other signals differ from the actual device due to buffering and control gating.

Breakpoint Capabilities

The IE-78320 has four types of break functions: event detection breaks, command breaks, fail-safe breaks, and manual breaks. Event detection breaks can be set to stop emulation on: address, data, status, external data, or loop count for main bus activity (addresses 0H to 0FDFFH and 0FFD0H to 0FFDFH); address, status, external data, or loop count for CPU internal bus activity (addresses 0FE00H to 0FFCFH and 0FFE0H to 0FFFFH); matching a condition on external sense clip number one; executed instruction count; four parallel instruction address breakpoints or up to a four-level sequential instruction address breakpoint. Combinations of the above conditions can be specified as a break event and enabled for real-time emulation.

Once a break event associated with main bus activity or CPU internal bus activity is reached during emulation, several instructions are executed before emulation is stopped. The exact number of instructions slipped (slippage) depends on the instructions in the prefetch queue and if the emulation CPU is accessing internal ROM or external memory. Slippage does not occur on parallel or sequential instruction address break events.

Command breaks can be specified on the command line of the non-real-time emulation command. Non-real-time emulation can be stopped when an internal register condition is satisfied or a specified number of instructions have been executed.

Fail-safe break conditions occur unconditionally and include a non-map access break, write protected break and turbo access break. A non-map access break occurs when an attempt is made to access a non-mapped memory area or non-existing special function register (SFR). A write-protected break occurs when an attempt is made to write to read-only emulation memory or SFR. A turbo access break occurs when a continuous fetch operation is performed on any off-chip emulation memory.

A manual break occurs when the ESC key is input during non-real-time execution, or the STP or reset command is input during real-time execution.

Trace Capabilities

The IE-78320 has a 2K x 44-bit trace RAM for storing emulation data from each machine cycle. The addresses, data, and CPU status of the main bus are always traced along with either the addresses and status of the CPU internal bus or the external sense clips as selected by the user. There are three types of trace displays: frame mode, instruction mode, and instruction mode with macro service. In the frame mode display, the frame number and type, address and data information and external sense clip status are displayed for each frame in the order in which they are traced. In instruction mode, the executed instructions are displayed with their frame number, bus cycle status, instruction address, data, external sense clip data, label, mnemonics, and operands. In the instruction mode with macro service, macro service reads and writes are added to the instruction mode display.

A number of trace display options are available. These include the display of all trace data, the display of all frames related to branch processing and the occurrence of an interrupt, the display of only the frames meeting the trace data search conditions, the display of five lines before or after the frame meeting the trace data search condition and the display of a specified number of lines following the detection of the specified break condition.

During real-time program execution without breakpoints, the break condition can be used to stop the tracer a specified number of frames after the break condition is satisfied. At tracer stop time, the trace buffer can be viewed, new trace conditions set, and the tracer restarted while the program continues to execute in real-time.

CMOS Protection

The latch-up warning circuit is activated when a CMOS latch-up condition occurs in the emulation CPU or any of its peripheral CMOS devices. A protection circuit isolates the power supply to the emulation CPU, its peripheral CMOS devices and all TTL devices driving the CMOS devices and the message "Emulation CPU Latchup !" is displayed.

Utilities

The upload/download commands provide easy loading and saving of hex files to and from a host computer. The on-board assembler/disassembler allows the user to avoid debugging in machine code. The symbolic debugging commands allow the use of labels instead of absolute addresses. Full data manipulation commands are available for memory, the general registers, and special function registers. Initialization commands allow the user to choose a clock source, a base number, to specify the serial parameters for channel two and to define the system memory map. Other commands are available to evaluate an arithmetic expression, to output an external trigger signal when an specified event has occurred, and to control an NEC PG-series PROM programmer.

System Mode

The IE-78320 can be connected to an IBM PC®, PC/XT®, PC AT®, or PC-9800 series by an RS232C port and operated in system mode. By using the accompanying control software, the debugging capabilities of the IE-78320 are greatly increased. The controller program has a macro command file capability, allowing the user to execute a defined set of commands automatically. The on-line help facility, the history command display, and the ability to send the console display to a printer or to the disk ease debugging tasks. The uploading and downloading capability can be utilized to upload and download both object code and symbol information. MS-DOS® programs can be executed without terminating the controller program. Other advantages are a verify command that compares memory to hex files, an alter symbol command, and a termination command for exiting to the operating system.

Table 1 lists commands available for both the stand-alone and system modes of the IE-78320. Commands listed in table 2 supplement table 1, but can only be used in the system mode.

Table 1. Stand-Alone and System Mode Commands

Command	Function
ASM	Assembles source code line by line
BRA	Specifies break events in program or internal data memory area
BRD	Selects external signal as break event
BRE	Sets a number of instructions executed as break event
BRM	Enables break events
BRS	Sets parallel or sequential instruction address breakpoints
BRn	ORs various break events together (n = 0 to 3)
CLK	Selects internal or external clock
CNT	Displays elapsed emulation time and number of instructions executed
DAS	Disassembles program memory
DLY	Changes/displays number of frames to be traced after trace trigger has been detected
LOD	Loads hex format file into program memory
MAP	Displays/changes memory map
MAT	Evaluates arithmetic expression
MDR	Displays/modifies mode registers of emulator CPU
MEM	Displays/ changes/fills/moves/exchange/ searches/ verifies/tests memory
MOD	Sets channel two serial parameters

Table 1. Stand-Alone and System Mode Commands (cont)

Command	Function
MOV	Moves memory content to different mapping area
OUT	Outputs external trigger
PGM	Controls PG series programmer from IE-78320
REG	Displays/modifies registers of emulator CPU
RES	Resets the IE-78320 and/or emulator CPU
RUN	Executes programs in real-time or non-real-time
SAV	Saves contents of memory onto disk
SFR	Displays/modifies special function registers of emulator CPU
SPR	Displays/modifies special registers of emulator CPU
STP	Stops emulation CPU during real-time emulation
SYM	Adds/deletes/displays/changes/loads/saves symbols
TRD	Displays trace data
TRF	Sets condition for trace buffer search
TRG	Starts real-time tracer during real-time emulation
TRM	Selects CPU internal bus or external sense clips for tracing
TRP	Displays/moves trace buffer pointer
VRY	Compares contents of an object file with memory

Table 2. System Mode Only Commands

Command	Function
COM	Creates command file
DIR	Displays disk directory
DOS	Allows execution of MS-DOS programs
EXT	Terminates IE-78320 controller program operation
HIS	Displays last twenty commands
HLP	Displays format of commands
LOD	Loads object code and symbol files
LST	Sends console display to disk or printer
SAV	Saves contents of memory and the debug environment onto disk
STR	Automatically executes command string file

Equipment Supplied

The IE-78320-R package consists of the following:

- IE-78320-R housing
- IE-78320 user's manuals
- PC-9800 series system disk
- IBM PC system disk
- AC power cable
- AC ground adapter
- Ground cable
- Spare fuse
- RS-232C interface cable
- Warranty policy and registration card

Basic Specifications

- Weight: 8.5 kg
- External dimensions:
  length, 370 mm; width, 160 mm;
  height, 283 mm
- Power source: 100 V AC, 50/60 Hz

Environmental Characteristics

- Operating temperature range: 10 to +40°C
- Storage temperature range: 20 to +45°C
- Ambient humidity range: 10 to 90% relative humidity

Documentation

For further information on IE-78320 operation, NEC Electronics Inc. provides the following manuals with the in-circuit emulator:

- IE-78320 μPD78320/322 In-Circuit Emulator Hardware Manual
- IE-78320 μPD78320/322 In-Circuit Emulator Software Manual

Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

CC7831X
C Compiler Package for the μPD7831X/μPD7831XA Series

Description

The CC7831X C compiler package for the NEC μPD7831X/μPD7831XA microcomputers consists of a Kernighan and Ritchie compatible C cross compiler (CC310), relocatable assembler (RA310), linker (LK310), librarian (LB310), locater (LC310), and an emulator controller program. The CC7831X C compiler package is available for use on an MS-DOS® system with a freestanding system as target (embedded system).

Features

- □ Kernighan and Ritchie standard C
  - — unsigned, enum, typedef, interrupt keywords
  - — extern, auto, static, register keywords
- □ Legal C code verification integrated into the compiler
- □ User-selectable and directable output files, list and full cross reference files
- □ Macro definitions
- □ Branch optimization
- □ Conditional assembly
- □ Simple diagnostics
- □ Powerful librarian

Ordering Information

Part Number	System	Description
CCMSD-I5DD-7831X	MS-DOS	5-1/4 inch double-density floppy diskette

C CROSS COMPILER (CC310)

Description

The CC310 C cross compiler converts standard C source code into relocatable object modules. The same relocatable object format is used for all relocatable object files in the C compiler package, regardless of how generated (by an assembler or compiler).

MS-DOS is a registered trademark of Microsoft Corporation.

Compiler Options

The CC310 C compiler supports the following options during compilation:

- Integer size control
- Include file control
- Defining/undefining constants
- Local symbol information included in object files
- Prologue/epilogue control
- Forced stack checking before each C function
- Packed data allocation
- Special relocatable data segment

C Library Functions

The CC310 C compiler library includes most of the important C library functions that apply to PROM based embedded systems. All library functions reside in the supplied library files. Header files that declare the set of library functions are also included.

The following character operation macros are available:

CHARACTER HANDLING <ctype.h>

Classification Macros:

isalnum isalpha isascii iscntrl isdigit isgraph
islower isprint ispunct isspace isupper isxdigit

Conversion Macros:

toascii tolower toupper

The following library functions are available:

NON-LOGICAL JUMPS <setjmp.h>

longjmp setjmp

FORMATTED INPUT/OUTPUT <stdio.h>

sscanf sprintf

GENERAL UTILITIES

crt0 (startup for C programs)
cipt (interrupt system support)
chkstk (check for stack overflow)

STRING HANDLING <string.h>

strcat strchr strcmp strcpy strcspn strlen strncat strncmp strncpy strpbrk strrchr strspn strtok

MATHEMATICS

abs atoi atol

Memory Models

CC310 supports only the small memory model, since the μPD7831X/μPD7831XA can only address a maximum of 64K bytes of program memory.

RELOCATABLE ASSEMBLER (RA310)

Description

RA310 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction is valid for the target μPD7831X, μPD7831XA, or μPD7832X microcomputer and produces a listing file and a relocatable object module.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Capability

RA310 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence differs from a subroutine call because the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler but are not translated into machine code during assembly. Basic assembler directives include: storage definition and allocation directives (DB,DW, DS, DBIT); symbol directives (EQU, SET); location counter control directive (ORG). Program control directives include: segment directives (CSEG, CSEG FIXED, CSEG CALLT0, CSEG CALLT1, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); register assignment directives (RSS); macro directives (MACRO, LOCAL, REPT, IRP, ENDM, EXITM); automatic BR instruction selection directive (BR) and assembly termination directive (END).

Assembler Controls

There are two types of assembler controls available for RA310. The primary controls specified in the assembler command line or at the beginning of the source module are as follows:

- Processor selection
- Output object file selection
- Output list file selection
- Listing format controls
- Date specification
- Optimization selection
- Workfile drive selection
- Symbol letter case selection

General controls, specified in the source program, are as follows:

- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing subtitles
- Conditional assembly controls

LINKER (LK310)

LK310 combines multiple relocatable object modules and library modules and converts them into a single relocatable object module. The linker resolves PUBLIC/EXTRN references between modules, creating a relocatable output module that contains both relocatable object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK310 may be specified in either the command line or in a parameter file. Linker options include specifying the date and the absolute load module name, specifying the creation of a list file containing a link map, and specifying the letter case for symbols.

LOCATER (LC310)

LC310 converts a relocatable object module with no external references into an ASCII hexadecimal format absolute object code file. The locater outputs two files: an absolute load file in an expanded seven-bit ASCII hexadecimal format, which can be downloaded to a PROM programmer and a symbol file for the symbolic debugger. Locater options include specifying the starting address and order for code/data/stack segments, specifying areas of memory to be protected from being assigned, and specifying the creation of a map file with symbol tables.

LIBRARIAN (LB310)

LB310 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

The librarian creates and maintains library files containing relocatable object modules. Modules can be added to or deleted from a library file, or the contents of the library file can be listed.

EMULATOR CONTROLLER PROGRAM

Absolute object files produced by the CC7831X C compiler package can be debugged using an NEC standalone in-circuit emulator. An NEC emulator controller program allows you to communicate with the emulator through an RS-232C serial line. The emulator controller program is available to run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS. It provides the following features:

- Uploading/downloading of object/symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Storage of debug session on disk
- Storage of last 20 commands for recall

IBM PC, PC/XT, and PC AT are trademarks of International Business Machines Corporation.

LICENSE AGREEMENT

CC7831X is sold under terms of a license agreement, which is included with purchased copies. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

DOCUMENTATION

For further information on source program formats, C compiler and assembler operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- CC78XXX C Compiler μPD78XXX C Compiler User's Manual
- CC78XXX C Compiler μPD78XXX Relocatable Assembler User's Manual

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

CC7832X C Compiler Package for the μPD7832X Series

Description

The CC7832X C compiler package for the NEC μPD7832X microcomputers consists of a Kernighan and Ritchie compatible C cross compiler (CC320), relocatable assembler (RA310), linker (LK310), librarian (LB310), locater (LC310), and an emulator controller program. The CC7832X C compiler package is available for use on an MS-DOS® system with a free-standing system as target (embedded system).

Features

- □ Kernighan and Ritchie standard C
  - — unsigned, enum, typedef, interrupt keywords
  - — extern, auto, static, register keywords
- □ Legal C code verification integrated into the compiler
- □ User-selectable and directable output files, list and full cross reference files
- □ Macro definitions
- □ Branch optimization
- □ Conditional assembly
- □ Simple diagnostics
- □ Powerful librarian

Ordering Information

Part Number	System	Description
CCMSD-I5DD-7832X	MS-DOS	5-1/4 inch double-density floppy diskette

C CROSS COMPILER (CC320)

Description

The CC320 C cross compiler converts standard C source code into relocatable object modules. The same relocatable object format is used for all relocatable object files in the C compiler package, regardless of how generated (by an assembler or compiler).

MS-DOS is a registered trademark of Microsoft Corporation.

Compiler Options

The CC320 C compiler supports the following options during compilation:

- Integer size control
- Include file control
- Defining/undefining constants
- Local symbol information included in object files
- Prologue/epilogue control
- Forced stack checking before each C function
- Packed data allocation
- Special relocatable data segment

C Library Functions

The CC320 C compiler library includes most of the important C library functions that apply to PROM based embedded systems. All library functions reside in the supplied library files. Header files that declare the set of library functions are also included.

The following character operation macros are available:

CHARACTER HANDLING <ctype.h>

Classification Macros:

isalnum isalpha isascii iscntrl isdigit isgraph
islower isprint ispunct isspace isupper isxdigit

Conversion Macros:

toascii tolower toupper

The following library functions are available:

NON-LOGICAL JUMPS <setjmp.h>

longjmp setjmp

FORMATTED INPUT/OUTPUT <stdio.h>

sscanf sprintf

GENERAL UTILITIES

crt0 (startup for C programs)
cipt (interrupt system support)
chkstk (check for stack overflow)

STRING HANDLING <string.h>

strcat strchr strcmp strcpy strcspn strlen strncat
strncmp strncpy strpbrk strrchr strspn strtok

MATHEMATICS

abs atoi atol

Memory Models

CC320 supports only the small memory model since the μPD7832X can only address a maximum of 64K bytes of program memory.

RELOCATABLE ASSEMBLER (RA310)

Description

RA310 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction is valid for the target μPD7831X, μPD7831XA, or μPD7832X microcomputer and produces a listing file and a relocatable object module.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Capability

RA310 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence differs from a subroutine call because the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

Assembler Directives

Assembler directives give instructions to the assembler but are not translated into machine code during assembly. Basic assembler directives include: storage definition and allocation directives (DB,DW, DS, DBIT); symbol directives (EQU, SET); location counter control directive (ORG). Program control directives include: segment directives (CSEG, CSEG FIXED, CSEG CALLT0, CSEG CALLT1, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); register assignment directives (RSS); macro directives (MACRO, LOCAL, REPT, IRP, ENDM, EXITM); automatic BR instruction selection directive (BR) and assembly termination directive (END).

Assembler Controls

There are two types of assembler controls available for RA310. The primary controls specified in the assembler command line or at the beginning of the source module are as follows:

- Processor selection
- Output object file selection
- Output list file selection
- Listing format controls
- Date specification
- Optimization selection
- Workfile drive selection
- Symbol letter case selection

General controls, specified in the source program, are as follows:

- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing subtitles
- Conditional assembly controls

LINKER (LK310)

LK310 combines multiple relocatable object modules and library modules and converts them into a single relocatable object module. The linker resolves PUBLIC/EXTRN references between modules, creating a relocatable output module that contains both relocatable object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK310 may be specified in either the command line or in a parameter file. Linker options include specifying the date and the absolute load module name, specifying the creation of a list file containing a link map, and specifying the letter case for symbols.

LOCATER (LC310)

LC310 converts a relocatable object module with no external references into an ASCII hexadecimal format absolute object code file. The locater outputs two files: an absolute load file in an expanded seven-bit ASCII hexadecimal format, which can be downloaded to a PROM programmer and a symbol file for the symbolic debugger. Locater options include specifying the starting address and order for code/data/stack segments, specifying areas of memory to be protected from being assigned, and specifying the creation of a map file with symbol tables.

LIBRARIAN (LB310)

LB310 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

The librarian creates and maintains library files containing relocatable object modules. Modules can be added to or deleted from a library file, or the contents of the library file can be listed.

EMULATOR CONTROLLER PROGRAM

Absolute object files produced by the CC7832X C compiler package can be debugged using an NEC standalone in-circuit emulator. An NEC emulator controller program allows you to communicate with the emulator through an RS-232C serial line. The emulator controller program is available to run on the IBM PC®, PC/XT®, or PC AT® under MS-DOS. It provides the following features:

- Uploading/downloading of object/symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Storage of debug session on disk
- Storage of last 20 commands for recall

LICENSE AGREEMENT

CC7832X is sold under terms of a license agreement, which is included with purchased copies. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users for one year.

IBM PC, PC/XT, and PC AT are trademarks of International Business Machines Corporation.

DOCUMENTATION

For further information on source program formats, C compiler and assembler operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- CC78XXX C Compiler μPD78XXX C Compiler User's Manual
- CC78XXX C Compiler μPD78XXX Relocatable Assembler User's Manual

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

NEC
NEC Electronics Inc.

RA78K3 Relocatable Assembler Package for the µPD7831X/7832X

Description

The RA78K3 relocatable assembler package converts symbolic source code for the µPD7831X and µPD7832X eight/sixteen-bit, single-chip microcomputers into executable absolute address object code. The RA78K3 relocatable assembler package consists of four separate programs: assembler (RA78K3), linker (LK78K3), locater (LC78K3), and librarian (LB78K3).

RA78K3 translates a symbolic source module into a relocatable object module. The assembler verifies that each instruction assembled is valid for the target microcomputer specified at assembly time and produces a listing file and a relocatable object module.

LK78K3 combines multiple relocatable object and library modules and converts them to a single relocatable object module. LC78K3 converts a relocatable object module with no external references into an ASCII hexadecimal format absolute object code file.

LB78K3 allows commonly used relocatable object modules to be stored in one file and linked into multiple programs, greatly increasing programming efficiency. When a library file is included in the input of the linker, the linker extracts from the library file only those modules required to resolve external references and links them with the other modules.

Features

- □ Absolute address object code output
- □ User-selectable and directable output files
- □ Macro definitions
- □ Branch optimization
- □ Conditional assembly
- □ Extensive error reporting
- □ Powerful librarian
- □ Runs under MS-DOS® and VAX®/VMS® operating systems

Ordering Information

Part Number	System	Description
RA78K3-D52	MS-DOS	5-1/4 inch double-density floppy diskette
RA78K3-VVT1	VAX/VMS	9-track 1600 BPI magnetic tape

Program Syntax

An RA78K3 source module consists of a series of code, data, or bit segments. Each segment consists of statements composed of up to four fields: symbol, mnemonic, operand, and comment.

The symbol field may contain a label, whose value is the instruction or data address, or a name which represents an instruction address, data address, or constant. The mnemonic field may contain an instruction or assembler directive. The operand field contains the data or expression for the specified instruction or directive. The comment field allows explanatory comments to be added to a program.

Character constants are translated into seven-bit ASCII codes. Numeric constants may be specified as binary, octal, decimal, or hexadecimal. Arithmetic expressions may include the operators +, -, *, /, MOD, OR, AND, NOT, XOR, EQ, NE, LT, LE, GT, GE, SHR, SHL, LOW, HIGH, ., ( ), and character constants.

Macro Definition

RA78K3 allows the definition of macro code sequences with up to five parameters, LOCAL symbols, and special repeated code sequences. The macro code sequence differs from a subroutine call: the invocation of a macro in the source code results in the direct replacement of the macro call with the defined code sequence.

MS-DOS is a registered trademark of Microsoft Corporation.
VAX and VMS are registered trademarks of Digital Equipment Corporation.

Assembler Directives

Assembler directives give instructions to the assembler. They are not translated into machine code during assembly. Basic assembler directives include: storage definition and allocation directives (DB, DW, DS, DBIT); symbol directives (EQU, SET); and location counter control directive (ORG). Program control directives include: segment directives (CSEG, DSEG, BSEG, ENDS); linkage directives (NAME, PUBLIC, EXTRN, EXTBIT); macro directives (MACRO, LOCAL, REPT, IRP, EXITM, ENDM); automatic BR instruction directive (BR); register assignment directive (RSS); and assembly termination directive (END).

Assembler Controls

The RA78K3 assembler (figure 1) has two types of controls. Primary controls are specified in the assembler command line or at the beginning of the source module and are as follows:

- Processor selection
- Output object creation selection
- Output list file selection
- Listing format controls
- Optimization selection
- Work file drive specification

General controls are specified in the source program and are as follows:

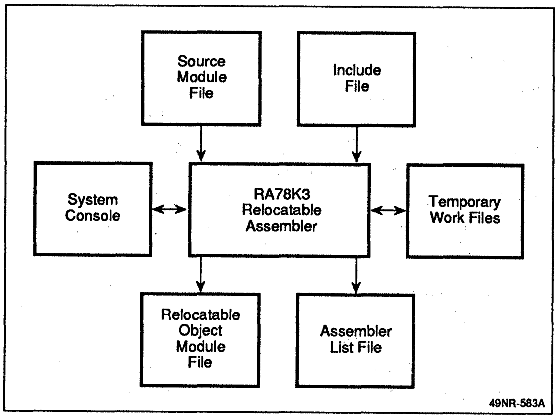
- Inclusion of other source files
- Page eject
- Generation/suppression of listing
- Listing titles
- Conditional assembly controls

The listing file contains either the complete assembly listing or only the lines with errors, and a symbol or cross-reference table. The symbol table shows all defined symbols in alphabetical order, with the types, attributes, and the values initially assigned to them.

The cross-reference table contains all defined symbols and the numbers of all statements referring to them. The object file contains the relocatable object module. This is an NEC proprietary relocatable object module format.

If the optimization option is chosen, the assembler will generate the most efficient code by converting, wherever possible, three-byte absolute branches into two-byte relative branches.

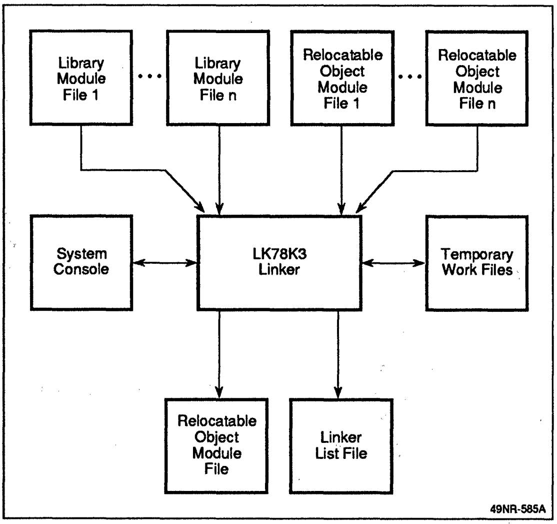
Figure 1. Relocatable Assembler Functional Diagram



Linker

The LK78K3 linker (figure 2) combines several relocatable object modules, resolving PUBLIC/EXTRN references between modules, to create a relocatable output module. This output module contains both relocatable object code and symbol information. The linker will also search library files for required modules to resolve external references. The linker controls for LK78K3 can be specified in either the command line or in a parameter file. The programmer can specify the date, and absolute load module name, and control the creation of a list file containing a link map.

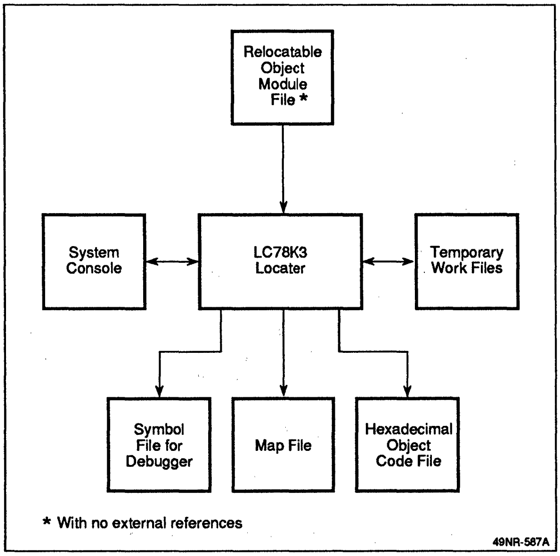
Figure 2. Linker Functional Diagram



Locater

The LC78K3 (figure 3) locater outputs two files: an absolute load file in a seven-bit ASCII expanded hexadecimal format, which can be downloaded to a PROM programmer; and a symbol file for the symbolic debugger. The programmer can specify the starting address and order for code/data/stack segments, and can protect areas of memory from being assigned. The programmer can specify that a map file with symbol tables be created.

Figure 3. Locater Functional Diagram


Relocatable Object Module File *
System Console
LC78K3 Locater
Temporary Work Files
Symbol File for Debugger
Map File
Hexadecimal Object Code File
* With no external references
49NR-587A

Librarian

The LB78K3 librarian creates and maintains library files containing relocatable object modules. This reduces the number of files to be linked together by storing several modules in a single file. This provides an easy way to link frequently used modules into programs. Modules can be added to, deleted from, or replaced within a library file; or the contents of the library file can be listed.

Operating Environment

The NEC RA78K3 package runs under a variety of operating systems. One version runs on an MS-DOS system with one or more disk drives and at least 128K of system memory. Another version runs on a Digital Equipment Corporation VAX computer under the VMS (Version 4.1 or later) operating system.

Emulator Controller Program

Absolute object files produced by the RA78K3 relocatable assembler package can be debugged with the appropriate NEC stand-alone in-circuit emulator. NEC emulator controller programs allows communication with the emulator through an RS-232C serial line. An emulator controller program can run on the IBM PC®, PC/XT®, OR PC AT® under MS-DOS and is provided with the in-circuit emulator at no extra charge.

These emulator controller programs provide the following features:

- Uploading/downloading of object and symbol files
- Symbolic debugging capability
- Complete emulator control from host console
- On-line help facilities
- Macro command file capabilities
- Host system directory and file display
- Disk storage of debug session
- Storage of last 20 commands for recall

License Agreement

RA78K3 is sold under terms of a license agreement, which is included with the assembler. The accompanying card must be completed and returned to NEC Electronics Inc. to register the license. Software updates are provided free to registered users.

Documentation

For further information on source program formats, assembler operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- RA78K3 μPD7831X/μPD7832X Relocatable Assembler Package, Language Manual
- RA78K3 μPD7831X/μPD7832X Relocatable Assembler Package, Operation Manual (MS-DOS)
- RA78K3 μPD7831X/μPD7832X Relocatable Assembler Package, Operation Manual (VMS)

This documentation is provided with purchased copies of the package. Additional copies may be obtained from NEC Electronics Inc.

IBM PC, PC/XT, and PC AT are registered trademarks of Internaltional Business Machines Corporation.

NEC
NEC Electronics Inc.

ST78K3
Structured Assembler Preprocessor for the μPD783XX Series

Description

The ST78K3 structured assembler preprocessor is a companion program to the RA78K3 relocatable assembler for the NEC μPD783XX series of microcomputers. ST78K3 converts a source code file containing structured assembly statements into a pure assembly language source file, which then can be assembled with RA78K3.

ST78K3 will convert a structured assembly statement into one or more μPD783XX assembly language instructions which perform the desired operation. Since ST78K3 converts only the structured assembly statements and does not convert μPD783XX assembly language instructions, a structured source program can include a combination of μPD783XX structured assembly statements and assembly language.

ST78K3 enables the assembly language programmer to use some of the structures and syntax of higher-level languages such as the C language. This improves program readability and reliability, and increases programmer productivity.

Features

□ Control structures for conditions, looping, and switch-case

□ Preprocessor directives for conditional code generation

□ C-like representation of comparison operations

□ C-like representation of assignment/arithmetic operations

□ Increment and decrement operators

□ Allow use of all μPD783XX mnemonics, registers, and features

□ Runs under MS-DOS® and VAX®/VMS® operating systems

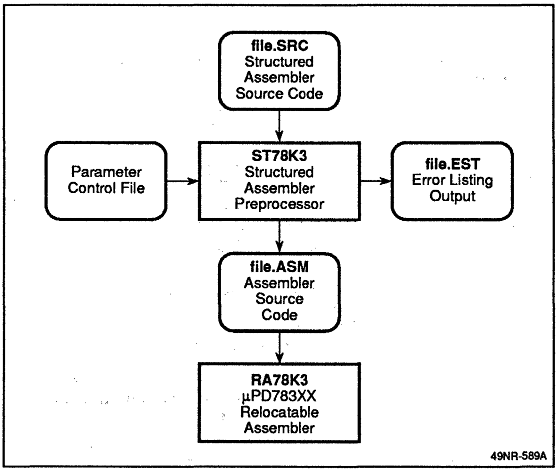
Ordering Information

The ST78K3 structured assembler preprocessor is included in the following software packages at no cost:

- RA78K3 μPD783XX Relocatable Assembler Package

MS-DOS is a registered trademark of Microsoft Corporation.
VAX and VMS are registered trademarks of Digital Equipment Corporation.

Structured Assembler Preprocessor Functional Diagram


file.SRC
Structured Assembler Source Code
Parameter Control File
ST78K3 Structured Assembler Preprocessor
file.EST
Error Listing Output
file.ASM
Assembler Source Code
RA78K3
μPD783XX Relocatable Assembler
49NR-589A

Summary Of Structured Language

A line of source code for ST78K3 contains either a structured assembly statement or a μPD783XX assembly language statement. μPD783XX assembly language statements (μPD783XX instructions, RA78K3 directives, or RA78K3 controls) pass through ST78K3 without change.

Structured assembly statements consist of preprocessor directives, assignment statements, and control statements. These statements are entered one per line, and are terminated by a line feed character. An optional comment may follow a semicolon at the end of the statement; all text following a semicolon is ignored by ST78K3.

Preprocessor directives cause ST78K3 to include or omit portions of code. Assignment statements cause ST78K3 to generate one or more μPD783XX assembly language instructions to alter the contents of a register or variable. Control statements cause ST78K3 to generate the necessary instructions to test conditions and change control flow based on those conditions.

Preprocessor Directives

ST78K3 preprocessor directives set and test variables, allowing conditional processing of code; include external files; and map instructions to μPD783XX CALT table reference instructions. Table 1 lists the preprocessor directives and their functions.

Table 1. Preprocessor Directives and Functions

Directive	Function
#define NAME value	Defines the variable NAME, set to the supplied value.
#ifdef ABC <statements> #else <statements> #endif	If ABC has been defined as above, or on the command line with the -D option, the first set of statements is processed and the second set ignored; if ABC has not been defined, or defined as zero, the first set of statements is ignored and the second set is processed.
#include "filename"	The named file is read from disk and processed as if included in the source.
#defcallt @LABEL CALL !label #endcallt	Whenever the instruction "CALL !label" is encountered in the source program, it is replaced by "CALLT [@LABEL]". The label must be defined in the CALLT table.

Assignment, Increment, and Decrement Statements

ST78K3 provides the ability to represent an assignment, or an assignment with an arithmetic operation, in C language syntax:

```
destination <assign-op> source
```

The assignment operators allow either simple assignment, or the combination of an assignment with an arithmetic operation on the source and destination.

Examples:

```
A = B     ;Move contents of B register to A
A + =[HL] ;Add contents of memory at HL to A,
          ;store in A
```

Where an assignment requires an intermediate register to hold the value being assigned, the register is designated by naming it in parentheses following the assignment operation.

Examples:

```
DATA1 = B (A)   ;Store contents of B into memory at
                ;DATA1, using A as temporary storage
BC & = HL (XA)  ;and BC with HL, store in BC,
                ;use XA as temp
```

The increment and decrement operators (+ + and --) operate on a single operand.

Table 2 lists the assignment operators with examples and functions.

Table 2. Assignment Operators with Examples and Functions

Operator	Example	Function
=	A = B	A ← B
< - >	A < - > B	Contents of A and B are exchanged
+ =	A + = B	A ← A + B
-=	A -= B	A ← A - B
*=	AX *= B	AX ← AX * B
/=	AX /= C	AX ← AX / C
&=	A &= B	A ← A & B (logical AND)
|=	A |= B	A ← A | B (logical OR)
^=	A ^= B	A ← A ^ B (logical XOR)
> > =	A> > =B	$(CY \leftarrow A_0, A_{n-1} \leftarrow A_n, ..., A_{max} \leftarrow 0)$ x B times
< < =	A< < =B	$(CY \leftarrow A_{max}, A_{n+1} \leftarrow A_n, ...A_0 \leftarrow 0)$ x B times
+ +	A+ +	A ← A + 1
--	A--	A ← A - 1

Control Statements

Control statements allow conditions to be tested. Based on the results of the test, blocks of code are allowed to be executed or skipped. Reserved words in the control statement define the start and end of blocks of code, and expressions to be evaluated.

Example:

```
if ( A = = [HL])      ;The condition is tested.
   P5 = B (A)         ;If A equals the content of memory
   A = [HL]           ;at HL, this code is executed.
else
   A + = [HL]         ;Otherwise, this code is executed.
   A - = B
   P5 = A
endif
```

Table 3 shows the control statements and their functions.

Table 3. Control Statements and Function

Control Statement	Function
if - elseif - else - endif	Test variable expressions
if_bit - elseif_bit - else - endif	Test bit expressions
switch - case - default - ends	Select based on variable
for - next	Loop, test variable
while - endw	Loop, test variable
repeat - until	Loop, test variable
while_bit - endw	Loop, test bit
repeat - until_bit	Loop, test bit
break	Exit control block
continue	Skip to top of block
goto LABEL	Branch to label

Variable And Bit Expressions

Variable expressions for tests consist of a single value, comparison between two variables, or a logical combination of comparisons. Bit expressions test individual bits. Table 4 shows examples of comparisons:

Table 4. Examples of Variable Expression Comparisons

Comparison	Meaning
if (A)	True if A is non-zero
if (A < B)	True if A is less than B
if ((A < B) && (A > C))	True if A is less than B and greater than C
if_bit (P1.2)	True if bit 2 of P1 is 1
if_bit (!P1.2)	True if bit 2 of P1 is 0

The allowable expressions using variables are shown in table 5.

Table 5. Expressions and Examples

Expression	Example
Primary	(A)
Term	(A < = B)
Term && Term	((A < B) && (A > C)) (logical AND)
Term || Term	((A= =C) || (A= =B)) (logical OR)

A primary value for a variable expression is a register name or defined symbol. A term consists of two primary values compared with a binary operator. Table 6 lists the supported binary operators and their meanings.

Table 6. Binary Operators

Binary Operator	Meaning
= =	Equals
! =	Not equal
>	Greater than
> =	Greater than or equal
<	Less than
< =	Less than or equal

Bit expressions test individual bits of registers, ports, or memory locations. Table 7 shows the acceptable forms of bit expressions.

Table 7. Bit Expressions and Examples

Bit Expression	Example
Bit_primary	(P0.1)
!Bit_primary	(!CY)
Bit_primary && Bit_primary	(A.0 && CY)
Bit_primary || Bit_primary	(P0.2 || CY)

A Bit_primary can be either a reserved word bit identifier, such as a bit of a register or port (P0.1, CY), or a bit definition symbol (SB0 EQU P0.2).

ST78K3 Operation And Controls

ST78K3 consists of four files: ST78K3.EXE, ST78K3.OMA (μPD78310A/312A), ST78K3.OMB (μPD78320/322/327/328), and ST78K3.OMC (μPD78330/334). Before invoking ST78K3, the user must copy the appropriate file to ST78K3.OM1. For example, if the user is developing code for the μPD78310A/312A, the user must type:

```
C>COPY ST78K3.OMA ST78K3.OM1
```

ST78K3 is invoked by specifying the name of the source file, followed by optional controls.

Example:

```
C>ST78K3 ABC.SRC -DXYZ=3
```

ST78K3 reads the specified source file and produces an output assembly language file, which can be input to RA78K3. The output file contains all lines provided in the input source file, plus those generated by ST78K3. Lines containing no statements for the structured assembler are passed through unchanged. Lines with structured assembly statements are placed in the output preceded by a semicolon. RA78K3 treats these lines as comments. These commented lines are then followed by the code generated by ST78K3.

The controls for ST78K3 are specified in the preprocessor command line or in a parameter file invoked in the command line. Table 8 lists the ST78K3 preprocessor controls and functions.

Table 8. ST78K3 Preprocessor Controls

Control	Function
-Ofilename	Specify name of output assembly source file
-Ffilename	Specify name of parameter file to be read
-Efilename	Specify name of error listing file
-Dsymbol[=value]	Define a symbol (like #define in code)
-I[d:][directory]	Define path for include file
-WTn1,n2,n3	Define TAB settings for generated code
-SCcharacter	Defines word symbol last character

The -O option allows the name of the output file to be specified. If not specified, the output file name defaults to the name of the input source file with the extension .ASM.

The -F option allows a parameter file to be specified, which will be read by ST78K3. This parameter file can contain a list of controls to be given to ST78K3, instead of or in addition to those specified on the command line.

The -E option specifies the name of the error listing file. The error file contains the file name, error number, description of error and the line containing the error. If the -E option is not specified, the error file name defaults to the name of the input source file with the extension .EST.

The -D control allows a symbol to be defined on the command line, with an optional value provided. If a symbol is defined but no value is specified, the value defaults to 1. If the source file contains a #define directive which specifies a variable with the same name as the -D control, the value on the command line will override the value in the #define directive.

The -I control specifies a drive or directory other than the current drive and directory to search for include files.

The -WT control specifies the number of TAB characters to insert before labels, instruction mnemonics, and instruction operands generated by ST78K3. This allows clear separation of assembly language instructions coded in the source file from those generated by ST78K3.

The -SCcharacter control specifies the character used as the last character in a word symbol. The character must be a letter of the alphabet or the @, _ or ?. This allows ST78K3 to distinguish between word and byte operations. Symbols which end in this character are treated as word symbols and will generate a word operation (ie. MOVW). If the -SC operation is not specified, ST78K3 assumes that a symbol ending with the character "P" or "p" is a word symbol.

Documentation

For further information on source program formats, preprocessor operation, and actual program examples, NEC Electronics Inc. provides the following documentation:

- ST78K2/ST78K3 μPD782xx/μPD783xx Structured Assembler Preprocessor User's Manual

This documentation is provided with purchased copies of the RA78K3 μPD783xx relocatable assembler package. Additional copies may be obtained from NEC Electronics Inc.

PG-1500 Series EPROM Programmer

Description

The PG-1500 series is a stand-alone EPROM programmer for programming 256-kilobit to 1-megabit EPROMs and EPROM/OTP devices for NEC's 4/8/16-bit single-chip microcomputers and digital signal processors. The system consists of the PG-1500 base programmer, interchangeable programmer adapter modules for standard EPROM devices and the μPD75XX/75XXX series 4-bit microcomputers, and a variety of programmer adapters to support the individual devices and package types. The PG-1500 can be controlled from either a remote terminal or host computer via an RS-232C serial port, or directly from the on-board keypad in stand-alone mode.

Features

- □ Interchangeable modules for programming:
  - — 256-kilobit to 1-megabit EPROMs
  - — NEC μPD75XX and μPD75XXX series 4-bit microcomputers
  - — NEC μPD78XX and μPD78XXX series 8-bit microcomputers
  - — NEC V-series 16-bit microcomputers
  - — NEC μPD77XXX digital signal processors
- □ 512K-bytes data RAM
- □ Silicon signature read function
- □ PROM insertion error detection circuitry
- □ Address splitting for 16/32-bit microprocessors
- □ Memory edit function to change/confirm PG-1500 buffer
- □ Address/data/message display LCD
- □ RS-232C serial interface
- □ Centronics compatible parallel interface
- □ Power-on diagnostics
- □ Supports three data transfer formats
  - — Intel extended hex (Note 1)
  - — Extended Tektronix hex (Note 2)
  - — Motorola S (Note 3)
- □ Two modes of operation
  - — Remote controlled
  - — Stand-alone
- □ Host Controller Program for IBM PC® Series

IBM PC is a registered trademark of International Business Machines Corporation

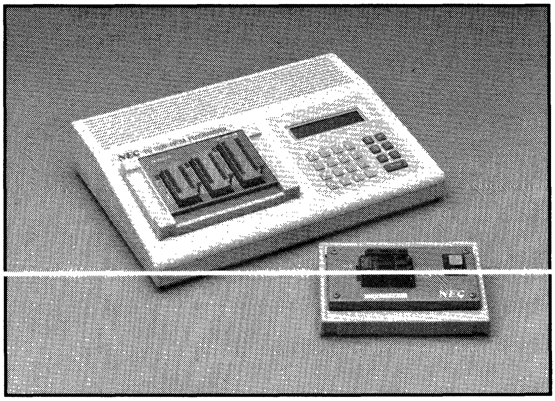
Notes:

(1) Developed by Intel Corporation.

(2) Developed by Tektronix Corporation.

(3) Developed by Motorola Inc.

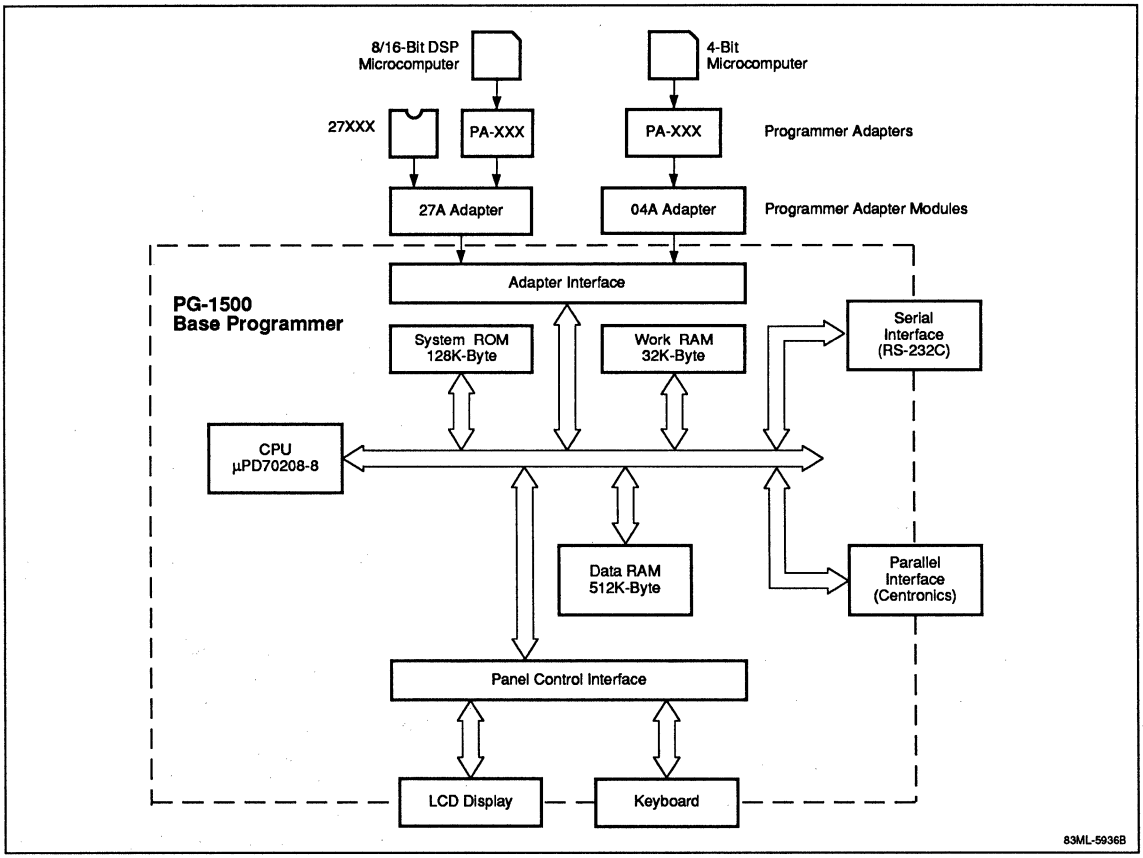
PG-1500 Series



Ordering Information

Part Number	Description
PG-1500	PG-1500 Series EPROM Programmer for 27XXX EPROMS, NEC 4/8/16 microcomputers, and DSP devices (includes 027A and 04A Programming Adapter Modules)
PA-70P322L	Programmer Adapter for µPD70P322K
PA-71P301GF	Programmer Adapter for µPD71P301GF
PA-71P301GQ	Programmer Adapter for µPD71P301GQ
PA-71P301KA	Programmer Adapter for µPD71P301KA
PA-71P301KB	Programmer Adapter for µPD71P301KB
PA-71P301L	Programmer Adapter for µPD71P301L
PA-75P54CS	Programmer Adapter for µPD75P54/64CS, µPD75P54/64G
PA-75P56CS	Programmer Adapter for µPD75P56/66CS, µPD75P56/66G
PA-75P008CU	Programmer Adapter for µPD75P008CU/GB
PA-75P036CW	Programmer Adapter for µPD75P036CW
PA-75P036GC	Programmer Adapter for µPD75P036GC
PA-75P108CW	Programmer Adapter for µPD75P108CW/DW/BCW, µPD75P116CW
PA-75P108G	Programmer Adapter for µPD75P108G/BGF, µPD75P116GF
PA-75P116GF	Programmer Adapter for µPD75P108G/BGF, µPD75P116GF
PA-75P216ACW	Programmer Adapter for µPD75P216ACW
PA-75P308GF	Programmer Adapter for µPD75P308GF, µPD75P316GF/AGF
PA-75P308K	Programmer Adapter for µPD75P308K, µPD75P316AK
PA-75P328GC	Programmer Adapter for µPD75P328GC
PA-75P402CT	Programmer Adapter for µPD75P402CT
PA-75P402GB	Programmer Adapter for µPD75P402GB
PA-75P516GF	Programmer Adapter for µPD75P516GF
PA-75P516K	Programmer Adapter for µPD75P516K
PA-77P25C	Programmer Adapter for µPD77P25C/D
PA-77P56C	Programmer Adapter for µPD77P56CR/G
PA-77P230R	Programmer Adapter for µPD77P230R
PA-78CP14CW	Programmer Adapter for µPD78CP14CW, DW
PA-78CP14GF	Programmer Adapter for µPD78CP14GF
PA-78CP14GQ	Programmer Adapter for µPD78CP14G/R
PA-78CP14L	Programmer Adapter for µPD78CP14L
PA-78P214CW	Programmer Adapter for µPD78P214CW
PA-78P214GJ	Programmer Adapter for µPD78P214GJ
PA-78P214GQ	Programmer Adapter for µPD78P214GQ
PA-78P214L	Programmer Adapter for µPD78P214L
PA-78P224GJ	Programmer Adapter for µPD78P224GJ
PA-78P224L	Programmer Adapter for µPD78P224L
PA-78P238GC	Programmer Adapter for µPD78P238GC
PA-78P238GJ	Programmer Adapter for µPD78P238GJ
PA-78P238KF	Programmer Adapter for µPD78P238KF
PA-78P238LQ	Programmer Adapter for µPD78P238LQ
PA-78P312CW	Programmer Adapter for µPD78P312ACW/DW
PA-78P312GF	Programmer Adapter for µPD78P312AGF
PA-78P312GQ	Programmer Adapter for µPD78P312AGQ/R
PA-78P312L	Programmer Adapter for µPD78P312AL
PA-78P322GJ	Programmer Adapter for µPD78P322GJ
PA-78P322KC	Programmer Adapter for µPD78P322KC
PA-78P322KD	Programmer Adapter for µPD78P322KD
PA-78P322L	Programmer Adapter for µPD78P322L

Figure 1. PG-1500 System Block Diagram



Architecture

The PG-1500 base unit contains an NEC µPD70208 (V40™) microprocessor with 128K bytes of monitor ROM, 32K bytes of working RAM, 512K bytes of data memory, an RS-232C serial port, a Centronics compatible parallel interface, an LCD display, and a 23-key keypad. Figure 1 shows a block diagram of the PG-1500.

The PG-1500 has two interchangeable programmer adapter modules: one for 27XXX EPROMS, NEC's 4/8/16 bit microcomputers, and DSP devices which use the µPD27C256A programming algorithm (027A board), and another for NEC's µPD75XX/75XXX 4-bit microcomputers which must be programmed in a serial fashion (04A board). These adapter modules plug directly into the top of the PG-1500 and can accept a wide variety of programmer socket adapters to support NEC's devices. Refer to the PG-1500 Programming Adapters Selection Guide for a list of all available adapters.

On power-up, the PG-1500 performs a self-diagnostic on its internal memory, its data bus, its power supply, and its reference voltages.

Operation

The PG-1500 operates in stand-alone mode from the on-board keypad, or in remote control mode from an external terminal or from a host computer via an RS-232C serial port.

Stand-Alone Mode

Table 1 lists the PG-1500 commands available in stand-alone mode.

Table 1. PG-1500 Commands in Stand-Alone Mode

Command	Function
DEVICE SELECT	Selects the EPROM to be used
DEVICE BLANK	Checks if the EPROM is blank
DEVICE COPY	Reads data from the EPROM
DEVICE PROG	Writes data into the EPROM
DEVICE VERIFY	Verifies EPROM contents against PG-1500 buffer
DEVICE CONT	Performs BLANK, PROG, VERIFY commands in sequence
EDIT CHANGE	Display/change the contents of the PG-1500 buffer
EDIT INITIAL	Initializes the PG-1500 buffer
EDIT MOVE	Moves a block of data within PG-1500 buffer
EDIT SEARCH	Searches PG-1500 buffer for 1-, 2-, or 4-byte patterns
EDIT C-SUM	Performs checksum on all data in PG-1500 buffer
FUNCTION S-IN	Inputs data from serial port in three formats
FUNCTION S-OUT	Outputs data from serial port in three formats
FUNCTION REMOTE	Sets PG-1500 to remote control mode
FUNCTION P-IN	Inputs data from parallel port in three formats
FUNCTION MODE	Sets up the RS-232C serial port parameters

The stand-alone commands fall into three groups:

- DEVICE commands associated with the device to be programmed
- EDIT commands for interacting with the PG-1500 memory buffer
- FUNCTION commands for setting up and controlling the PG-1500

The DEVICE commands are available to check if an EPROM device is blank, to copy data from the device to the PG-1500 buffer, to write the buffer data to the device, and to compare the data in the device with the data in the buffer. Blank checking, programming, and verification of the device can be performed sequentially using a single command.

To support various 16- and 32-bit microprocessors, the PG-1500 can split the data in its buffer in a variety of ways. When a data file is loaded into the PG-1500, the complete file is stored in the buffer and can be dynamically split during writing and verification. The PG-1500 supports the address splitting modes described in table 2.

Table 2. Address Splitting Modes

Mode	Description
Normal	The data is not split at all. Each byte of data in the buffer is programmed into the device.
16EVN	Each byte of data on an even address in the buffer is programmed into the device.
16ODD	Each byte of data on an odd address in the buffer is programmed into the device.
32/2E	The first two bytes of every four bytes in the buffer is programmed into the device.
32/2O	The third and fourth byte of every four bytes in the buffer is programmed into the device.
32/4E1	The first byte of every four bytes in the buffer is programmed into the device.
32/4O1	The second byte of every four bytes in the buffer is programmed into the device.
32/4E2	The third byte of every four bytes in the buffer is programmed into the device.
32/4O2	The fourth byte of every four bytes in the buffer is programmed into the device.

This method of address splitting also allows the complete original file to be recreated in the buffer when reading from a set of master EPROMs.

A silicon signature is stored in all NEC devices and contains information on the device type, start and stop addresses, and programming voltages. The PG-1500 can read the silicon signature of the particular device being programmed either manually or automatically, or the device code can be entered manually.

The EDIT commands initialize the PG-1500 buffer to a known value, move a block of data from one location to another, and change/display data at a particular address. The PG-1500 buffer can also be searched for all occurrences of any 1-, 2-, or 4-byte pattern. Finally, a checksum can be calculated for all the data contained in the buffer.

The FUNCTION commands control the setup of the RS-232C serial port, whether the PG-1500 checks for a PROM insertion error, whether the PG-1500 is operated through the serial port, and how data is input/output from the PG-1500. Data can be input to the PG-1500 through either the RS-232C serial port or the Centronics compatible parallel port in Intel Extended Hex, Extended Tektronix Hex, or Motorola S formats. Data can also be output via the RS-232C port in any of these three formats.

Remote Control Mode

Table 3 lists the PG-1500 commands available in Remote Control Mode.

Table 3. PG-1500 Commands in Remote Control Mode

Command	Function
RR	Reads data from the EPROM
RS	Selects the EPROM to be used
RV	Verifies EPROM contents against PG-1500 buffer
RW	Writes data into EPROM
RZ	Checks if EPROM is blank
MC	Change the contents of the PG-1500 buffer
MD	Displays the contents of the PG-1500 buffer
MF	Initializes the PG-1500 buffer
PI	Inputs data from parallel port (Intel Extended HEX)
PM	Inputs data from parallel port (Motorola S)
PT	Inputs data from parallel port (Extended Tektronix HEX)
LI	Inputs data from serial port (Intel Extended HEX)
LM	Inputs data from serial port (Motorola S)
LT	Inputs data from serial port (Extended Tektronix HEX)
SI	Outputs data from serial port (Intel Extended HEX)
SM	Outputs data from serial port (Motorola S)
ST	Outputs data from serial port (Extended Tektronix HEX)
??	Help command

Host Controller Program

The PG-1500 can be controlled from an IBM PC series host computer using the accompanying PG-1500 controller program. The controller program has three modes of operation: control mode, auto mode, and terminal mode.

In the control mode, commands to be executed and parameters to be changed are selected from a screen display using the cursor control keys. The PG-1500 can be automatically configured from information contained in a optional configuration file. This file specifies the name of the file to be loaded, the ROM device, the address splitting mode, the HEX file format and whether the serial or parallel port is to be used for loading the data .

In auto mode, the controller program reads in the configuration file, configures itself accordingly, checks the ROM device, loads the file, writes the ROM and returns to the operating system when 1 set of ROM devices is completed.

In the terminal mode, all of the remote control commands listed in Table 3 are available for entry at the prompt. An additional operating system shell (OS) command allows execution of MS-DOS® programs without termination of the controller program. This OS command is also available in the control mode.

MS-DOS is a registered trademark of Microsoft Corporation

Equipment Supplied

The PG-1500 package includes the following:

- PG-1500 EPROM Programmer Base Unit
- 027A Socket Board for 27XXX EPROMS and μPD27C256A-like devices
- 04A Interface Board for NEC μPD75XX/μPD75XXX Microcomputers
- PG-1500 Controller Program Disk for IBM PC
- Power Cord
- Power Ground Plug Adapter
- Spare Fuses (2)
- PG-1500 EPROM Programmer User's Manuals
- Warranty Policy and Registration Card

Basic Specifications

- Power requirements:
  - 90 to 250 VAC, 50 to 60 Hz
- Environment conditions:
  - Operating temperature range: 10 to 35°C
  - Operating humidity range: 20 to 80% relative humidity
- RS-232C serial port:
  - Baud rates: 1200, 2400, 4800, 9600, 19200
  - Parity: none, even, odd
  - X-ON/X-OFF: on, off
  - Bit configuration: 7, 8
  - Stop bits: 1, 2

Documentation

For further information on the operation of the PG-1500, NEC provides the following documentation:

- PG-1500 EPROM Programmer User's Manual
- PG-1500 Controller Program User's Manual (IBM PC Based)

Section 10
Package Drawings

Package/Device Cross Reference

Package	Device, µPD
20-Pin Plastic Shrink DIP	7554CS 7554ACS 75P54CS 7564CS 7564ACS 75P64CS
20-Pin Plastic SOP	7554G 7554AG 75P54G 7564G 7564AG 75P64G
24-Pin Plastic Shrink DIP	7556CS 7556ACS 75P56CS 7566CS 7566ACS 75P66CS
24-Pin Plastic SOP (300 mil)	7556G 7556AG 75P56G 7566G 7566AG 75P66G
40-Pin Plastic DIP	7507C 7507HC 7508C 7508HC
40-Pin Plastic Shrink DIP	7507CU 7507HCU 7508CU 7508HCU
40-Pin Ceramic Piggyback DIP	75CG08E 75CG08HE
42-Pin Plastic DIP	7527AC 7528AC 7533C 7537AC 7538AC
42-Pin Plastic Shrink DIP	7527ACU 7528ACU 7533CU 7537ACU 7538ACU 7500xCU 75P008CU
42-Pin Ceramic Piggyback DIP	75CG28E 75CG33E 75CG38E
44-Pin Ceramic LCC (w/window)	71P301KA
44-Pin Plastic QFP	7507HGB 7508HGB 7500xGB 75P008GB 7533G

Package	Device, µPD	
44-Pin PLCC	71P301L	
52-Pin Plastic QFP (1.8-mm leads)	7507GC 7508GC	
52-Pin Plastic QFP (3.5-mm leads)	7225G	
64-Pin Shrink CERDIP (w/ 350-mil window)	75P108DW 78CP14DW 78P312ADW	
64-Pin Shrink CERDIP (w/ 300-mil window)	78P214DW	
64-Pin Plastic Shrink DIP	75028CW 75P036CW 75048CW 75P056CW 751xxCW 75P108CW 75P108BCW 75P116CW 75206CW 75208CW 75212ACW 75216ACW 75P216ACW 75268CW	78C10CW 78C10ACW 78C11CW 78C11ACW 78C12ACW 78C14CW 78CP14CW 78213CW 78214CW 78P214CW 78310ACW 78312ACW 78P312ACW
64-Pin Ceramic LCC (w/window)	71P301KB	
64-Pin Ceramic Piggyback Shrink DIP	75CG208E 75CG216AE	
64-Pin Ceramic Piggyback QUIP	78CG14E	
64-Pin Ceramic Piggyback QFP	75CG208EA 75CG216AEA	
64-Pin Plastic QFP (2.55 mm thick)	75028GC 75P036GC 75048GC 75P056GC 75104AGC 75108AGC 78C14AG	
64-Pin Plastic QFP (1.5 mm thick)	75108AG	
64-Pin Plastic QFP (2.7 mm thick)	71P301GF 7502GF 7503GF 751xxGF 75P108BGF 75P116GF 75206GF 75208GF 75212AGF 75216AGF 75268GF	78C10GF 78C10AGF 78C11GF 78C11AGF 78C12AGF 78C14GF 78CP14GF 78310AGF 78312AGF 78P312AGF

Package/Device Cross Reference (cont)

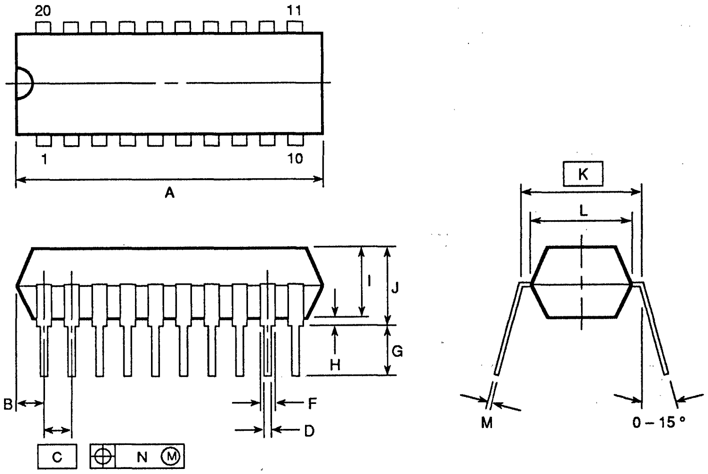
Package	Device, µPD	
64-Pin Plastic QFP (2.05 mm thick)	7227G 75104G 75106G 75108G 75P108G 75206G 75208G 78C10G-1B 78C11G-1B 78C14G-1B	
64-Pin Ceramic QUIP (w/window)	71P301RQ 78CP14R 78P214R 78P312AR	
64-Pin Plastic QUIP	71P301GQ 78C10G-36 78C10AGQ-36 78C11G-36 78C11AGQ-36 78C12AG-36 78C14G-36 78CP14G-36	78213GQ 78214GQ 78P214GQ 78310AGQ 78312AGQ 78P312AGQ
68-Pin PLCC	78C10L 78C10AL 78C11L 78C11AL 78C12AL 78C14L 78CP14L 78213L 78214L 78P214L	78310AL 78312AL 78P312AL 78320L 78322L
74-Pin Plastic QFP	78213GJ 78214GJ 78P214GJ 78320GJ 78322GJ	

Package	Device, µPD
80-Pin Ceramic LCC (w/window)	75P308K 75P316AK
80-Pin Plastic QFP (14 by 14 mm)	75328GC 75P328GC 78233GC 78234GC 78P238GC
80-Pin Plastic QFP (20 by 14 mm; 1.8-mm leads)	753xxGF 75P308GF 75P316GF 75P316AGF
80-Pin Plastic QFP (20 by 14 mm; 2.35-mm leads)	7228G 7228AG
84-Pin Plastic PLCC	78220L 78224L 78P224L 78233LQ 78234LQ 78P238LQ
94-Pin Ceramic LCC (w/window)	78P238KF
94-Pin Plastic QFP	78220GJ 78224GJ 78P224GJ 78233GJ 78234GJ 78P238GJ

20-Pin Plastic Shrink DIP

Item	Millimeters	Inches
A	19.57 max	.771 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.50 ±0.10	.020 $^{+.004}_{-.005}$
F	0.85 min	.033 min
G	3.2 ±0.3	.126 ±.012
H	0.51 min	.020 min
I	4.31 max	.170 max
J	5.08 max	.200 max
K*	7.62 (TP)	.300 (TP)
L	6.5	.256
M	0.25 $^{+0.10}_{-0.05}$	.010 $^{+.004}_{-.003}$
N	0.17	.007

* Item K to center of leads when formed parallel.

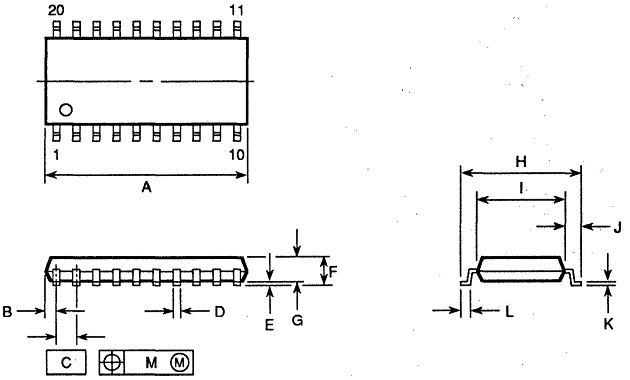


P20C-70-300B

49NR-593B (9/89)

20-Pin Plastic SOP

Item	Millimeters	Inches
A	13.00 max	.512 max
B	0.78 max	.031 max
C	1.27 (TP)	.050 (TP)
D	0.40 $^{+0.10}_{-0.05}$	.016 $^{+.004}_{-.003}$
E	0.1 ±0.1	.004 ±.004
F	1.8 max	.071 max
G	1.55	.061
H	7.7 ±0.3	.303 ±.012
I	5.6	.220
J	1.1	.043
K	0.20 $^{+0.10}_{-0.05}$	.008 $^{+.004}_{-.002}$
L	0.6 ±0.2	.024 $^{+.008}_{-.009}$
M	0.12	.005



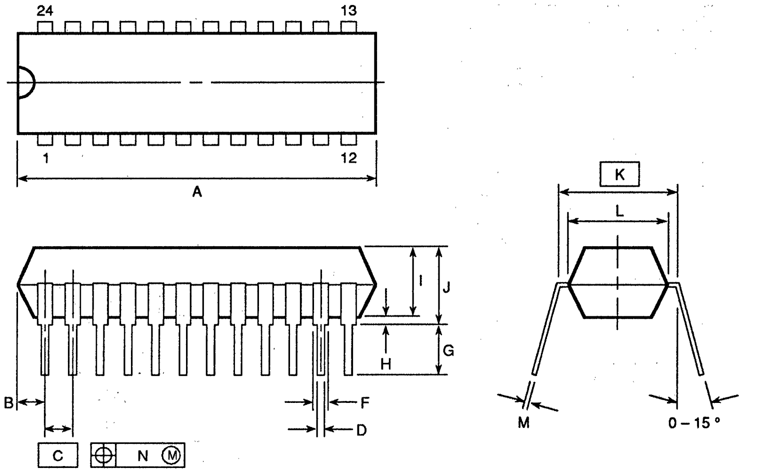
P20GM-50-300B, C

49NR-594B (9/89)

24-Pin Plastic Shrink DIP

Item	Millimeters	Inches
A	23.12 max	.911 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.50 ±0.10	.020 $^{+.004}_{-.005}$
F	0.85 min	.033 min
G	3.2 ±0.3	.126 ±.012
H	0.51 min	.020 min
I	4.31 max	.170 max
J	5.08 max	.200 max
K *	7.62 (TP)	.300 (TP)
L	6.5	.256
M	0.25 $^{+0.10}_{-0.05}$	.010 $^{+.004}_{-.003}$
N	0.17	.007

* Item K to center of leads when formed parallel.

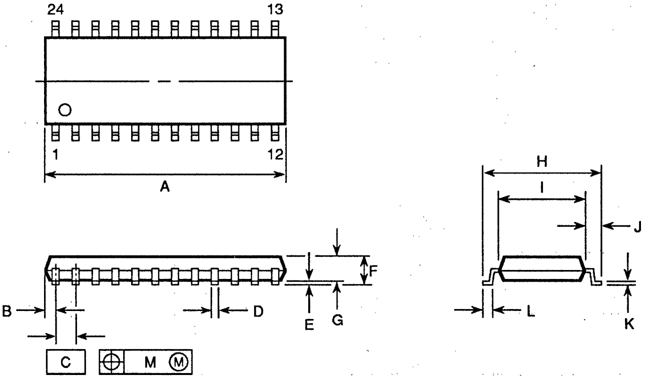


S24C-70-300B

49NR-596B (9/89)

24-Pin Plastic SOP (300 mil)

Item	Millimeters	Inches
A	15.54 max	.612 max
B	0.78 max	.031 max
C	1.27 (TP)	.050 (TP)
D	0.40 $^{+0.10}_{-0.05}$	.016 $^{+.004}_{-.003}$
E	0.1 ±0.1	.004 ±.004
F	1.8 max	.071 max
G	1.55	.061
H	7.7 ±0.3	.303 ±.012
I	5.6	.220
J	1.1	.043
K	0.20 $^{+0.10}_{-0.05}$	.008 $^{+.004}_{-.002}$
L	0.6 ±0.2	.024 $^{+.008}_{-.009}$
M	0.12	.005

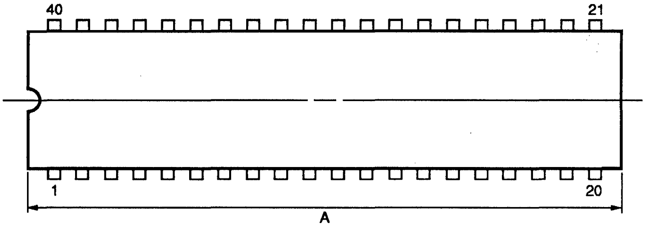


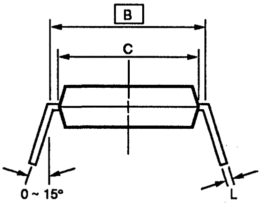
P24GM-50-300B

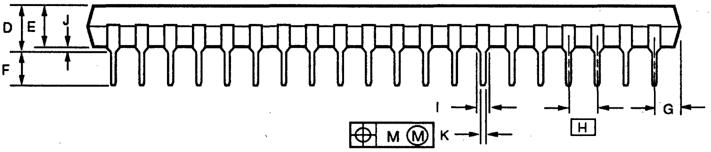
49NR-595B (9/89)

40-Pin Plastic DIP

Item	Millimeters	Inches
A	53.34 max	2.100 max
B	15.24 [TP]	.600 [TP]
C	13.2	.520
D	5.72 max	.225 max
E	4.31 max	.170 max
F	3.6 ±0.3	.142 ±.012
G	2.54 max	.100 max
H	2.54 [TP]	.100 [TP]
I	1.2 min	.047 min
J	0.51 min	.020 min
K	0.50 ±0.10	.020 ±.004
L	0.25 $^{+0.10}_{-0.05}$	.010 $^{+.004}_{-.002}$
M	0.25	.010







P40C-100-600A

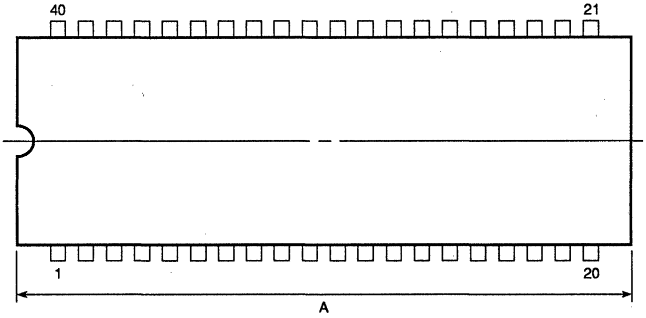
83vQ-6140B (6/89)

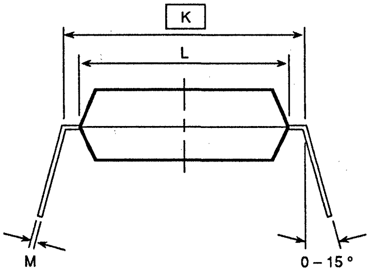
10

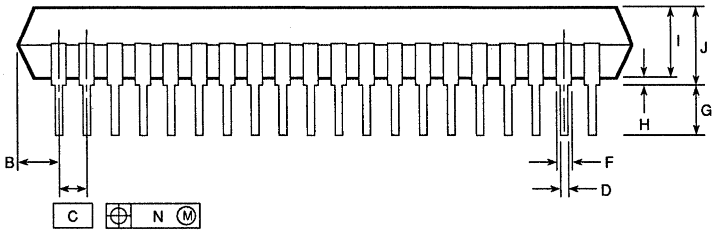
40-Pin Plastic Shrink DIP

Item	Millimeters	Inches
A	39.13 max	1.541 max
B	2.67 max	.106 max
C	1.778 (TP)	.070 (TP)
D	0.50 ±0.10	.020 $^{+.004}_{-.005}$
F	0.9 min	.035 min
G	3.2 ±0.3	.126 ±.012
H	0.51 min	.020 min
I	4.31 max	.170 max
J	5.08 max	.200 max
K *	15.24 (TP)	.600 (TP)
L	13.2	.520
M	0.25 $^{+0.10}_{-0.05}$	.010 $^{+.004}_{-.003}$
N	0.17	.007

* Item K to center of leads when formed parallel.





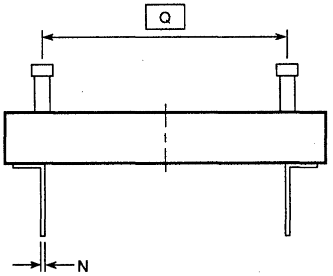


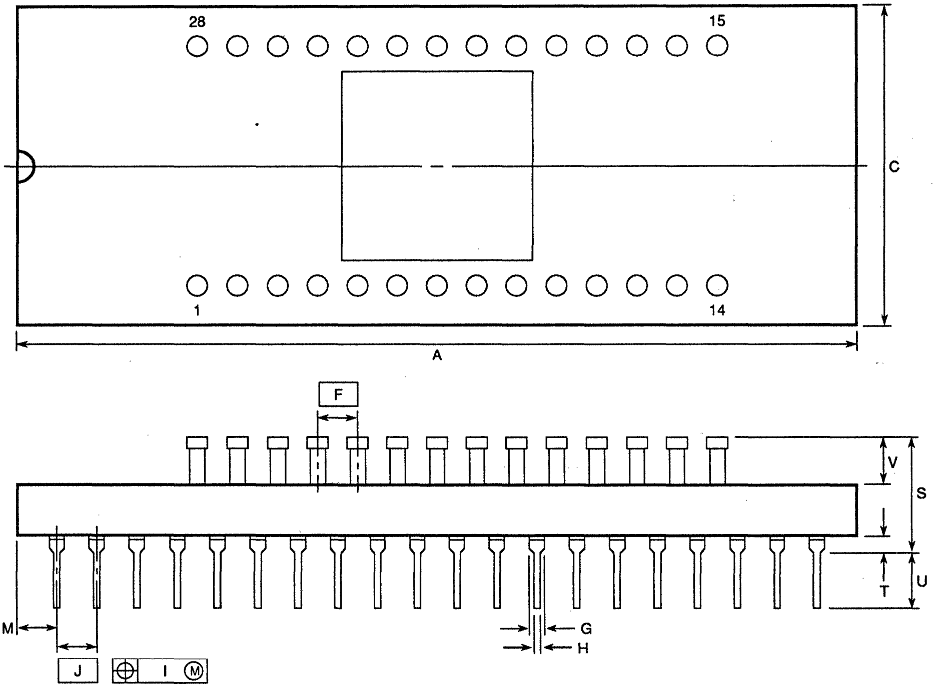
P40C-70-600A

49NR-544B (7/89)

40-Pin Ceramic Piggyback DIP

Item	Millimeters	Inches
A	53.34 max	2.100 max
C	20.32 ±0.4	.800 ±.016
F	2.54	.100
G	0.92 min	.036 min
H	0.46 ±0.05	.018 ±.002
I	0.25	.010
J	2.54 (TP)	.100 (TP)
M	2.54 max	.100 max
N	0.25 ±0.05	$.010\ ^{+.002}_{-.003}$
Q	15.24	.600
S	7.28 max	.287 max
T	1.0 min	.039 min
U	3.5 ±0.3	.138 ±.012
V	3.0 max	.118 max



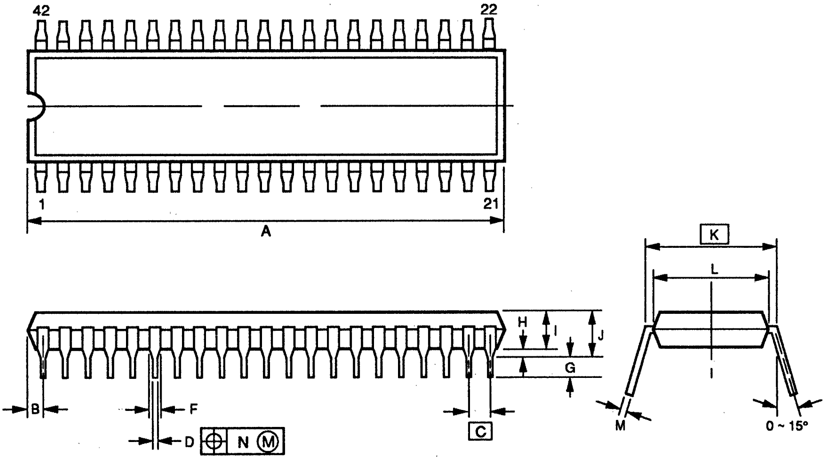


P40E-100-A

49NR-597B (9/89)

42-Pin Plastic DIP

Item	Millimeters	Inches
A	55.88 max	2.2 max
B	2.54 max	.100 max
C	2.54 [TP]	.100 [TP]
D	0.50 ± 0.10	$.020^{+.004}_{-.005}$
F	1.2 min	.047 min
G	3.6 ±0.30	.142 ±.012
H	0.51 min	.020 min
I	4.31 max	.170 max
J	5.72 max	.226 max
K	15.24 [TP]	.600 [TP]
L	13.2	.520
M	$0.25^{+0.10}_{-0.05}$	$.010^{+.004}_{-.003}$
N	0.25	.01



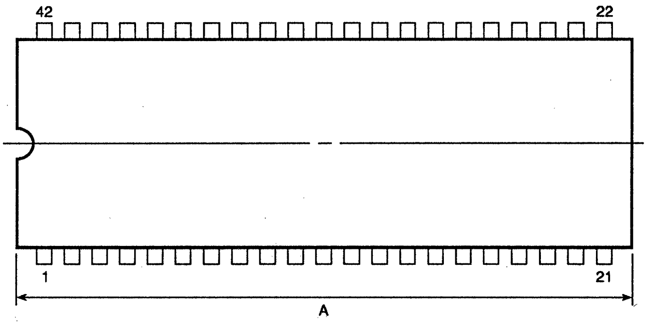
P42C-100-600A, B

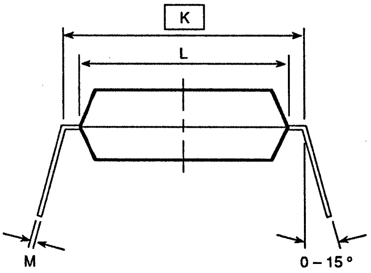
83IH-6028B (5/90)

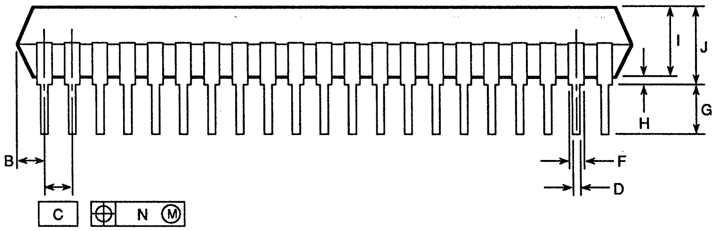
42-Pin Plastic Shrink DIP

Item	Millimeters	Inches
A	39.13 max	1.541 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.50 ± 0.10	$.020^{+.004}_{-.005}$
F	0.9 min	.035 min
G	3.2 ± 0.3	.126 ± .012
H	0.51 min	.020 min
I	4.31 max	.170 max
J	5.08 max	.200 max
K *	15.24 (TP)	.600 (TP)
L	13.2	.520
M	$0.25^{+0.10}_{-0.05}$	$.010^{+.004}_{-.003}$
N	0.17	.007

* Item K to center of leads when formed parallel.





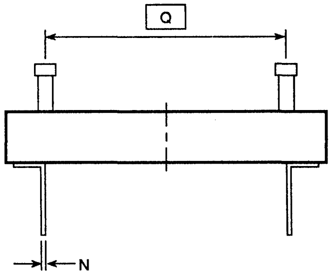


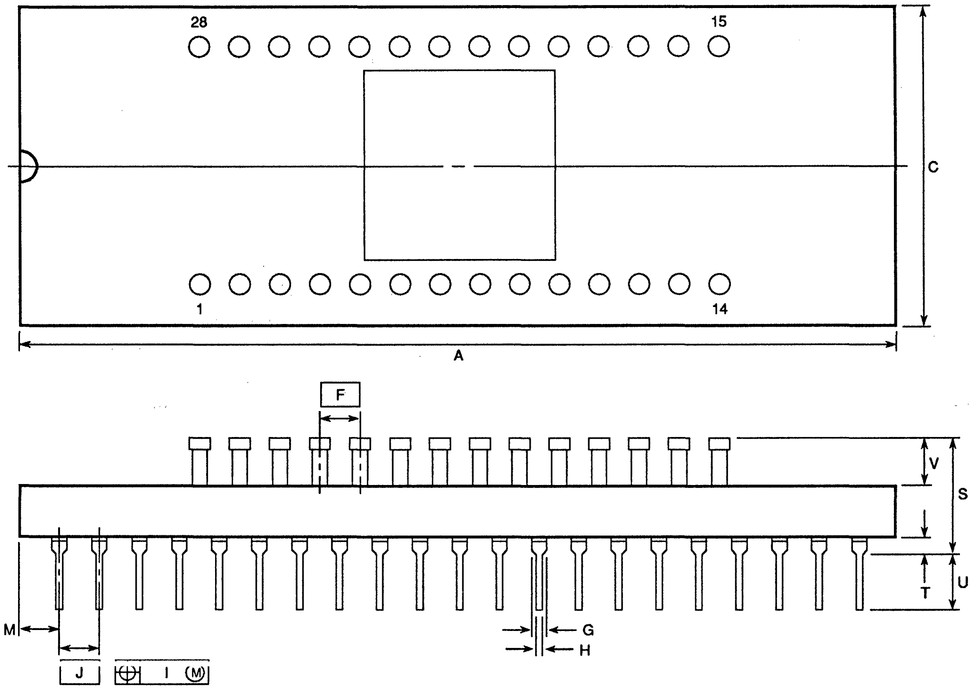
P42C-70-600A

49NR-557B (7/89)

42-Pin Ceramic Piggyback DIP

Item	Millimeters	Inches
A	55.88 max	2.200 max
C	20.32 ±0.4	.800 ±.016
F	2.54	.100
G	0.92 min	.036 min
H	0.46 ±0.05	.018 ±.002
I	0.25	.010
J	2.54 (TP)	.100 (TP)
M	2.54 max	.100 max
N	0.25 ±0.05	.010 $^{+.002}_{-.003}$
Q	15.24	.600
S	7.28 max	.287 max
T	1.0 min	.039 min
U	3.5 ±0.3	.138 ±.012
V	3.0 max	.118 max



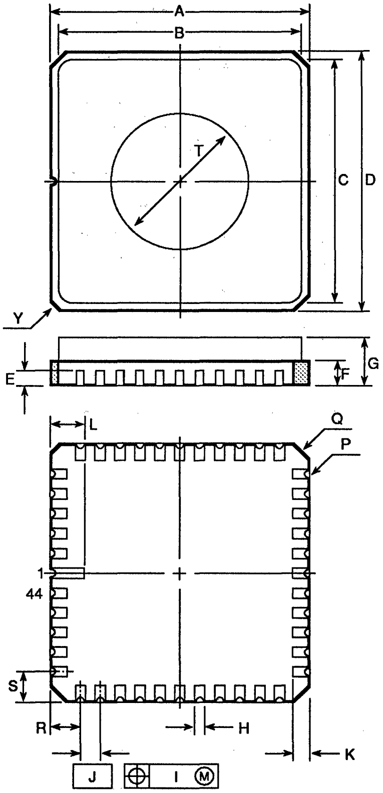


P42E-100-A

49NR-598B (9/89)

44-Pin Ceramic LCC (w/window)

Item	Millimeters	Inches
A	16.51 ±0.4	.650 ±.016
B	15.50	.610
C	15.50	.610
D	16.51 ±0.4	.650 ±.016
E	1.02	.040
F	1.52	.060
G	3.048 max	.120 max
H	0.64 ±0.10	.025 ±.004
I	0.12	.005
J	1.27 (TP)	.050 (TP)
K	1.27 ±0.2	.050 ±.008
L	2.16 ±0.2	.085 ±.008
P	0.2 rad	.008 rad
Q	1.02 cor	.040 cor
R	1.905	.075
S	1.905	.075
T	8.89 dia	.350 dia
Y	0.51 cor	.020

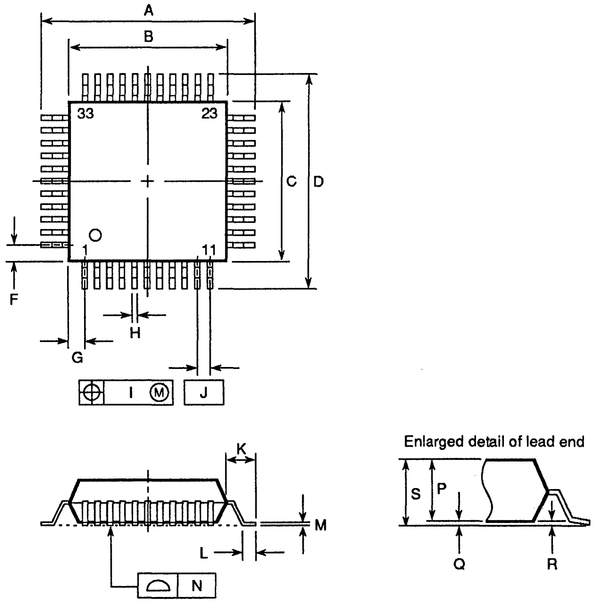


X44KW-50A-1

49NR-698B (3/90)

44-Pin Plastic QFP

Item	Millimeters	Inches
A	13.6 ±0.4	$.535^{+.017}_{-.016}$
B	10.0 ±0.2	$.394^{+.008}_{-.009}$
C	10.0 ±0.2	$.394^{+.008}_{-.009}$
D	13.6 ±0.4	$.535^{+.017}_{-.016}$
F	1.0	.039
G	1.0	.039
H	0.35 ±0.10	$.014^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.8 ±0.2	$.071^{+.008}_{-.009}$
L	0.8 ±0.2	$.031^{+.009}_{-.008}$
M	$0.15^{+0.10}_{-0.05}$	$.006^{+.004}_{-.003}$
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.119 max

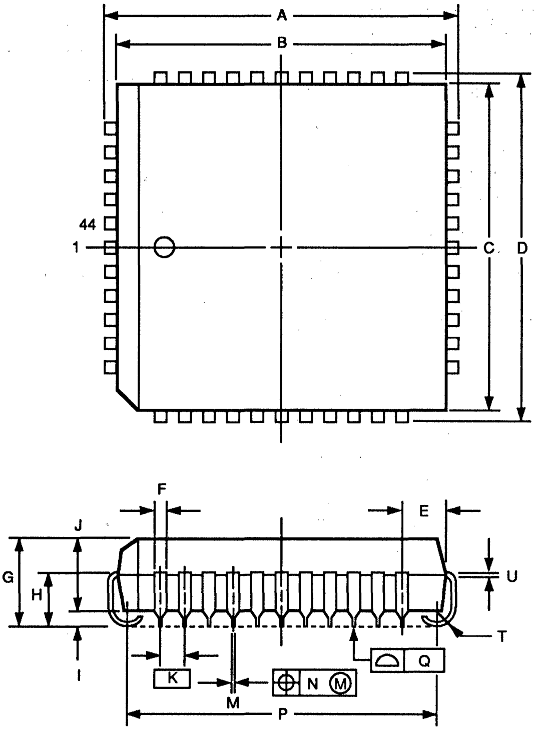


P44GB-80-3B4-1

49NR-556B (1/90)

44-Pin PLCC

Item	Millimeters	Inches
A	17.5 ±0.2	.689 ±.008
B	16.58	.653
C	16.58	.653
D	17.5 ±0.2	.689 ±.008
E	1.94 ±0.15	.076 ±.006
F	0.6	.024
G	4.4 ±0.2	.173 ±.008
H	2.8 ±0.2	.110 ±.008
I	0.9 min	.035 min
J	3.4	.134
K	1.27 (TP)	.050 (TP)
M	0.40 ±0.10	.016 ±.004
N	0.12	.005
P	15.50 ±0.20	.610 ±.008
Q	0.15	.006
T	0.8 radius	.031 radius
U	0.20 $^{+0.10}_{-0.05}$	.008 $^{+.004}_{-.002}$

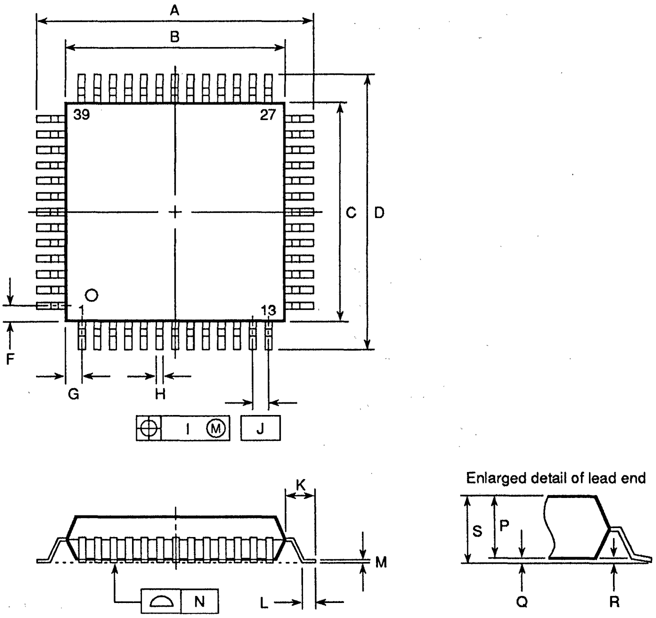


P44L-50A1-1

3/90 83YL-5804B

52-Pin Plastic QFP (1.8-mm leads)

Item	Millimeters	Inches
A	17.6 ±0.4	.693 ±.016
B	14.0 ±0.2	$.551^{+.009}_{-.008}$
C	14.0 ±0.2	$.551^{+.009}_{-.008}$
D	17.6 ±0.4	.693 ±.016
F	1.0	.039
G	1.0	.039
H	0.40 ±0.10	$.016^{+.004}_{-.005}$
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	1.8 ±0.2	$.071^{+.008}_{-.009}$
L	0.8 ±0.2	$.031^{+.009}_{-.008}$
M	$0.15^{+0.10}_{-0.05}$	$.006^{+.004}_{-.003}$
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.119 max

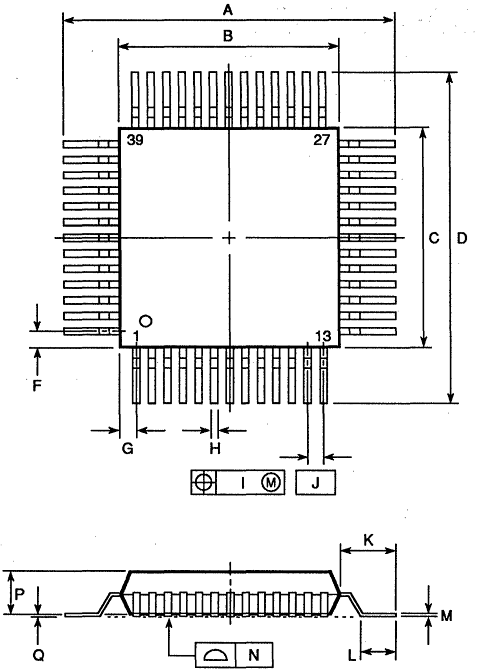


P52GC-100-3B6

49NR-493B (5/89)

52-Pin Plastic QFP (3.5-mm leads)

Item	Millimeters	Inches
A	21.0 ±0.4	.827 ±.016
B	14.0 ±0.2	$.551^{+.009}_{-.008}$
C	14.0 ±0.2	$.551^{+.009}_{-.008}$
D	21.0 ±0.4	.827 ±.016
F	1.0	.039
G	1.0	.039
H	0.40 ±0.10	$.016^{+.004}_{-.005}$
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	3.5 ±0.2	$.138^{+.008}_{-.009}$
L	2.2 ±0.2	$.087^{+.009}_{-.008}$
M	$0.15^{+0.10}_{-0.05}$	$.006^{+.004}_{-.003}$
N	0.15	.006
P	$2.6^{+0.2}_{-0.1}$	$.102^{+.009}_{-.004}$
Q	0.1 ±0.1	.004 ±.004



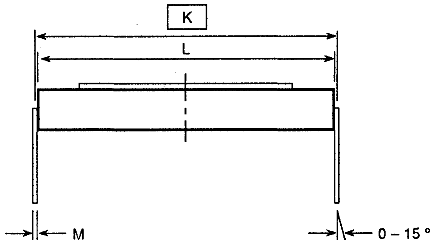
P52G-100-00

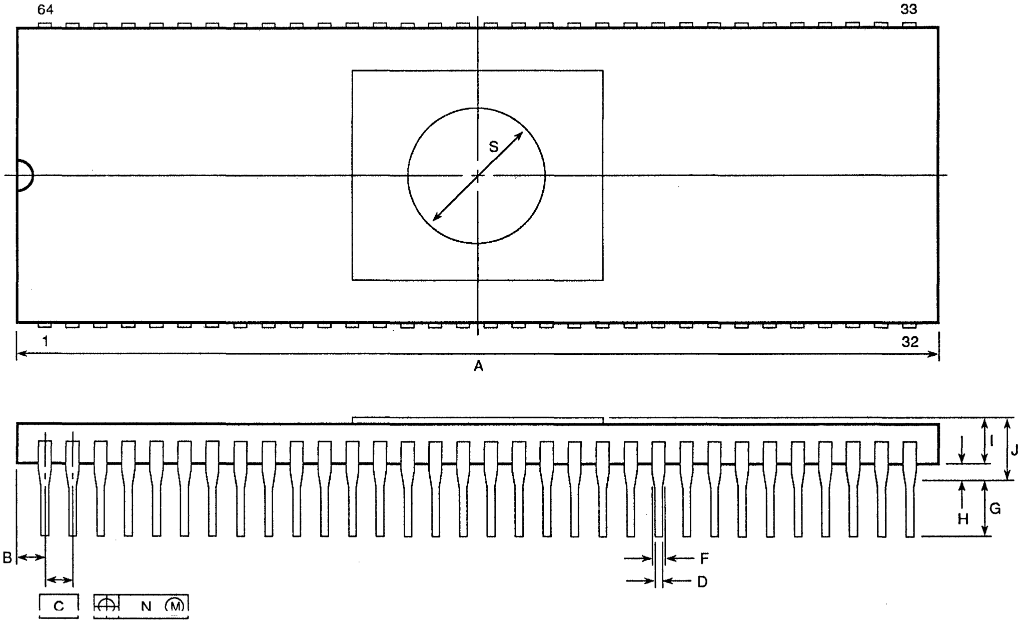
49NR-536B
(6/89)

64-Pin Shrink CERDIP (w/ 350-mil window)

Item	Millimeters	Inches
A	58.68 max	2.310 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.46 ±0.05	.018 ±.002
F	0.8 min	.031 min
G	3.5 ±0.3	.138 ±.012
H	1.0 min	.039 min
I	3.0	.118
J	5.08 max	.200 max
K*	19.05 (TP)	.750 (TP)
L	18.8	.740
M	0.25 ±0.05	.010 $^{+.002}_{-.003}$
N	0.25	.010
S	8.89 dia	.350 dia

* Item K to center of leads when formed parallel.





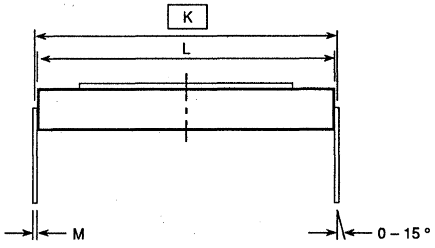
P64DW-70-750A

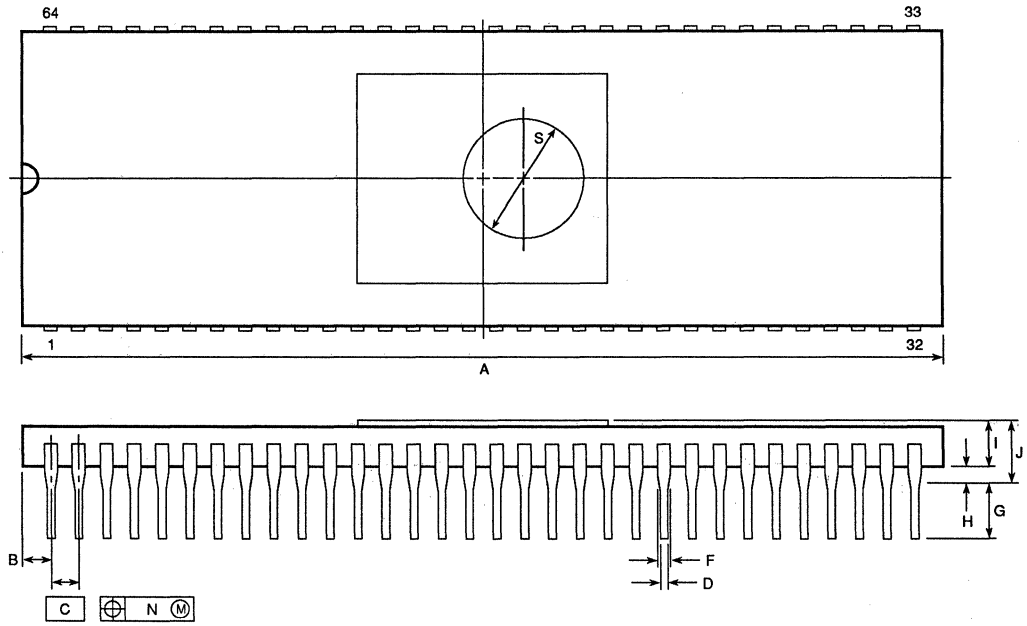
49NR-509B
(5/89)

64-Pin Shrink CERDIP (w/ 300-mil window)

Item	Millimeters	Inches
A	58.68 max	2.310 max
B	1.78 max	.070 max
C	1.778 (TP)	.070 (TP)
D	0.46 ±0.05	.018 ±.002
F	0.8 min	.031 min
G	3.5 ±0.3	.138 ±.012
H	1.0 min	.039 min
I	3.0	.118
J	5.08 max	.200 max
K *	19.05 (TP)	.750 (TP)
L	18.8	.740
M	0.25 ±0.05	.010 +.002 −.003
N	0.25	.010
S	7.62 dia	.300 dia

* Item K to center of leads when formed parallel.

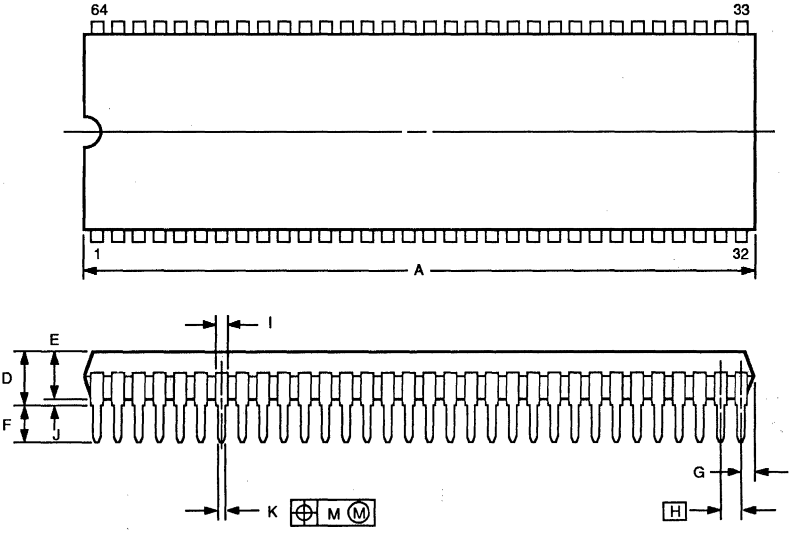




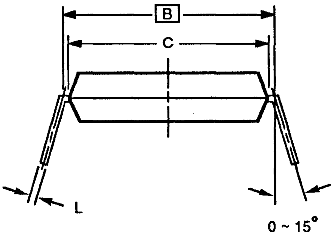
P64DW-70-750A1

49NR-691B (2/90)

64-Pin Plastic Shrink DIP



Item	Millimeters	Inches
A	58.68 max	2.310 max
B	19.05 (TP)	.750 (TP)
C	17.0	.669
D	5.08 max	.200 max
E	4.31 max	.170 max
F	3.2 ±0.3	.126 ±.012
G	1.78 max	.070 max
H	1.778 (TP)	.070 (TP)
I	0.9 min	.035 min
J	0.51 min	.020 min
K	0.50 ±0.10	.020 ±.004
L	$0.25^{+0.10}_{-0.05}$	$.010^{+.004}_{-.002}$
M	0.17	.007

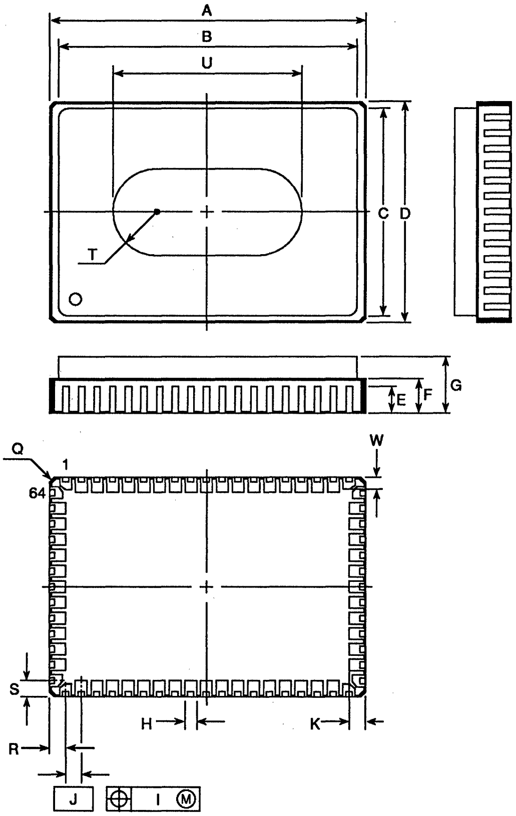


P64C-70-750A, C

83YL-5560B (5/89)

64-Pin Ceramic LCC (w/window)

Item	Millimeters	Inches
A	20.0 ±0.4	.787 ±.016
B	19.0	.748
C	13.2	.520
D	14.0 ±0.4	.550 ±.016
E	1.64	.065
F	2.14	.084
G	3.556 max	.140 max
H	0.70 ±0.10	.028 ±.004
I	0.1	.004
J	1.0 (TP)	.039 (TP)
K	1.0 ±0.2	.039 ±.008
Q	0.25 cor	.010 cor
R	1.0	.039
S	1.0	.039
T	3.0 rad	.118 rad
U	12.0	.472
W	0.8 ±0.2	.031 ±.008

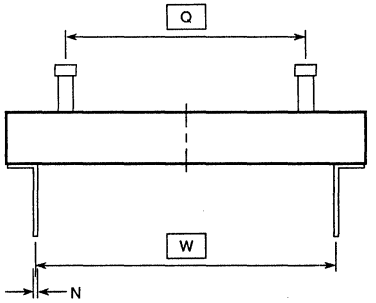


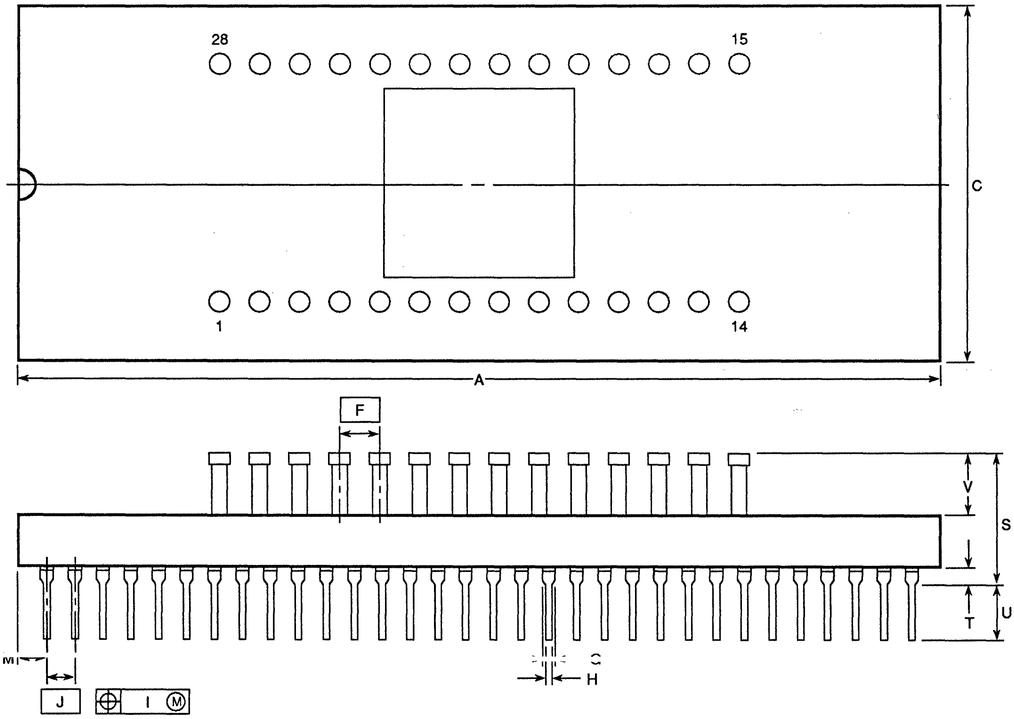
X64KW-100A-1

49NR-699B (5/90)

64-Pin Ceramic Piggyback Shrink DIP

Item	Millimeters	Inches
A	58.68 max	2.310 max
C	22.86 ±0.4	.900 ±.016
F	2.54	.100
G	0.80 min	.031 min
H	0.46 ±0.05	.018 ±.002
I	0.17	.007
J	1.778 (TP)	.070 (TP)
M	1.78 max	.070 max
N	0.25 ±0.05	.010 $^{+.002}_{-.003}$
Q	15.24	.600
S	8.28 max	.326 max
T	1.0 min	.039 min
U	3.5 ±0.3	.138 ±.012
V	3.9 max	.154 max
W	19.05	.750



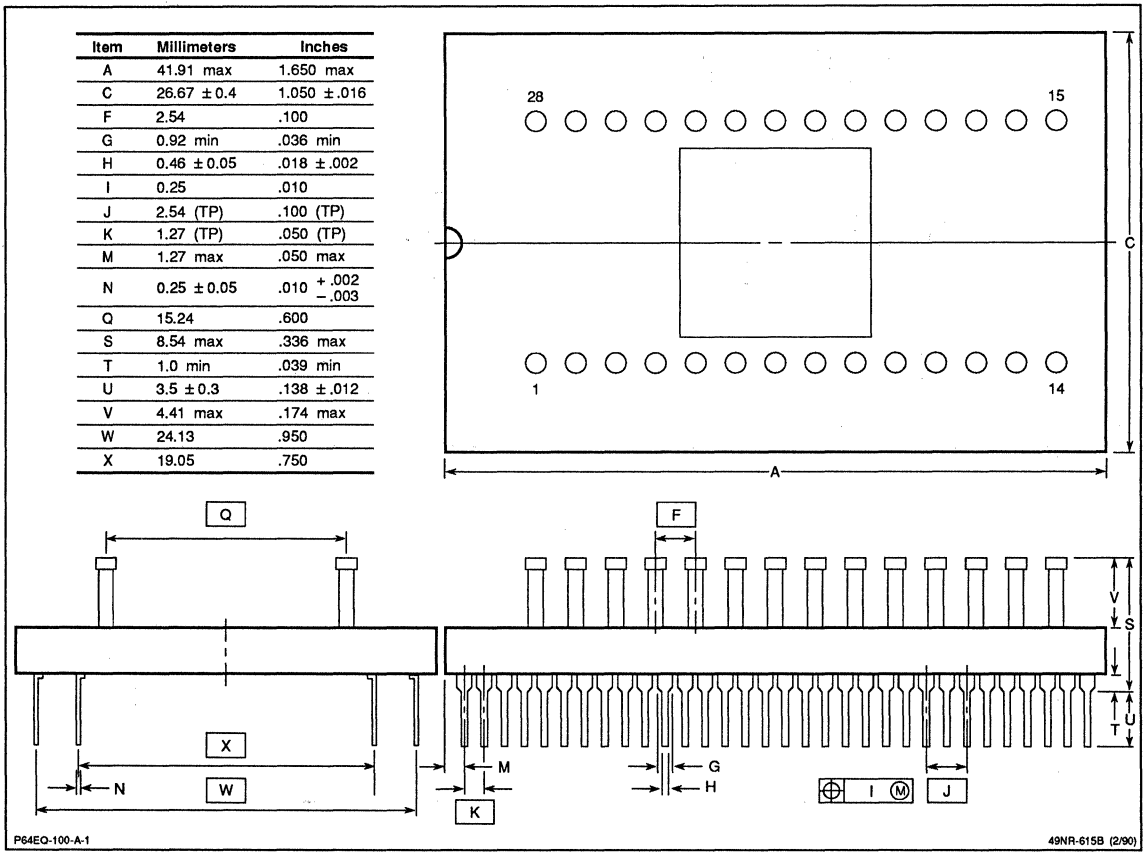


P64E-70-A

49NR-671B (1/90)

64-Pin Ceramic Piggyback QUIP

Item	Millimeters	Inches
A	41.91 max	1.650 max
C	26.67 ±0.4	1.050 ±.016
F	2.54	.100
G	0.92 min	.036 min
H	0.46 ±0.05	.018 ±.002
I	0.25	.010
J	2.54 (TP)	.100 (TP)
K	1.27 (TP)	.050 (TP)
M	1.27 max	.050 max
N	0.25 ±0.05	.010 $^{+.002}_{-.003}$
Q	15.24	.600
S	8.54 max	.336 max
T	1.0 min	.039 min
U	3.5 ±0.3	.138 ±.012
V	4.41 max	.174 max
W	24.13	.950
X	19.05	.750

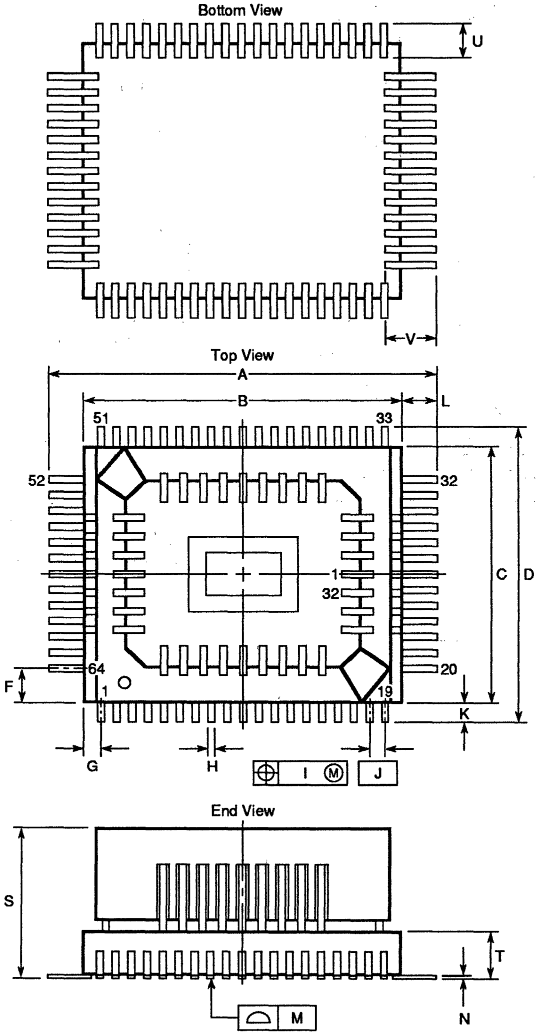


P64EQ-100-A-1

49NR-615B (2/90)

64-Pin Ceramic Piggyback QFP

Item	Millimeters	Inches
A	24.7 ±0.5	.972 +.021 −.020
B	20.3	.799
C	16.3	.642
D	18.7 ±0.5	.736 ±.020
E	1.27 (TP)	.050 (TP)
F	2.15	.085
G	1.15	.045
H	.40 ±0.10	.016 +.004 −.005
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	1.2 ±0.2	.047 +.009 −.008
L	2.2 ±0.2	.087 +.008 −.009
M	0.15	.006
N	0.15 ±0.05	.006 +.002 −.003
S	9.5 max	.374 max
T	3.0 max	.118 max
U	2.2 ±0.2	.087 +.008 −.009
V	3.2 ±0.2	.126 ±.008

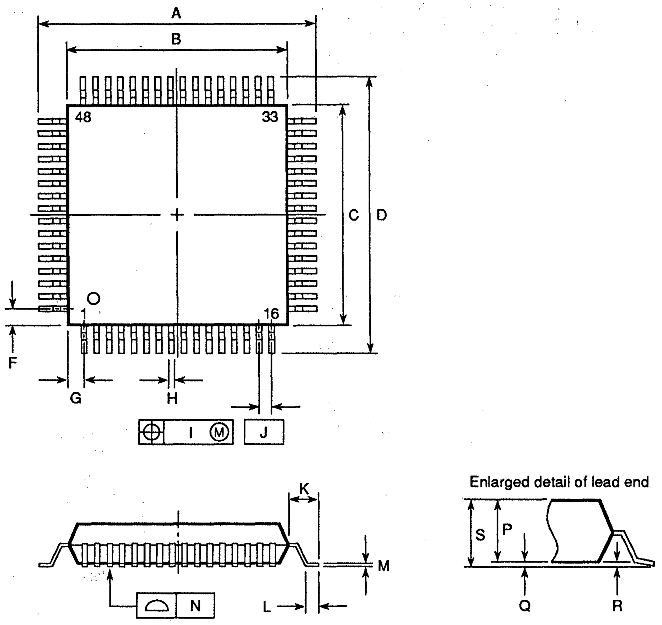


P64EA-100-A

49NR-672B (2/90)

64-Pin Plastic QFP (2.55 mm thick)

Item	Millimeters	Inches
A	17.6 ±0.4	.693 ±.016
B	14.0 ±0.2	$.551^{+.009}_{-.008}$
C	14.0 ±0.2	$.551^{+.009}_{-.008}$
D	17.6 ±0.4	.693 ±.016
F	1.0	.039
G	1.0	.039
H	0.35 ±0.10	$.014^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.8 ±0.2	.071 ±.008
L	0.8 ±0.2	$.031^{+.009}_{-.008}$
M	$0.15^{+0.10}_{-0.05}$	$.006^{+.004}_{-.003}$
N	0.15	.006
P	2.55	.100
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	2.85 max	.112 max

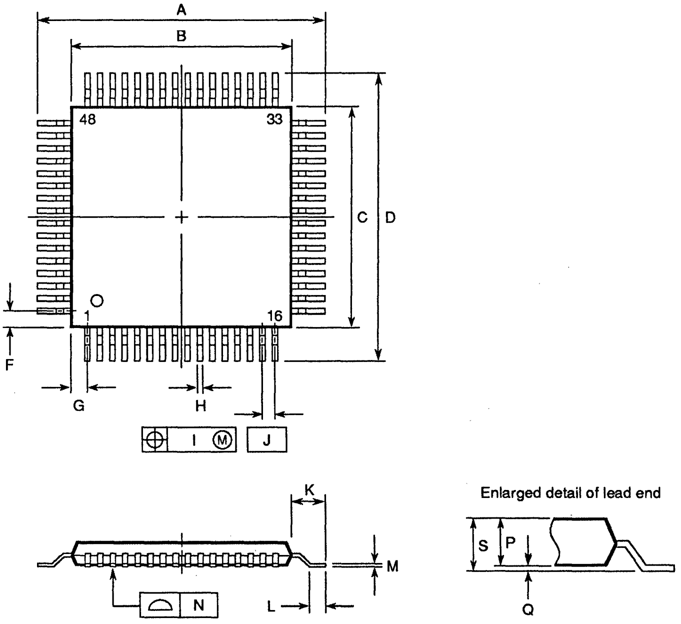


P64GC-80-AB8-1

49NR-669B (1/90)

64-Pin Plastic QFP (1.5 mm thick)

Item	Millimeters	Inches
A	18.4 ±0.4	.724 $^{+.017}_{-.016}$
B	14.0 ±0.2	.551 $^{+.009}_{-.008}$
C	14.0 ±0.2	.551 $^{+.009}_{-.008}$
D	18.4 ±0.4	.724 $^{+.017}_{-.016}$
F	1.0	.039
G	1.0	.039
H	0.35 ±0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	2.2 ±0.2	.087 $^{+.008}_{-.009}$
L	1.0 ±0.2	.039 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	1.5 ±0.1	.059 ±.004
Q	0.0 ±0.1	.000 ±.004
S	1.7 max	.067 max

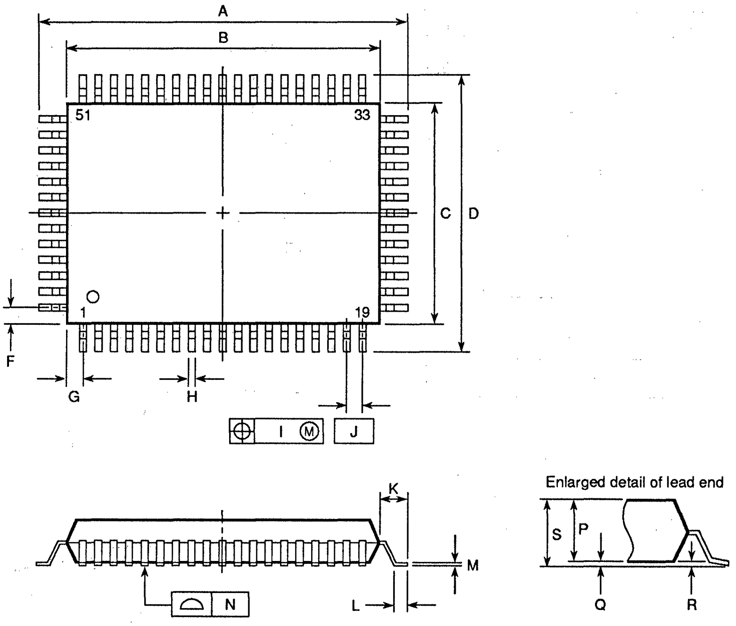


P64G-80-22-1

49NR-670B (1/90)

64-Pin Plastic QFP (2.7 mm thick)

Item	Millimeters	Inches
A	23.6 ±0.4	.929 ±.016
B	20.0 ±0.2	.795 +.009 −.008
C	14.0 ±0.2	.551 +.009 −.008
D	17.6 ±0.4	.693 ±.016
F	1.0	.039
G	1.0	.039
H	0.40 ±0.10	.016 +.004 −.005
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	1.8 ±0.2	.071 +.008 −.009
L	0.8 ±0.2	.031 +.009 −.008
M	0.15 +0.10 −0.05	.006 +.004 −.003
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.119 max

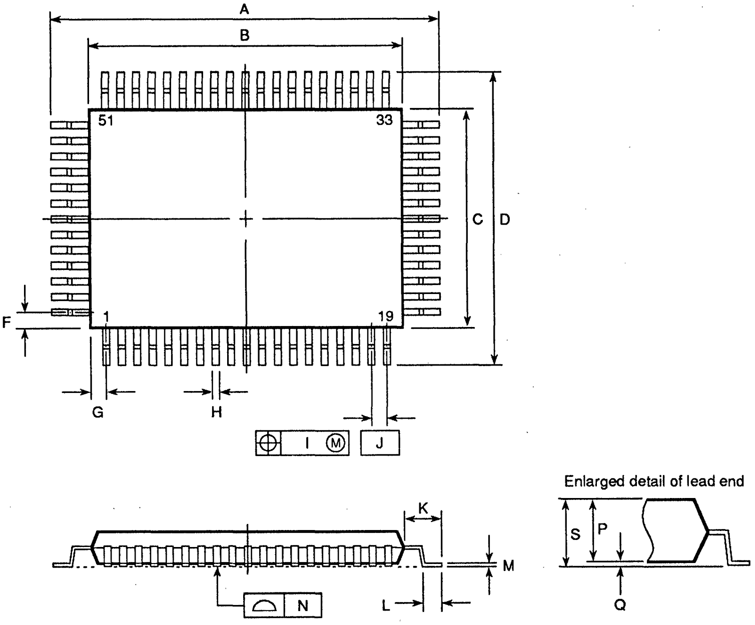


P64GF-100-3B8, 3BE-1

49NR-599B (2/90)

64-Pin Plastic QFP (2.05 mm thick)

Item	Millimeters	Inches
A	24.7 ±0.4	.972 +.017 −.016
B	20.0 ±0.2	.795 +.009 −.008
C	14.0 ±0.2	.551 +.009 −.008
D	18.7 ±0.4	.736 ±.016
F	1.0	.039
G	1.0	.039
H	0.40 ±0.10	.016 +.004 −.005
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	2.35 ±0.2	.093 +.008 −.009
L	1.2 ±0.2	.047 +.009 −.008
M	0.15 +0.10 −0.05	.006 +.004 −.003
N	0.15	.006
P	2.05 +0.2 −0.1	.081 +.008 −.005
Q	0.1 ±0.1	.004 ±.004
S	2.45 max	.096 max

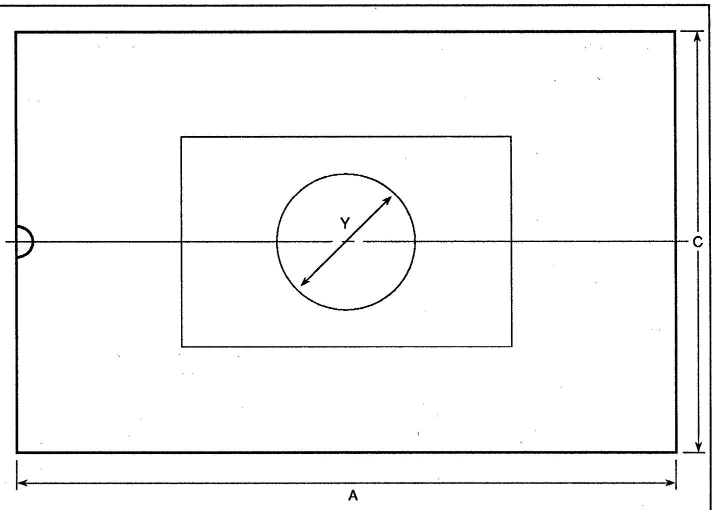


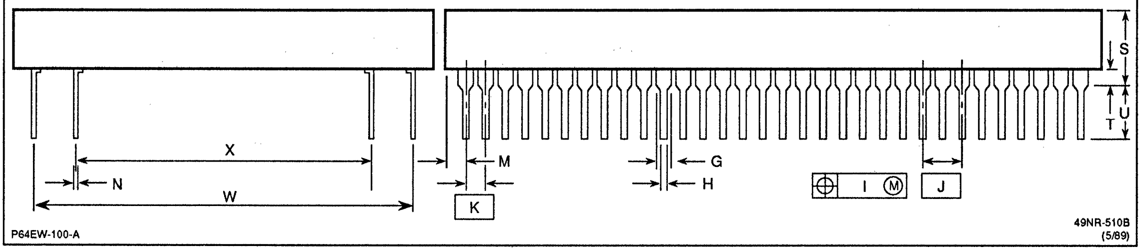
P64G-100-12, 1B-1

49NR-543B (2/90)

64-Pin Ceramic QUIP (w/window)

Item	Millimeters	Inches
A	41.91 max	1.650 max
C	26.67 ±0.4	1.050 ±.016
G	0.92 min	.036 min
H	0.46 ±0.05	.018 ±.002
I	0.25	.010
J	2.54 (TP)	.100 (TP)
K	1.27 (TP)	.050 (TP)
M	1.27 max	.050 max
N	0.25 ±0.05	.010 $^{+.002}_{-.003}$
S	4.72 max	.186 max
T	1.0 min	.039 min
U	3.5 ±0.3	.138 $^{+.012}_{-.013}$
W	24.13	.950
X	19.05	.750
Y	8.89 dia	.350 dia



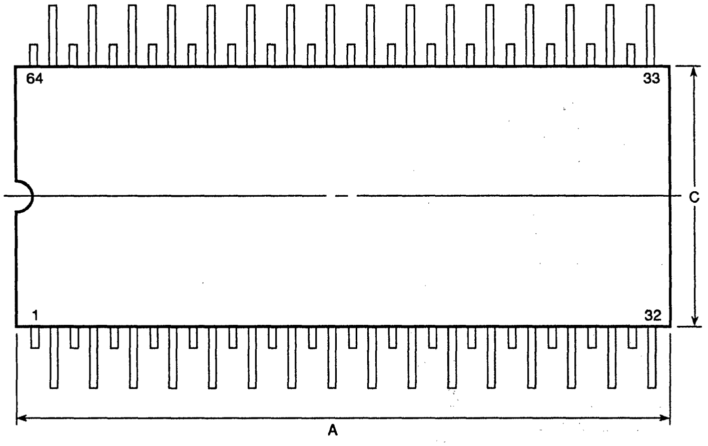


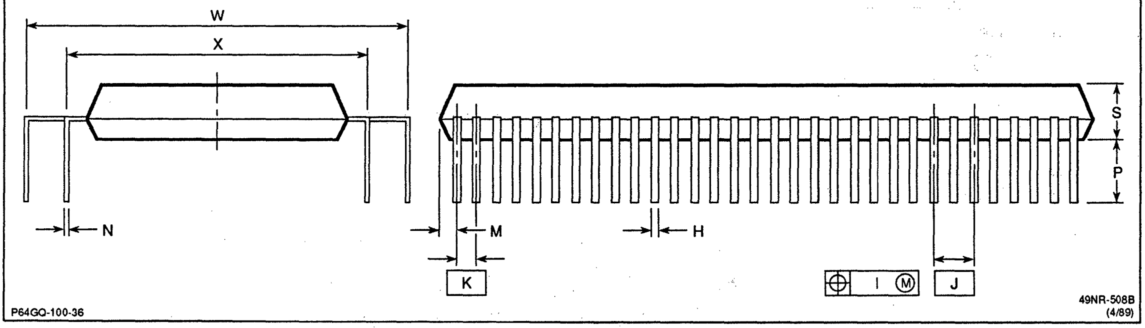
P64EW-100-A

49NR-510B
(5/89)

64-Pin Plastic QUIP

Item	Millimeters	Inches
A	$41.5\ ^{+0.3}_{-0.2}$	$1.634\ ^{+.012}_{-.008}$
C	16.5	.650
H	0.50 ±0.10	$.020\ ^{+.004}_{-.005}$
I	0.25	.010
J	2.54 (TP)	.100 (TP)
K	1.27 (TP)	.050 (TP)
M	$1.1\ ^{+0.25}_{-0.15}$	$.043\ ^{+.011}_{-.006}$
N	$0.25\ ^{+0.10}_{-0.05}$	$.010\ ^{+.004}_{-.003}$
P	4.0 ±0.3	$.157\ ^{+.013}_{-.012}$
S	3.6 ±0.1	$.142\ ^{+.004}_{-.005}$
W	24.13 ±1.05	.950 ±.042
X	19.05 ±1.05	.750 ±.042



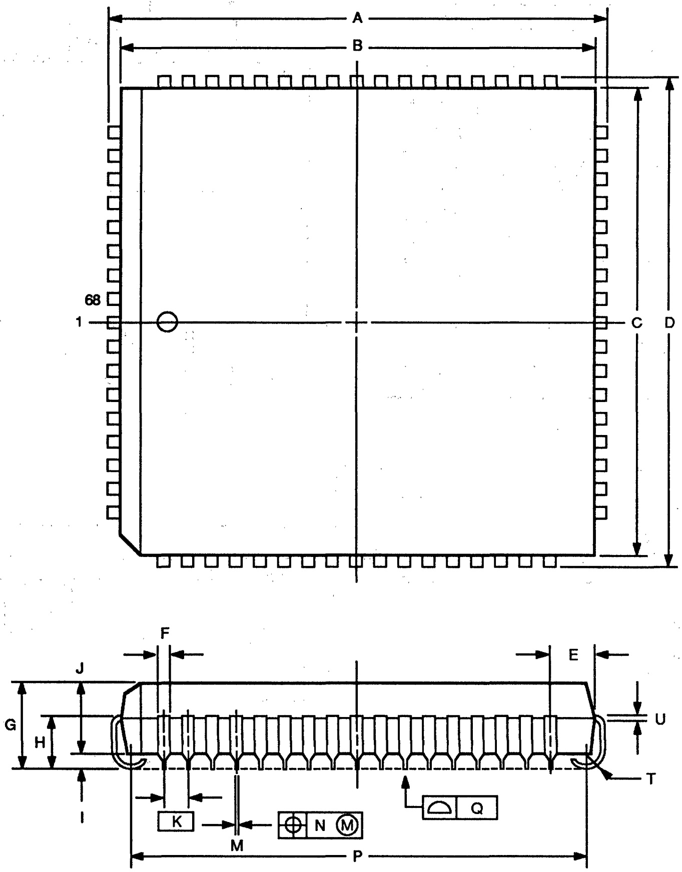


P64GQ-100-36

49NR-508B
(4/89)

68-Pin PLCC

Item	Millimeters	Inches
A	25.2 ±0.2	.992 ±.008
B	24.20	.953
C	24.20	.953
D	25.2 ±0.2	.992 ±.008
E	1.94 ±0.15	$.076^{+.007}_{-.006}$
F	0.6	.024
G	4.4 ±0.2	$.173^{+.009}_{-.008}$
H	2.8 ±0.2	$.110^{+.009}_{-.008}$
I	0.9 min	.035 min
J	3.4	.134
K	1.27 (TP)	.050 (TP)
M	0.40 ±0.10	$.016^{+.004}_{-.005}$
N	0.12	.005
P	23.12 ±0.20	$.910^{+.009}_{-.008}$
Q	0.15	.006
T	0.8 radius	.031 radius
U	$0.20^{+0.10}_{-0.05}$	$.008^{+.004}_{-.002}$

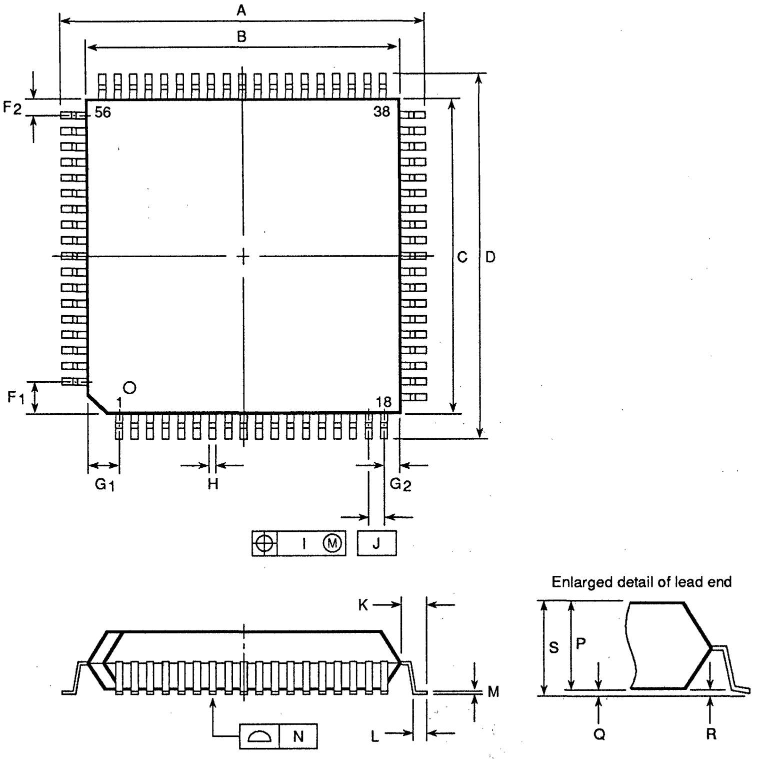


P68L-50A1-1

(2/90)
83YL-5561B

74-Pin Plastic QFP

Item	Millimeters	Inches
A	23.2 ±0.4	.913 +.017 −.016
B	20.0 ±0.2	.787 +.009 −.008
C	20.0 ±0.2	.787 +.009 −.008
D	23.2 ±0.4	.913 +.017 −.016
F_1	2.0	.079
F_2	1.0	.039
G_1	2.0	.079
G_2	1.0	.039
H	0.40 ±0.10	.016 +.004 −.005
I	0.20	.008
J	1.0 (TP)	.039 (TP)
K	1.6 ±0.2	.063 ±.002
L	0.8 ±0.2	.031 +.009 −.008
M	0.15 +0.10 −0.05	.006 +.004 −.005
N	0.15	.006
P	3.7	.146
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	4.0 max	.158 max

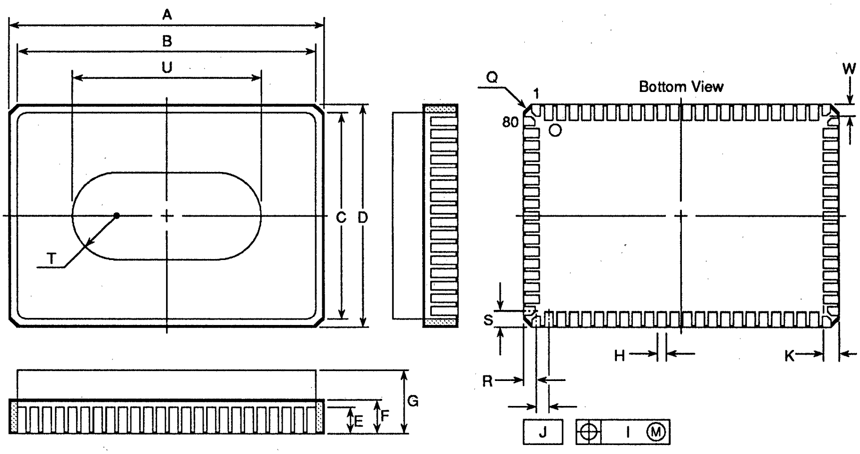


S74GJ-100-5BJ-1

49NR-347B (2/90)

80-Pin Ceramic LCC (w/window)

Item	Millimeters	Inches
A	20.0 ±0.4	$.787^{+.017}_{-.016}$
B	19.0	.748
C	13.2	.520
D	14.2 ±0.4	.559 ±.016
E	1.64	.065
F	2.14	.084
G	4.064 max	.160 max
H	0.51 ±0.10	.020 ±.004
I	0.08	.003
J	0.8 (TP)	.031 (TP)
K	1.0 ±0.2	$.039^{+.009}_{-.008}$
Q	0.5 cor	.020 cor
R	0.8	.031
S	1.1	.043
T	3.0 rad	.118 rad
U	12.0	.472
W	0.75 ±0.2	$.030^{+.008}_{-.009}$

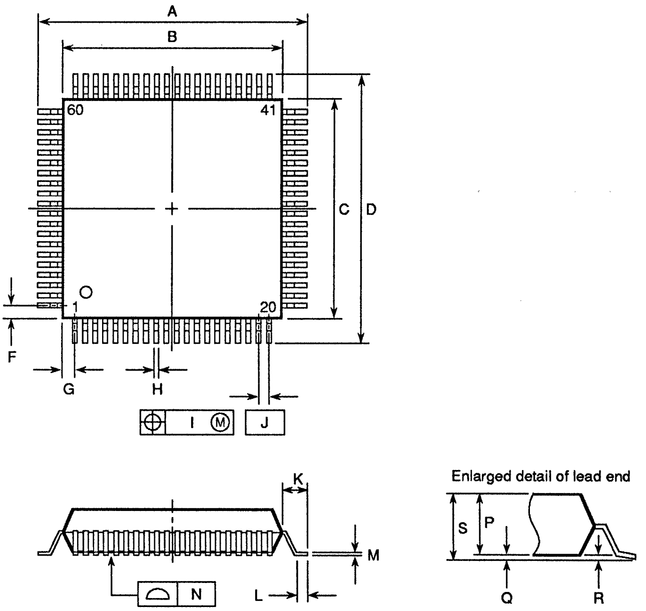


X80KW-80A

49NR-617B (11/89)

80-Pin Plastic QFP (14 by 14 mm)

Item	Millimeters	Inches
A	17.2 ±0.4	.677 ±.016
B	14.0 ±0.2	$.551^{+.009}_{-.008}$
C	14.0 ±0.2	$.551^{+.009}_{-.008}$
D	17.2 ±0.4	.677 ±.016
F	0.8	.031
G	0.8	.031
H	0.30 ±0.10	$.012^{+.004}_{-.005}$
I	0.13	.005
J	0.65 (TP)	.026 (TP)
K	1.6 ±0.2	.063 ±.008
L	0.8 ±0.2	$.031^{+.009}_{-.008}$
M	$0.15^{+0.10}_{-0.05}$	$.006^{+.004}_{-.003}$
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.119 max

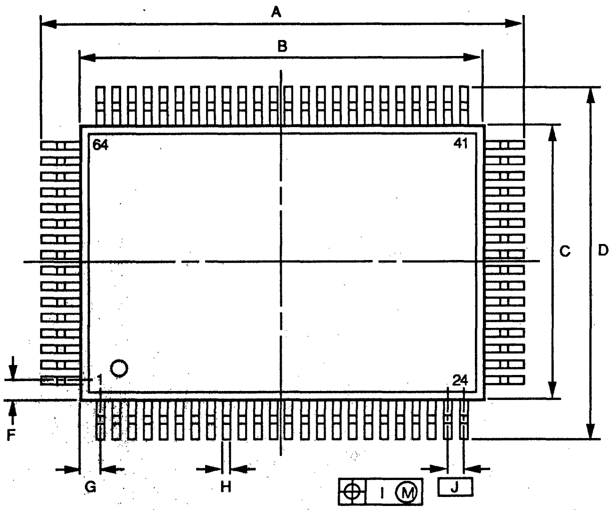


S80GC-65-3B9-1

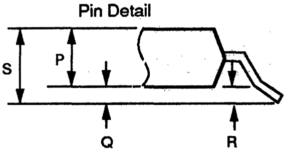
49NR-591B (2/90)

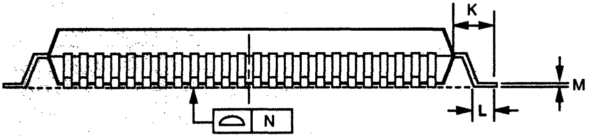
80-Pin Plastic QFP (20 by 14 mm; 1.8-mm leads)

Item	Millimeters	Inches
A	23.6 ±0.4	.929 ±.016
B	20.0 ±0.2	.787 $^{+.009}_{-.008}$
C	14.0 ±0.2	.551 $^{+.009}_{-.008}$
D	17.6 ±0.4	.693 ±.016
F	1.0	.039
G	0.8	.031
H	0.35 ±0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.8 ±0.2	.071 $^{+.009}_{-.008}$
L	0.8 ±0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.002}$
N	0.15	.006
P	2.7	.106
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	3.0 max	.118 max



Pin Detail



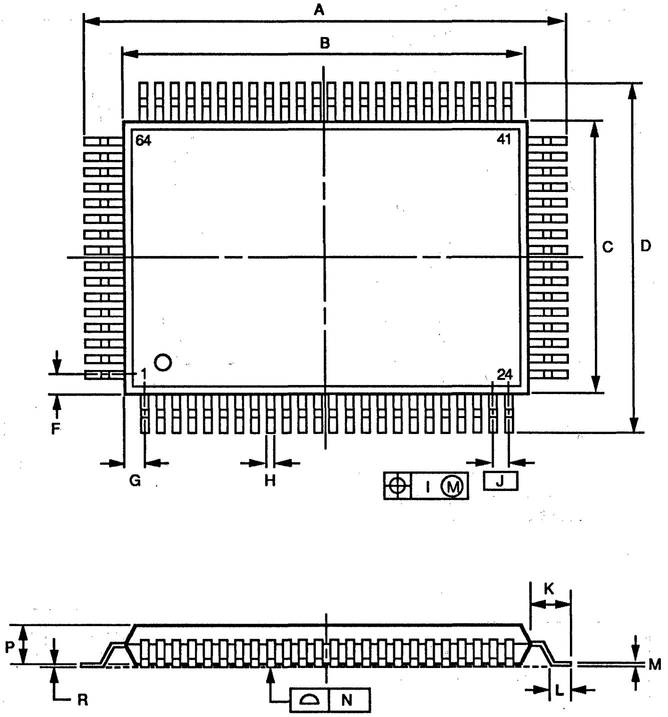


P80GF-80-3B9-1

83IH-5543B (2/90)

80-Pin Plastic QFP (20 by 14 mm; 2.35-mm leads)

Item	Millimeters	Inches
A	24.7 ±0.4	.972 ±.016
B	20.0 ±0.2	.787 $^{+.009}_{-.008}$
C	14.0 ±0.2	.551 $^{+.009}_{-.008}$
D	18.7 ±0.4	.736 ±.016
F	1.0	.039
G	0.8	.031
H	0.35 ±0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	2.35 ±0.2	.093 $^{+.009}_{-.008}$
L	1.2 ±0.2	.047 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.002}$
N	0.15	.006
P	2.05 $^{+0.2}_{-0.1}$	.081 $^{+.008}_{-.004}$
R	0.1 ±0.1	.004 ±.004

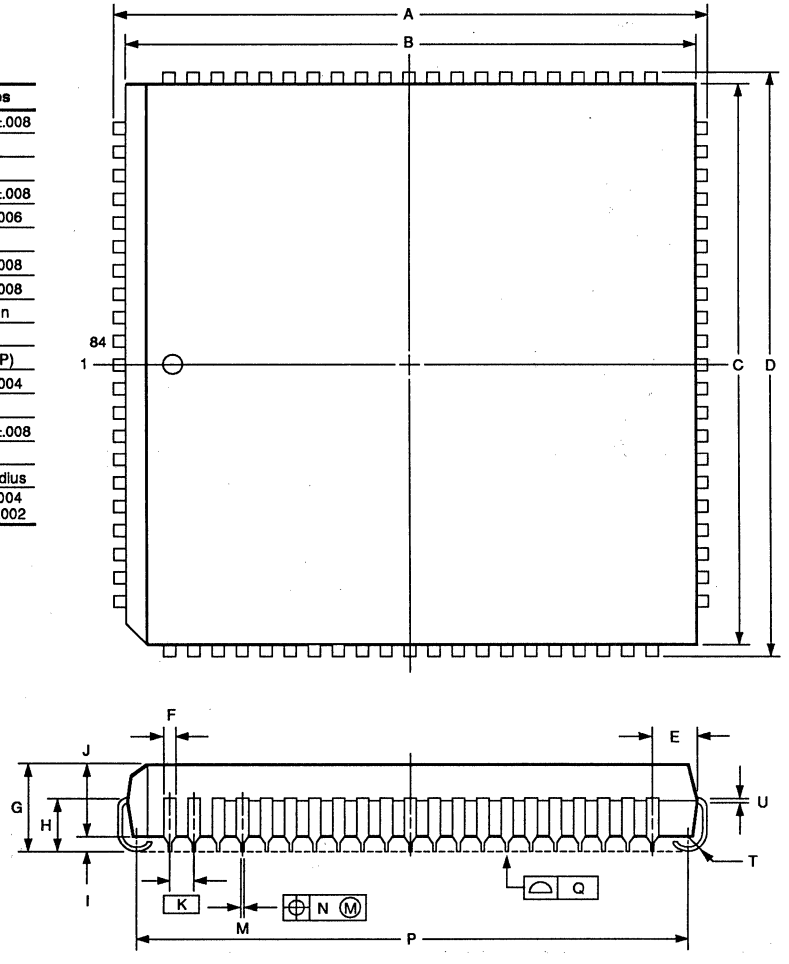


P80G-80-12

83SL-6222B (6/89)

84-Pin PLCC

Item	Millimeters	Inches
A	30.2 ±0.2	1.189 ±.008
B	29.28	1.153
C	29.28	1.153
D	30.2 ±0.2	1.189 ±.008
E	1.94 ±0.15	.076 ±.006
F	0.6	.024
G	4.4 ±0.2	.173 ±.008
H	2.8 ±0.2	.110 ±.008
I	0.9 min	.035 min
J	3.4	.134
K	1.27 (TP)	.050 (TP)
M	0.40 ±0.10	.016 ±.004
N	0.12	.005
P	28.20 ±0.20	1.110 ±.008
Q	0.15	.006
T	0.8 radius	.031 radius
U	$0.20^{+0.10}_{-0.05}$	$.008^{+.004}_{-.002}$

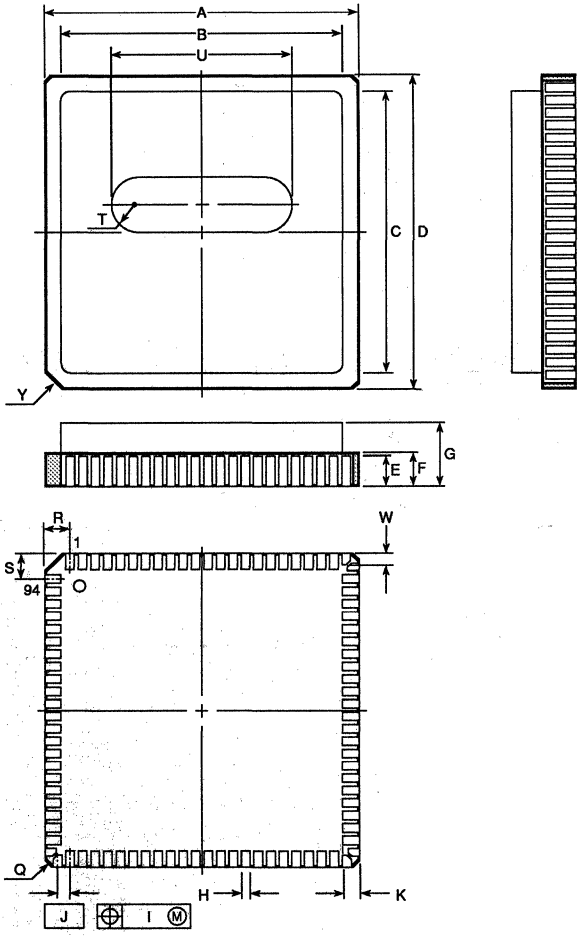


P84L-50A3-1

(2/90)
83YL-5806B

94-Pin Ceramic LCC (w/window)

Item	Millimeters	Inches
A	20.0 ±0.4	.787 ±.017
B	18.0	.709
C	18.0	.709
D	20.0 ±0.4	.787 ±.017
E	1.94	.076
F	2.14	.084
G	4.064 max	.160 max
H	0.51 ±0.10	.020 ±.004
I	0.08	.003
J	0.8 (TP)	.031 (TP)
K	1.0 ±0.2	.039 ±.008
Q	0.3 cor	.012 cor
R	1.6	.063
S	1.6	.063
T	1.75 rad	.069 rad
U	11.5	.453
W	0.75 ±0.2	.030 ±.008
Y	1.0 cor	.039 cor

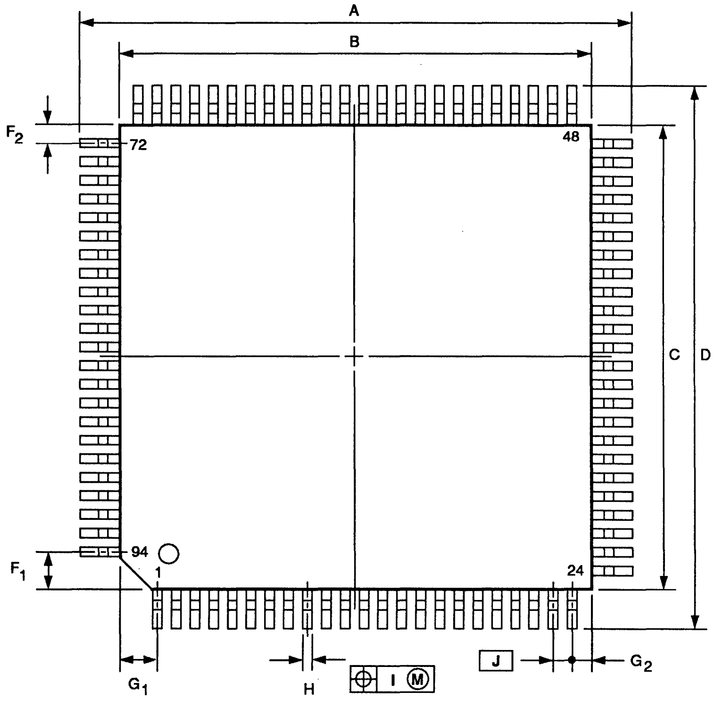


X94KW-80A

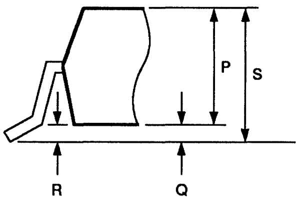
49NR-704B (3/90)

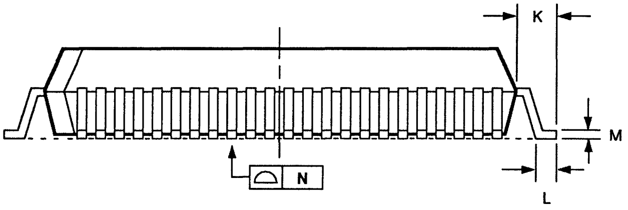
94-Pin Plastic QFP

Item	Millimeters	Inches
A	23.2 ±0.4	.913 $^{+.017}_{-.016}$
B	20.0 ±0.2	.787 $^{+.009}_{-.008}$
C	20.0 ±0.2	.787 $^{+.009}_{-.008}$
D	23.2 ±0.4	.913 $^{+.017}_{-.016}$
F_1	1.6	.063
F_2	0.8	.031
G_1	1.6	.063
G_2	0.8	.031
H	0.35 ±0.10	.014 $^{+.004}_{-.005}$
I	0.15	.006
J	0.8 (TP)	.031 (TP)
K	1.6 ±0.2	.063 ±.008
L	0.8 ±0.2	.031 $^{+.009}_{-.008}$
M	0.15 $^{+0.10}_{-0.05}$	.006 $^{+.004}_{-.003}$
N	0.15	.006
P	3.7	.146
Q	0.1 ±0.1	.004 ±.004
R	0.1 ±0.1	.004 ±.004
S	4.0 max	.158 max



Detail of lead end





S94GJ-80-5BG-1

(2/90)
83YL-5810B

NEC

NEC Electronics Inc.

CORPORATE HEADQUARTERS

401 Ellis Street
P.O. Box 7241
Mountain View, CA 94039
TEL 415-960-6000
TLX 3715792

For literature, call toll-free 8 a.m. to 4 p.m. Pacific time:

1-800-632-3531

50053