PanaXSeries

The One to Watch for Constant Innovation-Making the Future Come Alive

MICROCOMPUTER MN1870

MN1872423/3223/4023/4823

LSI User's Manual

Pub.No.21223-010E

Panasonic

Request for your special attention and precautions in using the technical information and semiconductors described in this book

(1) An export permit needs to be obtained from the competent authorities of the Japanese Government if any of the products or technologies described in this book and controlled under the "Foreign Exchange and Foreign Trade Law" is to be exported or taken out of Japan.

(2) The contents of this book are subject to change without notice in matters of improved function. When finalizing your design,therefore,ask for the most up-to-date version in advance in order to check for any changes.

(3) We are not liable for any damage arising out of the use of the contents of this book, or for any infringement of patents or any other rights owned by a third party.

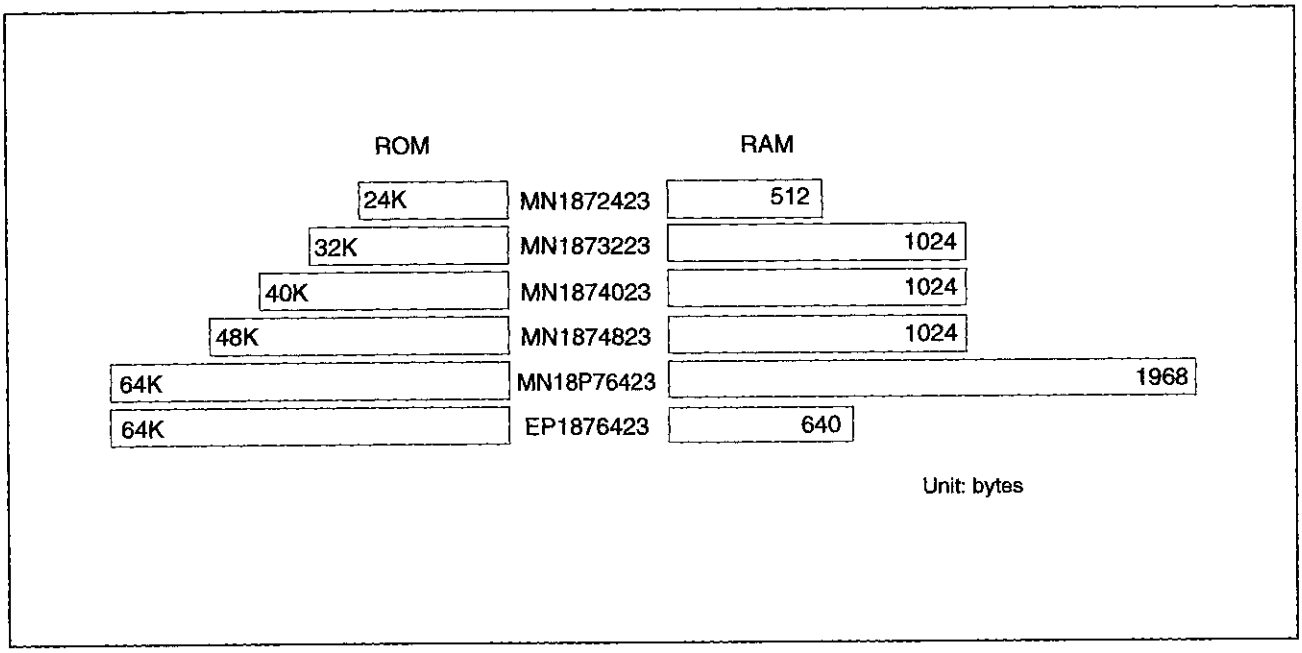
(4) No part of this book may be reprinted or reproduced by any means without written permission from our company.

(5) This book deals with standard specifications. Ask for the latest individual Product Standards or Specifications in advance for more detailed information required for your design,purchasing and applications.

If you have any inquiries or questions about this book or our semiconductors, please contact one of our sales offices listed at the back of this book or Matsushita Electronics Corporation's Sales Department.

How to Read this Manual

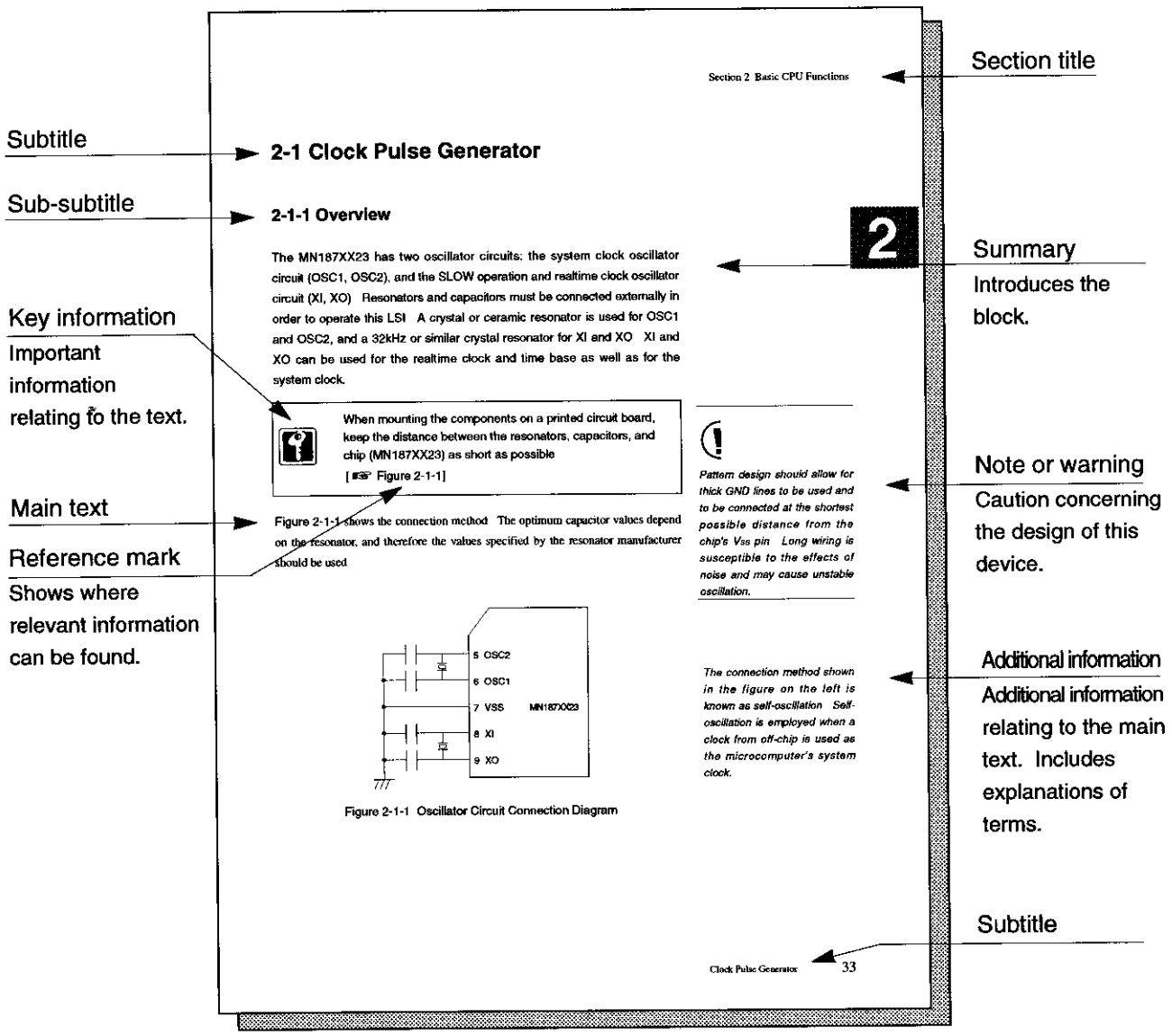
MN187XX23 microcomputers are available in a number of ROM/RAM versions to suit a variety of applications.
In addition to mask versions, EPROM and piggyback versions are also available which can programmed by the user. In this manual the general term "MN187XX23" is used to refer to these various products.



This manual gives an overview of the MN1874823's functions, followed by a description of the basic CPU functions, and port, timer, serial, display and other peripheral functions.

■ Format of this Manual

This manual consists of titles, summaries, and notes and warnings. The layout and meaning of each part is shown below.



■ Finding Information

There are five ways of finding the information you need quickly.

(1) To find the beginning of each chapter, refer to the index at the front of the manual.
(2) To look for a particular title, refer to the Table of Contents at the front of the manual.
(3) To find information from an illustration or table, refer to the Lists of Figures and Tables at the front of the manual.
(4) The chapter title and subtitle are shown at the top and bottom of each page, respectively. A general idea of the contents can therefore be obtained quickly by leafing through the manual.

■ Related Manuals

This manual has been written for users wishing to design circuits using the MN187XX23. The following related manuals are also available.

MN1870 Series Instruction Manual
(Description of instructions)
MN1870 Series Cross Assembler User's Manual
(Assembler syntax and notation)
MN1870 Series CL/1 User's Manual—Operating Volume
(CL/1 compiler installation, start-up, and options)
MN1870 Series CL/1 User's Manual—Language Volume
(High-level language syntax)
MN1870 Series CL/1 Source Code Debugger User's Manual
(Use of CL/1 source code debugger)
MN1870 Series In-Circuit Emulator Installation Manual
(In-circuit emulator installation)

■ Note

If you have any queries or comments concerning the contents of this manual, please contact your nearest Semiconductor Design Center (a list is given at the end of the manual).

Table of Contents/Lists of Figures and Tables

Table of Contents

0

Chapter 1 Overview

Chapter 2 Basic CPU Functions

Chapter 3 I/O Port Functions

Chapter 4 Timer Functions

Chapter 5 Serial Functions

Chapter 6 Display Functions

Lists of Figures and Tables

List of Figures

Chapter 1 Overview

Chapter 2 Basic CPU Functions

Chapter 3 I/O Port Functions

Chapter 4 Timer Functions

Chapter 5 Serial Functions

Chapter 6 Display Functions

Chapter 7 PWM Output

Chapter 8 A/D Conversion

Chapter 9 Noise Cancellation

Chapter 10 Remote Control Transmission

Chapter 11 Automatic Data Transfer

Chapter 12 Appendix

List of Tables

Chapter 1 Overview

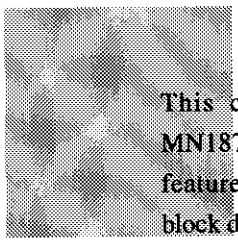
Chapter 2 Basic CPU Functions

Chapter 3 I/O Port Functions

Chapter 4 Timer Functions

Chapter 5 Serial Functions

Chapter 1 Overview



This chapter summarizes the functions of the MN187XX23, and includes descriptions of the main features, pin configuration and functions, function block diagrams, and MN1870 Series instruction system.

1-1 Product Overview

1

1-1-1 Overview

The MN1870 Series comprises 8-bit single-chip microcomputers that incorporate a variety of peripheral functions. These microcomputers are mainly used in such products as VCRs, DCCs, TVs, CD and LD players, printers, telephones, HA equipment, pagers, air conditioners, PPCs, remote controllers, fax machines, and musical instruments.

Panasonic's microcomputer family is made up of this 8-bit series plus the 4-bit MN1700 Series and MN1500 Series used in washing machines, rice cookers, cameras, and so on, and the 16-bit MN10200 Series, etc., used in high-level business and communications equipment, measuring instruments, robots, and so forth. The appropriate series can thus be selected to suit the particular application.

The MN1870 Series comprises 8-bit microcomputers which are ideal for use in a range of applications, from increasingly high-tech consumer products to industrial equipment.

This manual describes the MN187XX23, one of the products in the MN1870 Series. The MN187XX23 features 24 to 64-Kbyte ROM and 512 to 1024-Kbyte RAM, plus on-chip peripherals including six timers, three serial interfaces, a fluorescent light panel drive function (FLP), PWM output function, analog input, remote control noise filter, and remote control transmission functions, providing an ideal system configuration for use as the panel controller microcomputer for VCR tuning timers, CD players, DAT players, and so on.

Two oscillator circuits (4.19MHz or 8.38MHz/32kHz) are built in, allowing the system clock to be switched between high and low speed. An automatic data transfer (ADT) function activated by an interrupt source provides efficient data transfer processing.

The machine cycle is 0.95µs with (source oscillation fosc=4.19MHz) or 0.477µs (source oscillation fosc=8.38MHz). The package is an 84-pin QFP.

1-1-2 Features

The main features of the MN187XX23 are listed below.

- Fluorescent light panel drive function (FLP)
- Remote control transmission/reception
- PWM output
- 8-bit A/D converter (8 channels)
- High speed: 0.95µs/0.477µs execution speed (at 4.19MHz/8.38MHz)

1-1-3 Product List

In this manual, the generic name "MN187XX23" covers the models listed below. All these models have the same functions.

Table 1-1-1 Product List

Model	ROM Size	RAM Size	Package	Type
MN1872423	24 Kbytes	512 bytes	84QFP	Mask ROM versions
MN1873223	32 Kbytes	1024 bytes		
MN1874023	40 Kbytes	1024 bytes		
MN1874823	48 Kbytes	1024 bytes		
MN18P76423	64 Kbytes	1968 bytes		EPROM version
EP1876423	64 Kbytes	640 bytes*		Piggyback version

* *The EP1876423 (piggyback version) is only available for the MN1872423.*

1-2 Hardware and Software Functions

1-2-1 Hardware Functions

ROM size	24,576 to 65,536 bytes*	
RAM size	512 to 1,024 bytes* (data area and stack area)	
Machine cycle	Switchable by software: 0.95µs/0.477µs (at 4.19MHz/ 8.38MHz) or 125µs (at 32kHz)	
External interrupts	Interrupt levels:	4
	(Including 2 with noise filter)	
Internal interrupts	Interrupt levels:	11
	6 timer interrupt levels	
	3 serial interrupt levels	
	1 KYS (key-scan) interrupt level	
	1 ADT (automatic data transfer) interrupt level	
Timer counters	8-bit programmable timers:	5
	Pulse width measurement function	
	Event count function	
	Timer output function	
	Time-base counter:	1
	Watchdog timer:	1
Serial interfaces	8-bit synchronous:	3
	MN1500 Series upward-compatible	
	Transmission/reception with any bit length from 1 to 8	
	Software programmable start condition function allowing	
	MSB-first or LSB-first transfer and clock polarity switching	
Fluorescent light panel drive function	Max. 16 segments × 16 digits Can also be used for key-scanning 15-step dimmer	

☞* Depends on the model. For details, see section "1-1-3, Product List."

PWM output function	14-bit resolution: (7-bit basic cycle + 7-bit additional pulse) 15.6ms (2^{14}/fs) repeat cycle, 122µs (2^7/fs) basic cycle	1
	8-bit resolution: 244µs repeat cycle	2
Analog input	8-bit A/D converter:	8 channels
Buzzer output	2kHz/4kHz (at fosc=4 .19MHz, fs/2^9 or fs/2^8)	
Remote control receive function	Reception by noise-filtered external interrupt input	
Remote control transmit function	Choice of carrier signal/no carrier signal and synchronous output/asynchronous output	
Standby modes	HALT and STOP modes Total of 9 modes selectable with clock switching (NORMAL/IDLE/SLOW mode) combinations	
Operating voltage range	4.5 to 5.5V (at 8.38MHz) 2.2 to 5.5V (at 32kHz and in STOP mode)	
I/O pins	Memory-mapped I/O General-purpose input/output: High-voltage output: (multiplexed as FLP output) Multiplexed input/output: (multiplexed as interrupt input, serial interface, etc.)	 8 32 33
Package	84QFP (18 mm square)	
Piggyback	EP1876423*	
EPROM version	MN18P76423	
Emulator	PX-ICE1870/80 + PX-PRB1873223	
Process	Silicon gate CMOS	

* *The RAM size of the EP1876423 (piggyback version) version is 640 bytes.*
Available only for the MN1872423.

1-2-2 Software Functions

In the MN1870 Series, an instruction subset of the previous MN1880 Series has been partially enhanced, with 45 instructions and 144 possible methods of use as variations. A multi-loop function using the stack and an ADT function that can be activated by an interrupt source are also provided.
[☞ "MN1870 Series Instruction Manual"]

Major features are summarized below.

- The instruction system comprises five kinds of instructions: data transfer, logic operation, arithmetic operation, branch, and stack/hardware manipulation.
- There are three addressing modes: direct, register indirect, and immediate.
- There are three operand formats: single-operand (an operation is performed on the operand value, and the result is stored as the operand value); double-operand (an operation is performed on the source and destination operand values, and the result is stored as the destination operand value): and triple-operand (an operation is performed on the first source and second source operand values, and the result is stored as the destination operand value).

The instruction functions are shown in table 1-2-1.

Table 1-2-1 MN1870 Series Instruction Functions

Instruction Type	Description
Data transfer	Unidirectional move instructions (5) MOV, MOVL, RDTBL, MOV1, MOV1N Bidirectional move instructions (2) XCH, XCH4 Fixed-value move instructions (2) CLR, SET
Logic operation	Logic operation instructions (6) NOT, ROL, ROR, AND, OR, XOR
Arithmetic operation	Arithmetic operation instructions (12) ADDC, ADDD, ADDR, INC, DEC, SUBC, SUBD, MUL, DIV, CMP, CMPL, CMPM
Branch	Branch instructions (17) BR, SKIP, BC, BZ, BLE, BNC, BNZ, BGT, T1BNZ, T1BZ, LOOP, RLOOP, CMPBNE, CMPBE, CALL, RET, RETI
Stack/hardware manipulation	Stack/hardware manipulation instructions (5) PUSH, POP, NOP, STOP, HALT

1-2-3 Instruction Compatibility

The MN1870 Series has a partially enhanced subset of the previous MN1880 Series. The differences are explained here by showing the functions that have been added or deleted.

Instruction functions added in the MN1870 Series are shown in table 1-2-2.

Table 1-2-2 MN1870 Series Instruction Functions Not Provided in MN1880 Series

Instruction Type	Instruction Function
Data transfer	CLR (XP) bp SET (XP) bp
Arithmetic operation	INCL (da) DECL (da)
Branch	T1BZ (XP) bp, $label T1BNZ (XP) bp, $label CALL XP
Stack/hardware manipulation	STOP HALT

Instruction functions deleted in the MN1870 Series are shown in table 1-2-3.

Table 1-2-3 MN1880 Series Instruction Functions Not Provided in MN1870 Series

Instruction Type	Instruction Function	
Data transfer	REP	imm4
	ASL	(XP)
	ASL	(da)
	ASR	(XP)
	ASR	(da)
	MOVDA	(dad16), (das16)
	MOV	SP, XP
	MOV	XP, SP
	MOV	YP, XP
	MOV	XP, YP
	XCH	XP, LP
Arithmetic operation	CMPL	(da1), (da2)
	SUBCL	(XP), (YP)
	SUBCL	(dad), (das)
	ADDCL	(XP), (YP)
	ADDCL	(dad), (das)
	CMP	XPl, (da)
	CMP	YPl, (da)
	ADDR	XPl, (da1), (da2)
	ADDR	YPl, (da1), (da2)
	ADDR	XPl, XPl, XPh
	ADDR	YPl, YPl, YPh
Branch	PI	
Stack/hardware manipulation	WAIT	imm16

1-3 Pins

1-3-1 Pin Configuration

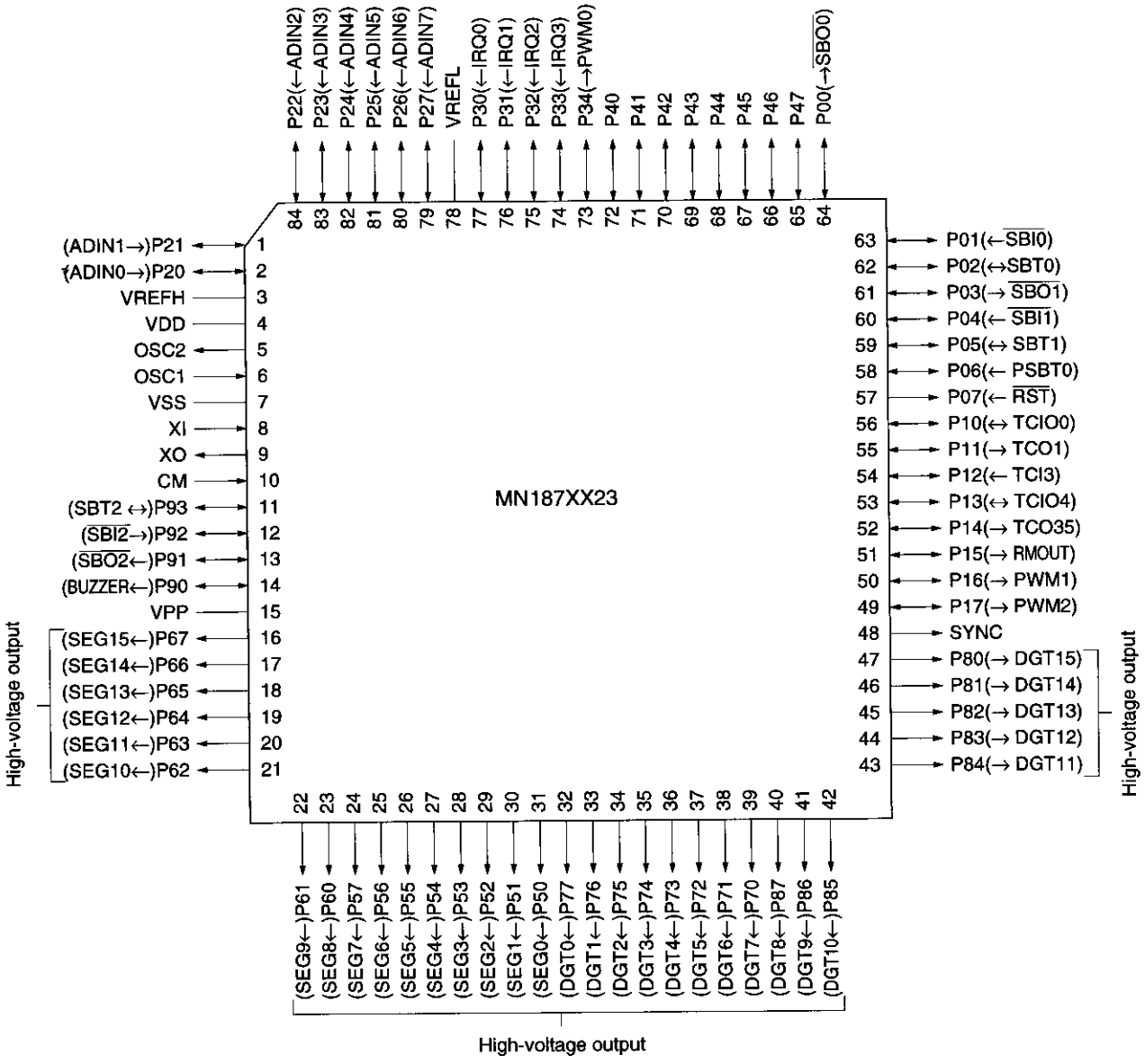


Figure 1-3-1 Pin Configuration (84QFP: Top View)

1-3-2 Pin Functions

Table 1-3-1 Pin Functions (1/5)

Pin No. 84QFP	Pin Name	I/O	Designation	Function
4 7	VDD VSS	I	Power supply pins	+5 V±10% to +3 V±10% is applied to VDD, and 0 V to VSS.
15	VPP	I	Pull-down voltage input for high-voltage output	The voltage supplied to the pull-down resistors for ports 5 to 8 is applied to this pin.
3 78	VREFH VREFL	I	A/D converter reference power supply	The A/D converter reference power supply high and low sides are applied to these pins.
6 5	OSC1 OSC2	I O	Clock input pin Clock output pin	Oscillation pins for connection of fosc (4.19MHz/8.38MHz) ceramic resonator or crystal resonator. When a clock is input from off-chip, input the clock to OSC1 and leave OSC2 open. Operation on an external clock is not possible when STOP mode or SLOW mode is used.
8 9	XI XO	I O	Clock input pin Clock output pin	Oscillation pins for connection of fx (32kHz) ceramic resonator or crystal resonator. When a clock is input from off-chip, input the clock to XI and leave the XO pin open. Operation on an external clock is not possible when STOP mode is used.
48	SYNC	O	Timing output pin	Outputs an $fs/2^{13}$ or $fx/2^{8}$ clock. Can be used as the standard clock for realtime clock regulation.
10	CM	I	Chip mode select pin	Connect to VSS.
57	$\overline{RST}$	I/O	Reset pin	Pin used to reset the chip at power-on, It is assigned to port P07 and incorporates a pull-up resistor (Typ. 50kΩ). When this pin is forced low, the chip's internal state is initialized. When the input is forced high again, the reset is released and reset interrupt handling is executed by hardware after the elapse of the system clock oscillation stabilization time. When 0 is written to port P07 and the chip is reset by software, or when the watchdog timer overflows, a low level is output. The output is of N-channel open-drain type, and when a capacitor is inserted between $\overline{RST}$ and VDD, connection of a discharge diode between $\overline{RST}$ and VDD is recommended.

Table 1-3-1 Pin Functions (2/5)

Pin No. 84QFP	Pin Name	I/O	Designation	Function
64~57	P00~P07	I/O	Standard input/output port 0	8-bit input/output port. Ports P00 to P06 are CMOS 3-state input/output ports, and P07 is an N-channel open-drain input/output port (shared with $\overline{RST}$). Ports P00 to P06 can be set as input or output bit-wise by the port 0 direction control register (P0DIR). On reset, these pins are set to input mode (output high-impedance). Use of a pull-up resistor can be selected for ports P00, P01, P03, P04, and P06 with the P0RON register. See the relevant sections for special functions. [☞ Table 1-3-1, Pin Functions: $\overline{SBO0}$, $\overline{SBI0}$, SBT0, $\overline{SBO1}$, $\overline{SBI1}$, SBT1, PSBT0, $\overline{RST}$ entries]
56~49	P10~P17	I/O	Standard input/output port 1	8-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 1 direction control register (P1DIR). On reset, these pins are set to input mode (output high-impedance). Use of a pull-up resistor can be selected for all pins with the P1RON register. See the relevant sections for special functions. [☞ Table 1-3-1, Pin Functions: TCIO0, TCO1, TCI3, TCIO4, TCO35, RMOUT, PWM1, and PWM2 entries]
2,1, 84~79	P20~P27	I/O	Standard input/output port 2	8-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 2 direction control register (P2DIR). On reset, these pins are set to input mode (output high-impedance). Use of a pull-up resistor can be selected for all pins with the P2HRON and P2LRON registers (in nibble units). See the relevant sections for special functions. [☞ Table 1-3-1, Pin Functions: ADIN0~ADIN7 entry]
77~73	P30~P34	I/O	Standard input/output port 3	5-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 3 direction control register (P3DIR). On reset, these pins are set to input mode (output high-impedance). Use of a pull-up resistor can be selected for ports P30 to P33 with the P3RON register. See the relevant sections for special functions. [☞ Table 1-3-1, Pin Functions: IRQ0~IRQ3 entry]

Table 1-3-1 Pin Functions (3/5)

Pin No. 84QFP	Pin Name	I/O	Designation	Function
72~65	P40~P47	I/O	Standard input/output port 4	8-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 4 direction control register (P4DIR). On reset, these pins are set to input mode (output high-impedance). Use of a pull-down resistor can be selected for all pins with the P4RON register.
31~24	P50~P57	O	High-break down voltage output port 5	8-bit P-channel open-drain output port. Port 5 pins incorporate a pull-down resistor between the pin and VPP. Byte-wise switching between port and segment (SEG0~SEG7) functions is performed by port/FLP select register 2 (PFSR2). On reset, these pins are set to low-level output.
23~16	P60~P67	O	High-break down voltage output port 6	8-bit P-channel open-drain output port. Port 6 pins incorporate a pull-down resistor between the pin and VPP that can be removed as a mask option. Bit-wise switching between port and segment (SEG8~SEG15) functions is performed by port/FLP select register 0 (PFSR0). On reset, if a pull-down resistor is incorporated these pins are set to low-level output.
39~32	P70~P77	O	High-break down voltage output port 7	8-bit P-channel open-drain output port. Port 7 pins incorporate a pull-down resistor between the pin and VPP. Byte-wise switching between port and digit (DGT0~DGT7) functions is performed by port/FLP select register 2 (PFSR2). On reset, if a pull-down resistor is incorporated these pins are set to low-level output.
47~40	P80~P87	O	High-break down voltage output port 8	8-bit P-channel open-drain output port. Port 8 pins incorporate a pull-down resistor between the pin and VPP that can be removed as a mask option. Byte-wise switching between port and digit (DGT8~DGT15) functions is performed by port/FLP select register 1 (PFSR1). On reset, if a pull-down resistor is incorporated these pins are set to low-level output.

Table 1-3-1 Pin Functions (4/5)

Pin No. 84QFP	Pin Name	I/O	Designation	Function
14~11	P90~P93	I/O	Standard input/output port 9	4-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 9 direction control register (P9DIR). On reset, these pins are set to input mode (output high-impedance). Use of a pull-up resistor can be selected for ports P91 and P92 with the P9RON register. See the relevant sections for special functions. [☞ Table 1-3-1, Pin Functions: BUZZER, $\overline{SBO2}$, $\overline{SBI2}$, and SBT2 entries]
64 61 13	$\overline{SBO0}$ $\overline{SBO1}$ $\overline{SBO2}$	O	Serial interface transmit data output pins	Serial interface transmit data output pins. The output structure is CMOS push-pull or N-channel open-drain type, selectable by software. $\overline{SBO0}$ is assigned to port P00, $\overline{SBO1}$ to port P03, and $\overline{SBO2}$ to port P91. When the serial interface is not used, these pins can be used as ordinary input/output pins.
63 60 12	$\overline{SBI0}$ $\overline{SBI1}$ $\overline{SBI2}$	I	Serial interface receive data input pins	Serial interface receive data input pins. $\overline{SBI0}$ is assigned to port P01, $\overline{SBI1}$ to port P04, and $\overline{SBI2}$ to port P92. When not used as receive pins, these pins can be used as ordinary input/output pins.
62 59 11	SBT0 SBT1 SBT2	I/O	Serial interface clock input/output pins	Serial interface clock input/output pins. The SBT0, SBT1, and SBT2 output structure is CMOS push-pull or N-channel open-drain type, selectable by software. SBT0 is assigned to port P02, SBT1 to port P05, and SBT2 to port P93. When the serial interface is not used, these pins can be used as ordinary input/output pins.
58	PSBT0	I	Serial interface clock input pin	Serial interface clock input pin. PSBT0 is assigned to port P06. When the serial interface is not used, PSBT0 can be used as an ordinary input/output pin.
56 53	TCIO0 TCIO4	I/O	Timer input/output pins	Timer 0 and timer 4 clock input/output pins. The output structure is CMOS 3-state type. TCIO0 is assigned to port P10, and TCIO4 to port P13. When not used as timer input/output pins, these pins can be used as ordinary input/output pins.
54	TCI3	I	Timer input pin	Timer 3 clock input pin.. TCI3 is assigned to port P12. When not used as a timer input pin, TCI3 can be used as an ordinary input/output pin.

Table 1-3-1 Pin Functions (5/5)

Pin No. 84QFP	Pin Name	I/O	Designation	Function
55 52	TCO1 TCO35	O	Timer output pins	TCO1 is the timer 1 clock output pin, and TCO35 the timer 3 or timer 5 clock output pin. The output structure is CMOS 3-state type. TCO1 is assigned to port P11, and TCO35 to port P14. When not used as timer output pins, these pins can be used as ordinary input/output pins.
14	BUZZER	O	Buzzer output pin	A CMOS 3-state output pin that outputs an fs/2^8 or fs/2^9 buzzer waveform. BUZZER is assigned to port P90. When not used as the buzzer output pin, BUZZER can be used as an ordinary input/output pin.
73 50 49	PWM0 PWM1 PWM2	O	PWM0 output pin PWM1 output pin PWM2 output pin	PWM0 is the 14-bit PWM CMOS 3-state output pin, and PWM1 and PWM2 are 8-bit PWM CMOS 3-state output pins. PWM0 is assigned to port P34, PWM1 to port P16, and PWM2 to port P17. When not used as PWM output pins, these pins can be used as ordinary input/output pins.
77 76 75 74	IRQ0 IRQ1 IRQ2 IRQ3	I	External interrupt input pins	Input pins that accept external interrupts on a negative edge or positive edge. The valid edge can be specified in the interrupt mode control register (IRQM). IRQ1 and IRQ2 incorporate a noise cancellation circuit for remote control reception. IRQ0, IRQ1, IRQ2, and IRQ3 are assigned to ports P34, P31, P32, and P33. When interrupts are disabled, these pins can be used as ordinary input/output pins.
2, 1, 84~79	ADIN0~ ADIN7	I	Analog output pins	A/D converter analog input pins. These pins are assigned to ports P20~P27. When the A/D converter is not used, they can be used as ordinary input/output pins.
51	RMOUT	O	Remote control transmission output pin	Remote control transmission output pin. The output structure is CMOS 3-state type. RMOUT is assigned to port P15. When remote control output is disabled, RMOUT can be used as an ordinary input/output pin.

1-4 Overview of Block Functions

1-4-1 Block Diagram

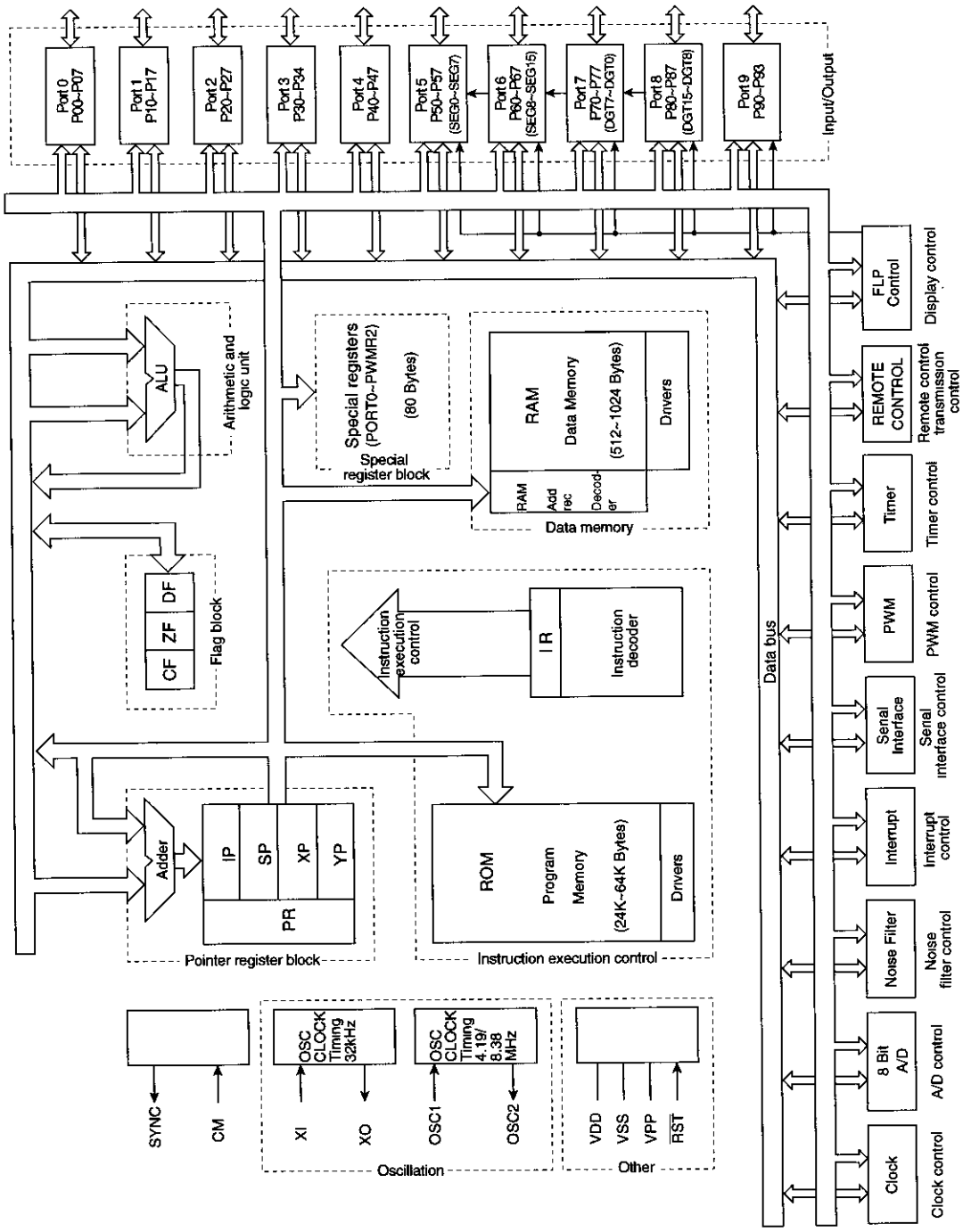


Figure 1-4-1 Block Function Diagram

1-4-2 Block Functions

Table 1-4-1 Block Functions (1/3)

Block		Function	Reference (Detailed Description)
Instruction execution control block	Instruction register IR	The CPU reads the next instruction to be executed from ROM and latches it in the instruction register (IR).	(2-2-3)
	Instruction decoder ID	The instruction decoder (ID) decodes the contents of the CPU's instruction register when in the instruction execution state, generates in sequence the control signals required to execute that instruction, and executes the instruction by controlling the various blocks in the chip.	(2-2-3)
	ROM	Read-only memory is program memory that stores the program to be executed.	(2-2-2)
Pointer register block	Instruction pointer IP	The instruction pointer (IP) is a 16-bit register that controls the order of execution of instructions in the program memory. In combination with the adder, it performs functions equivalent to those of a program counter.	(2-3-2)
	Stack pointer SP	The stack pointer is a 16-bit register that points to the address of the stack area that uses a part of the data RAM. The stack area is used to save the IP, etc., in the event of a subroutine call or interrupt, and can also be used in constructing multiple loops.	(2-3-3)
	Data pointer XP	XP is a 16-bit register used to address RAM space in register indirect addressing mode. XP is used by single-operand and double-operand instructions.	(2-3-4)
	Data pointer YP	YP is a 16-bit register used to address RAM space in register indirect addressing mode (the RDTBL instruction, only, addresses ROM space). YP is used as the source operand in a double-operand instruction.	(2-3-4)
	Adder	The adder is used for pointer register incrementing and address computation (calculation of the IP value) in a relative address branch.	———
Arithmetic and logic block	Arithmetic and logic unit ALU	The arithmetic and logic unit (ALU) performs an arithmetic operation (addition/subtraction, multiplication/division, decimal addition, increment/decrement, or compare) or a logic operation (AND, OR, XOR, complement, rotate, or nibble swap) on data input from the data bus to two 8-bit latches, and outputs the result to the data bus via an output latch.	(2-4)

Table 1-4-1 Block Functions (2/3)

Block		Function	Reference (Detailed Description)
Flag block	Flag status byte FS	The flag status byte (FS) consists of 3 flags that show the CPU's execution status.	(2-5)
	Carry flag CF	The carry flag (CF) is set when the result of an ALU operation overflows or underflows, and reset otherwise.	(2-5-2)
	Zero flag ZF	The zero flag (ZF) is set when the result of an ALU operation is 0, and reset otherwise.	(2-5-3)
	Direct flag DF	The direct flag (DF) controls direct addresses as shown below. DF = 0: The upper 8 bits of the address are processed as 0. DF = 1: The upper 8 bits of the XP or YP address are processed as the upper 8 bits of the RAM address.	(2-5-4)
Data memory block	RAM	Random access memory (RAM) is used for the stack area and as a data area for storing data needed during program execution.	(2-6)
Special register block	PORT0~ PWMR2	The special registers (PORT0~PWMR2) are allocated to RAM space, and can be read and written in the same way as RAM.	(2-7)
Clock control block	Clock	The CPU's instruction execution mode is controlled by the CPU mode register (CPUM).	(2-9)
Interrupt control block	Interrupt	Interrupts are controlled by the interrupt request flag register (IF) and the interrupt enable flag register (IE).	(2-8)
Serial interface control block	Serial Interface	Performs serial data transmission and reception. The serial interface mode register (SIM) controls the serial interface operating mode. Data transmission and reception is carried out by means of the transmit/receive shift buffer (SIBUF), and the number of transmit/receive bits is controlled by the serial interface bit counter (SBC).	(5)

Table 1-4-1 Block Functions (3/3)

Block		Function	Reference (Detailed Description)
Timer control block	Timer	This block can be used for normal timer operation, event counting, pulse width measurement, timer output, and operation as a watchdog timer. The timer mode register (TM) controls the timer operating mode and clock selection. The timer counters are used to count clock pulses, and have an 8-bit configuration.	(4)
Noise filter control block	Noise Filter	The remote control noise filter control register (RMC) is used to control the noise cancellation circuit for external interrupt inputs (IRQ1 and IRQ2).	(9)
Remote control transmission control block	REMOTE CONTROL	Remote control transmission is controlled by the remote control control register (RMC).	(10)
PWM control block	PWM	The PWM outputs are used for tuning, with 14-bit resolution for PWM0 and 8-bit resolution for PWM1 and PWM2.	(7)
A/D control block	8Bit A/D	The A/D control block is used for AFT signal discrimination with 8-bit resolution.	(8)
Display control block	FLP	Performs segment/digit pin setting, 15-stage dimmer control, 15 kinds of digit number setting.	(6)
Input/output block	P00~P93	P00 to P93 are input/output pins.	(3)
Oscillation block	OSC1 OSC2	The system clock resonator is connected to OSC1 and OSC2.	(2-1)
	XI XO	A 32kHz or similar resonator is connected to XI and XO. XI and XO have a realtime clock and time base function as well as their system clock function.	
Other	VSS VDD	VSS and VDD are power supply pins. +5V is applied to VDD.	
	$\overline{RST}$	$\overline{RST}$ is the reset pin. The operating state is entered when this pin goes high.	
	VPP	VPP is the FLP drive power supply pin. –30V is applied to this pin.	

1-5 Electrical Characteristics

Item \ Model Type	MN187XX23
Type of structure	MOS integrated circuit
Function	CMOS 8-bit single-chip microcomputer
ROM/RAM	

Model	ROM Size	RAM Size
MN1872423	24Kbytes	512 bytes
MN1873223	32Kbytes	1024 bytes
MN1874023	40Kbytes	1024 bytes
MN1874823	48Kbytes	1024 bytes
MN18P76423	64Kbytes	1968 bytes
EP1876423	64Kbytes	640 bytes

This LSI Manual gives the standard specifications. When using an LSI, please ask our sales staff for product specifications.

1-5-1 Absolute Maximum Ratings

	Item	Symbol	Rating	Unit
1	Supply voltage	V_{DD}	–0.3~+7.0	V
2		V_{PP}	V_{DD}–45~V_{DD}+0.3	V
3	Input pin voltage	V_{I1}	V_{SS}–0.3~V_{DD}+0.3	V
4	Output pin voltage	V_{O1}	V_{SS}–0.3~V_{DD}+0.3	V
5	I/O pin voltage	V_{IO1}	V_{SS}–0.3~V_{DD}+0.3	V
6	High-voltage I/O pin voltage	V_{IO2}	–40~V_{DD}+0.3	V
7	Peak output current	I_{OH} (peak)	–10	mA
8		I_{OL} (peak)	30	mA
9	Average output current*1	I_{OH} (avg)	–5	mA
10		I_{OL} (avg)	15	mA
11	High-voltage output current	$I_{OH(DGT)}$	–30	mA
12		$I_{OL(SEG)}$	–14	mA
13	Allowable dissipation	PT	1	W
14	Ambient operating temperature	T_{opr}	–20~+70	°C
15	Storage temperature	T_{stg}	–55~+125	°C

<Note> *1 Applies to any 100ms period

1-5-2 Operating Conditions

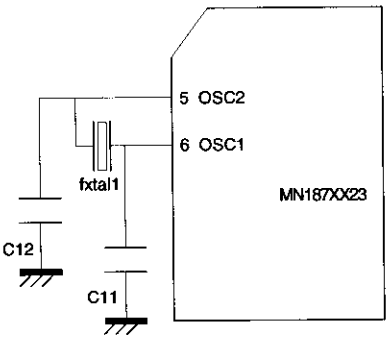
T_a=–20~+70°C, V_{DD}=5V, V_{SS}=0V

	Item	Symbol	Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
Supply voltage							
1	Supply voltage	V_{DD1}	32kHz ≤ f_{osc} ≤ 8 4MHz	4 5	5 0	5.5	V
2		V_{DD2}	32kHz ≤ f_{xi} ≤ 65kHz	2 2		5.5	
3		V_{PP}	V_{DD}–V_{PP} potential difference			35	
Operating speed[*2 *3]							
4	Instruction execution speed	t_{c1}	V_{DD}=4.3V~5.5V	0 475		125	µs
5		t_{c2}	V_{DD}=2 2V~5.5V	8 79		125	
Crystal oscillation[*3] **OSC1, OSC2**							
6	Crystal frequency	fxtal1	V_{DD}=4 3V~5.5V		4 19 8.38	8 4	MHz
7	External capacitance	C_{11}			30		pF
8		C_{12}			8		
Crystal oscillation[*4] **XI, XO**							
13	Crystal frequency	fxtal 2	V_{DD}=2.2V~5 5V	32		65	kHz
14	External capacitance	C_{21}			30		pF
15		C_{22}			22		

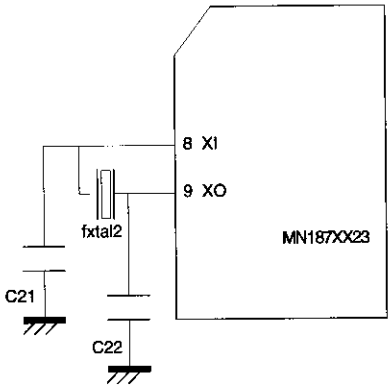
<Note>*2 t_{c1}: When OSC1 is used as the CPU clock

t_{c2}: When XI is used as the CPU clock

*3 Crystal oscillation 1
(Feedback resistor incorporated)
External capacitance is a reference value



*4 Crystal oscillation 2
(Feedback resistor incorporated)
External capacitance is a reference value



The instruction cycle is obtained by dividing the clock frequency by 4

1-5-3 DC Characteristics

(1) Supply current (no-load output)*5

V_{DD}=5.0V, Vss=0V, Ta=−20~+70°C

	Item		Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Supply current during operation	I_{DD1}	f_{osc}=8 38MHz			20	mA
2		I_{DD2}	f_{osc}=32kHz, V_{DD}=3 0V			100	μA
3	Supply current in STOP mode	I_{DD3}	Oscillation stopped, V_{DD}=3 0V			10	μA

<Note>*5 Measured at Ta = 25°C, in no-load state.

Supply current during operation I_{DD1} is measured with all I/O pins set to input mode, the oscillation mode set to NORMAL, the CM pin fixed at the Vss level and input pins at the V_{DD} level, and an 8.38MHz square wave with the V_{DD} and Vss potentials as the amplitude input from the OSC1 pin.

Supply current during operation I_{DD2} is measured with all I/O pins set to input mode, the oscillation mode set to SLOW, the CM pin fixed at the Vss level and input pins at the V_{DD} level, and a 32kHz square wave with the V_{DD} and Vss potentials as the amplitude input from the X1 pin.

The supply current in STOP mode is measured with the oscillation mode set to STOP, the CM pin fixed at the Vss level and input pins at the V_{DD} level, and the OSC1 and XI pins open.

(2) Input pin 1 CM

V_{DD}=5.0V, V_{SS}=0.0V, Ta=–20~+70°C

	Item		Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Input voltage	High-level	V_{DD}=2.7~5.5V	0 7V_{DD}		V_{DD}	V
2		Low-level	V_{DD}=2.7~5.5V	V_{SS}		0 3V_{DD}	V
3	Input leakage voltage		V_{IN}=0.0~5 0V			±10	μA

(3) I/O pin 3 P07 ($\overline{RST}$) (Schmitt input with internal pull-up resistor)

V_{DD}=5.0V, V_{SS}=0.0V, Ta=–20~+70°C

	Item		Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Input voltage	High-level	V_{DD}=4 3V~5.5V	0 75V_{DD}		V_{DD}	V
			V_{DD}=2.7V~4.3V	0 8V_{DD}		V_{DD}	V
2		Low-level	V_{DD}=2 7V~5 5V	V_{SS}		0.2V_{DD}	V
3	Input current	With pull-up resistor Without pull-down resistor	V_{IN}=1 5V		–150		μA
4	Input leakage current	Without pull-up resistor Without pull-down resistor	V_{IN}=5.0V			±10	μA
5	Output voltage	High-level					mA
6		Low-level	I_{OL}=1mA			0.5	V

(4) I/O pins 4 P00, P03, P11, P24~P27, P14~P17, P34, P40~P47, P90, P91

V_{DD}=5.0V, V_{SS}=0.0V, Ta=–20~+70°C

	Item		Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Input voltage	High-level	V_{DD}=2 7V~5 5V	0.7V_{DD}		V_{DD}	V
2		Low-level	V_{DD}=2.7V~5 5V	V_{SS}		0 3V_{DD}	V
3	Input leakage current		V_{IN}=0V~5.0V			±10	μA
4	Output voltage	High-level	I_{OH}=–500μA	4.6			V
5		Low-level	I_{OL}=2mA			0 5	V

(5) I/O pins 5 P01, P02, P04~P06, P10, P12, P13, P30~P33, P92, P93 (Schmitt input)

V_{DD}=5.0V, V_{SS}=0.0V, Ta=–20~+70°C

Item			Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Input voltage	High-level	V_{DD}=4 3~5 5V	0.75V_{DD}		V_{DD}	V
			V_{DD}=2 7~4 3V	0 8V_{DD}		V_{DD}	V
2		Low-level	V_{DD}=2.7~5.5V	V_{SS}		0.2V_{DD}	V
3	Input leakage current		V_{IN}=0V~5V			±10	µA
4	Output voltage	High-level	I_{OH}=–500µA	4 6			V
5		Low-level	I_{OL}=2mA			0.5	V

(6) High-voltage output pins 6 P70~P77, P80~P87 (DGT)

V_{DD}=5.0V, V_{SS}=0.0V, Ta=–20~+70°C

Item		Test Conditions	Allowable Values			Unit
			Min	Typ	Max	
1	Output current	V_{DD}=5V, V_{PP}=–30V V_{OH}=2 0V	–15			mA
2	Output leakage current	V_{DD}=5V, V_{PP}=–30V V_{OH}=–30V			±10	µA
3	Pull-down resistance	V_{OH}=5V, V_{PP}=–30V	30		250	kΩ

(7) High-voltage output pins 7 P50~P57, P60~P67 (SEG)

V_{DD}=5.0V, V_{SS}=0.0V, Ta=–20~+70°C

Item		Test Conditions	Allowable Values			Unit
			Min	Typ	Max	
1	Output current	V_{DD}=5V, V_{PP}=–30V V_{OH}=2 0V	–7			mA
2	Output leakage current	V_{DD}=5V, V_{PP}=–30V V_{OH}=–30V			±10	µA
3	Pull-down resistance	V_{OH}=5V, V_{PP}=–30V	30		250	kΩ

(8) Output pin 8 SYNC

V_{DD}=5.0V, V_{SS}=0.0V, T_a=–20~+70°C

	Item		Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Output voltage	High-level	I_{OH}=–250μA	4 6			V
2		Low-level	I_{OL}=1mA			0 5	V

(9) A/D converter

V_{DD}=5.0V, V_{SS}=0.0V, T_a=–20~+70°C

	Item		Test Conditions	Allowable Values			Unit
				Min	Typ	Max	
1	Resolution					8 0	bit
2	Linearity error		V_{DD}=5.0V, V_{SS}=0V,			±3	LSB
3	Differential linearity error		V_{REFH}=5.0V, V_{REFL}=0V			±3	
4	Zero transition	V_{OT}				+100	mV
5	Full-scale transition	V_{FT}		+4900			
6	Conversion time		f_{osc}=4 19MHz/8 38MHz		8 59		μs
7	Sampling time				0 954		
8	Reference voltage	V_{REFH}		V_{REFL}		V_{DD}	V
9		V_{REFL}		V_{SS}		V_{REFH}	
10	Analog input voltage	V_{ADIN}		V_{REFL}		V_{REFH}	
11	Analog input current	A_{ADIN}	V_{ADIN}=0~5V			±3	μA
12	Analog input leakage	ADILK	V_{ADIN}=0~5V			±3	
13	Ladder resistance	R_{ladd}		10	50	100	kΩ

1-5-4 Example of Use of Ceramic Resonator

A ceramic or crystal resonator can be used for the oscillation clock of the MN1873223. Figure 1-5-1 shows an example of the basic configuration when a comparatively low-cost ceramic resonator is used.

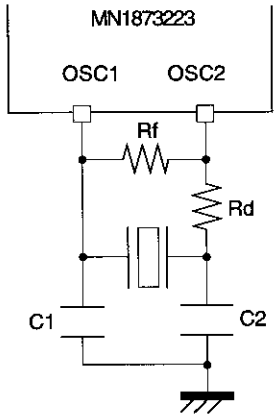


Figure 1-5-1 Example of Basic Ceramic Resonator Configuration

In the case of a crystal or ceramic resonator connected to OSC1 and OSC2 or XI/XO, the circuit constants will depend on floating capacitance, etc., in the resonator and connected circuitry, and must be decided upon in consultation with the resonator manufacturer.

—Reference—

Ceramic resonators manufactured by Matsushita Electronic Components, Ltd., are described below.

■ Example for 4.00MHz oscillation frequency

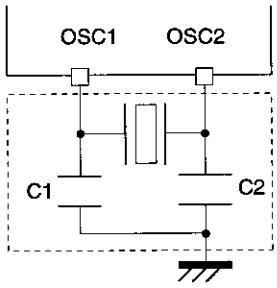
Ceramic resonator product code: EFOEC4004A [□] ([□]: allowable frequency range)

C1, C2=33pF (typ.) [incorporated in resonator]

■ Example for 8.38MHz oscillation frequency

Ceramic resonator product code: EFOEC8384A [□] ([□]: allowable frequency range)

C1, C2=33pF (typ.) [incorporated in resonator]



Resonators are also available which allow external connection of C1 and C2.

For queries concerning the above ceramic resonators and evaluation of oscillator circuit constants, please contact the Ceramics Division of Matsushita Electronic Components, Ltd. at the following numbers.

TEL 06-906-3192 (JAPAN)

FAX 06-908-7735 (JAPAN)

1-6 Mask Option Checklist

Date (MM/DD/YY)

SE No.

Model Name	

Customer	
Sigunature	

1. Oscillation Mode

Type A	Type B

<Note> Type A: Operation is started from a reset cycle in NORMAL mode
Type B: Operation is started from a reset cycle in SLOW mode.

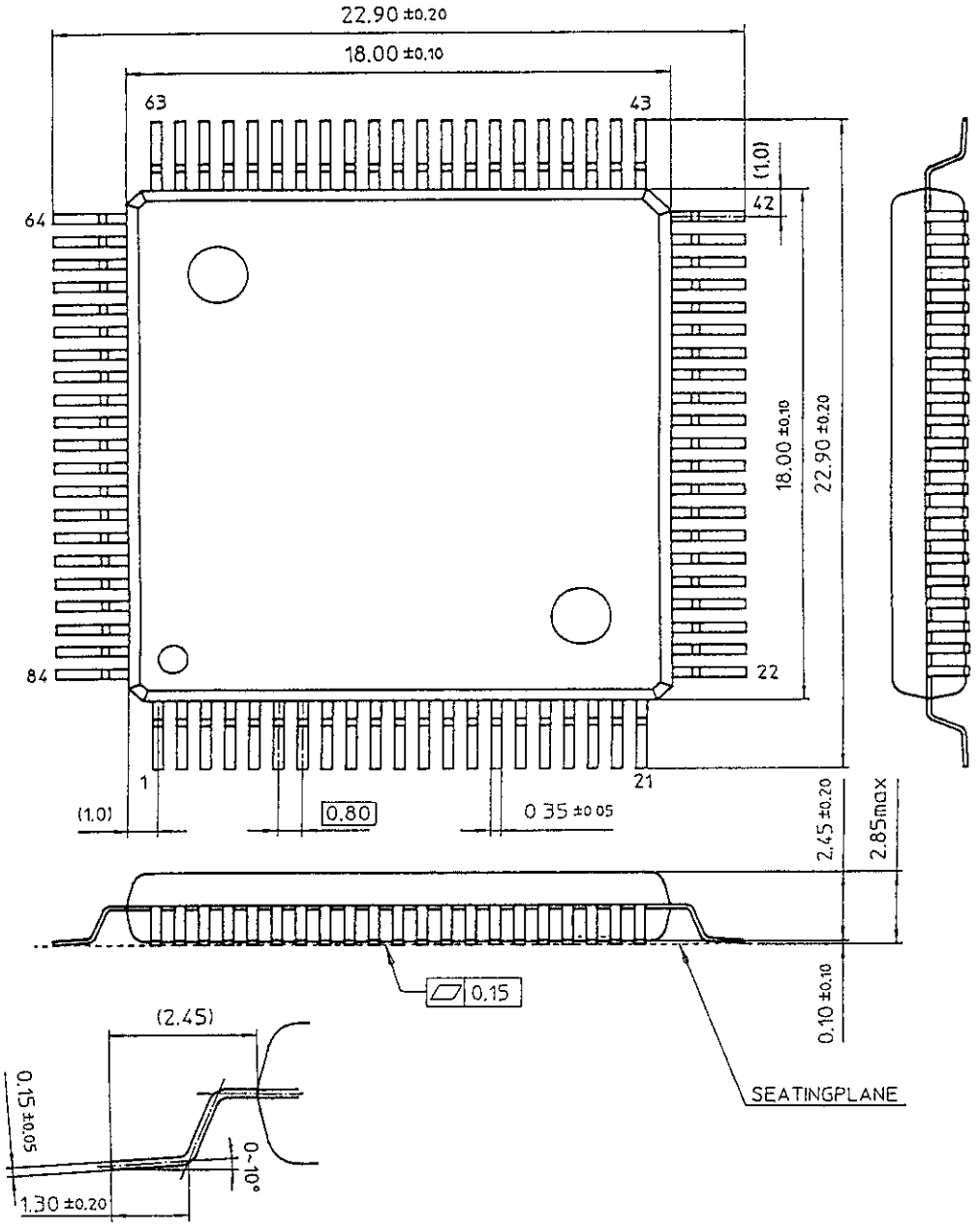
2. Pull-Down Resistor Settings

Port No.		Pull-Down Resistor	
		Yes	No
P6 Vpp pull-down	0		
	1		
	2		
	3		
	4		
	5		
	6		
	7		
P8 Vpp pull-down	0		
	1		
	2		
	3		
	4		
	5		
	6		
	7		

<Note>
Mask option contents are subject to change. When ordering a mask, please request the latest Option List from your Matsushita sales representative.

1-7 Package Dimensions

Sealant: epoxy resin, lead material: Copper, lead surface treatment: Palladium plating

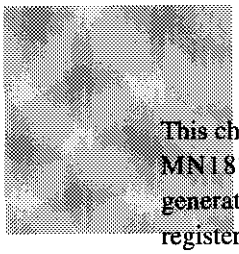


Package code name: *QFP084-P-1818E

Figure 1-7-1 84-Pin QFP (18 mm Square)

<Note>
The package dimension drawing is subject to change. Please request product specifications from our sales staff before using this product.

Chapter 2 Basic CPU Functions 2



This chapter describes the basic functional blocks of the MN1870 Series CPU, covering the clock pulse generator, instruction execution control block, pointer registers, flags, special registers, and interrupts.

2-1 Clock Pulse Generator

2-1-1 Overview

The MN187XX23 has two oscillator circuits: the system clock oscillator circuit (OSC1, OSC2), and the SLOW operation and realtime clock oscillator circuit (XI, XO). Resonators and capacitors must be connected externally in order to operate this LSI. A crystal or ceramic resonator is used for OSC1 and OSC2, and a 32kHz or similar crystal resonator for XI and XO. XI and XO can be used for the realtime clock and time base as well as for the system clock.

When mounting the components on a printed circuit board, keep the distance between the resonators, capacitors, and chip (MN187XX23) as short as possible.
[☞ Figure 2-1-1]

Pattern design should allow for thick GND lines to be used and to be connected at the shortest possible distance from the chip's Vss pin. Long wiring is susceptible to the effects of noise and may cause unstable oscillation.

Figure 2-1-1 shows the connection method. The optimum capacitor values depend on the resonator, and therefore the values specified by the resonator manufacturer should be used.

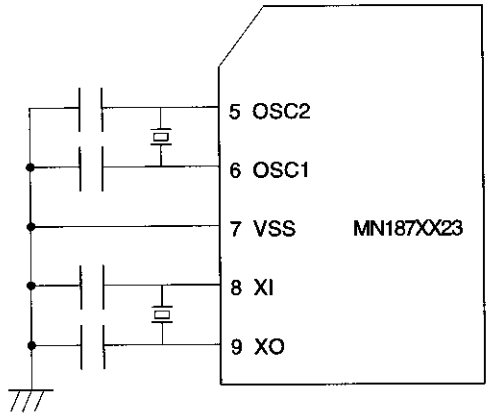


Figure 2-1-1 Oscillator Circuit Connection Diagram

The connection method shown in the figure on the left is known as self-oscillation. External clock is available when a clock from off-chip is used as the microcomputer's system clock.

2-1-2 Machine Cycle

Each state in the execution of an MN187XX23 instruction consists of a machine cycle comprising 4 clock cycles—S0, S1, S2, and S3—generated from the source oscillation (OSC1 and OSC2, or XI and XO).

CPU control and instruction execution use this machine cycle as the basic timing cycle. [☞ "MN1870 Series Instruction Manual"]

fosc is the source oscillation frequency, and fs is the system clock (machine cycle) frequency. The relationship between the two is: fs = fosc/4 (or fx/4)

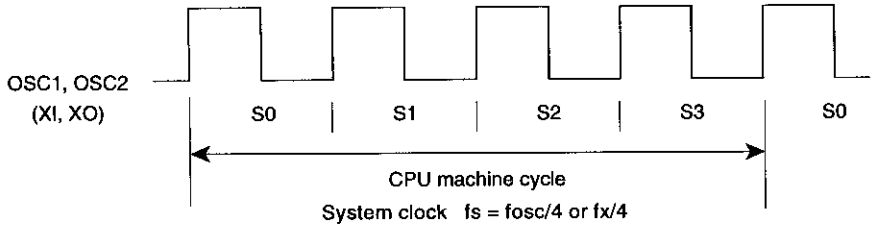


Figure 2-1-2 Machine Cycle

A machine cycle is generated from the source oscillation (OSC1 and OSC2, or XI and XO) and comprises 4 clock cycles—S0, S1, S2, and S3. [☞ Figure 2-1-2]

2-2 Instruction Execution Controller

2-2-1 Overview

The instruction execution control block is consist of a ROM area that stores instructions, an instruction queue buffer for instruction prefetching, an instruction register (IR), and an instruction decoder that decodes the contents of the instruction register and generates control signals for the various microcomputer blocks.

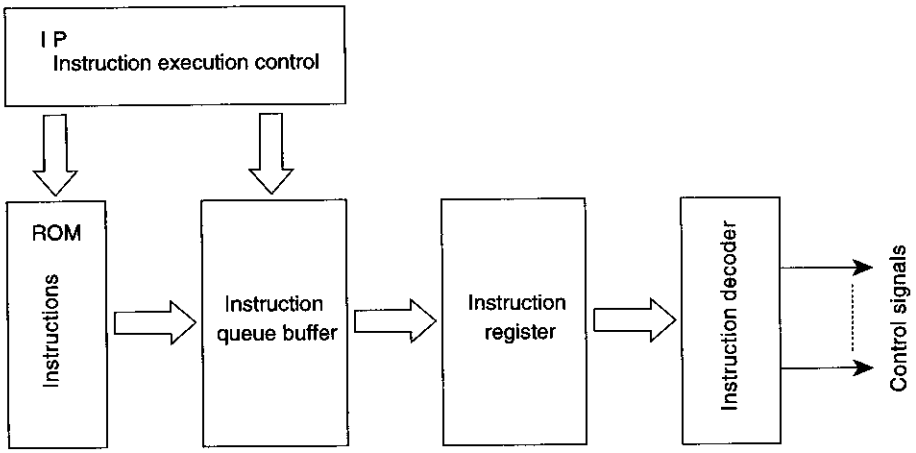


Figure 2-2-1 Instruction Execution Control Flowchart

2-2-2 ROM

ROM: Read Only Memory

ROM (read-only memory) is program memory that stores the program to be executed.

The MN1870 Series has separate instruction memory space (ROM) that holds instructions and data memory space (RAM) that holds data. ROM and RAM can both be accessed in the same CPU cycle, allowing fast instruction execution.
ROM can be used to store fixed data such as table values as well as instructions.

Table 2-2-1 Interrupt Handler Start Addresses

Interrupt Source		Vector Address	Priority
(Reset CPU)	(RESET)	X'0000'	High
External signal interrupt (edge input)	(IRQ0)	X'0004'	
External signal interrupt (edge input)	(IRQ1)	X'0006'	
External signal interrupt (edge input)	(IRQ2)	X'0008'	
External signal interrupt (edge input)	(IRQ3)	X'000A'	
Timer 0 interrupt	(TC0IRQ)	X'000C'	
Timer 1 interrupt	(TC1IRQ)	X'000E'	
Timer 2 interrupt	(TC2IRQ)	X'0010'	
Timer 3 interrupt	(TC3IRQ)	X'0012'	
Timer 4 interrupt	(TC4IRQ)	X'0014'	
Timer 5 interrupt	(TC5IRQ)	X'0016'	
Serial interface 0 interrupt	(SIF0IRQ)	X'0018'	
Serial interface 1 interrupt	(SIF1IRQ)	X'001A'	
Serial interface 2 interrupt	(SIF2IRQ)	X'001C'	
Key-scan interrupt	(KYSIRQ)	X'001E'	
Automatic data transfer interrupt	(ADTIRQ)	X'0020'	
Reserved interrupt	(RSVIRQ)	X'0022'	Low

Figure 2-2-2 shows the MN1873223's ROM address space and interrupt vector addresses.

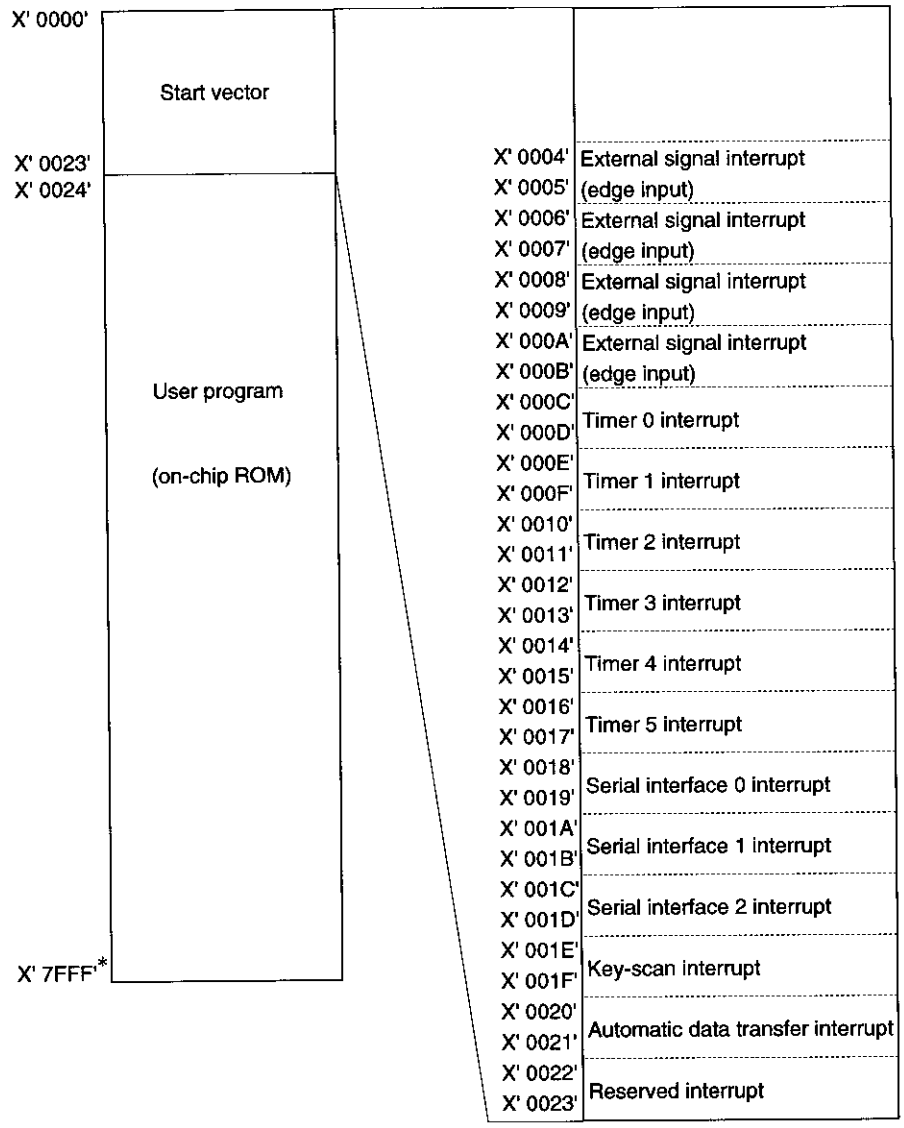


Figure 2-2-2 ROM Address Space and Interrupt Vector Addresses

* *As the ROM size varies according to the model, this address also varies as follows:*

MN1872423: X'5FFF'
MN1874023: X'9FFF'
MN1874823: X'BFFF'
MN187P6423: X'FFFF'
EP1876423: X'FFFF'

2-2-3 Instruction Register and Instruction Decoder

The CPU reads the next instruction to be executed from ROM and latches it in the instruction register (IR).

The instruction decoder (ID) decodes the contents of the CPU's instruction register in the instruction execution state, generates in sequence the control signals required to execute that instruction, and executes the instruction by controlling the various blocks in the chip.

2-2-4 Pipelining and Instruction Queue Operation

Pipelining is a process whereby instruction reading and decoding is performed simultaneously with instruction execution, so that as soon as execution of one instruction ends, the next instruction can be executed immediately. Pipelining thus speeds up instruction execution by enabling instructions to be executed in a continuous sequence. Pipelining is performed by means of the instruction queue and the instruction decoder.

The instruction queue is a single-stage instruction prefetch buffer. When the queue is empty in any state during instruction execution, control is performed to have the next instruction fetched into it.

In the last state of instruction execution, the first word (operation code) of the executing instruction is stored in the instruction register. At the same time, the next operand or operation code is fetched into the instruction queue, so that the next instruction can be executed immediately even if a direct address (da) or immediate data (imm) is needed in the first state of that instruction.

However, with some instructions, including branch instructions, the instruction queue becomes empty as soon as the operation code of the next instruction to be executed is stored in the instruction register in the last state. Therefore, a one-state queue wait occurs only in the case where the instruction queue is empty and a direct address (da) or immediate data (imm) is needed in the first state of the instruction to be executed.

☞* *For details of operation in the case of a queue wait, see the "MN1870 Series Instruction Manual."*

Control of the instruction queue is performed automatically by hardware, and need not be considered when writing a program. However, the operation of the instruction queue* must be understood when calculating program execution time.

The instruction decoder generated control signals for each instruction execution state by means of microprogram control.

The instruction decoder performs pipelining so that decoding is carried out one state before a control signal is needed. If a queue wait occurs due to instruction queue and instruction decoder pipelining, there is no loss of a state (loss of timing) due to instruction fetching and decoding.

When the single-stage instruction queue is empty, ROM is accessed. ROM access starts in the S2 cycle of a state and the ROM contents are read into the instruction queue in the next S1 cycle. The MN1870 Series uses pipelining for instruction decoding, performing instruction decoding and execution in parallel, and so increasing the instruction execution speed. [☞ Figure 2-2-3]

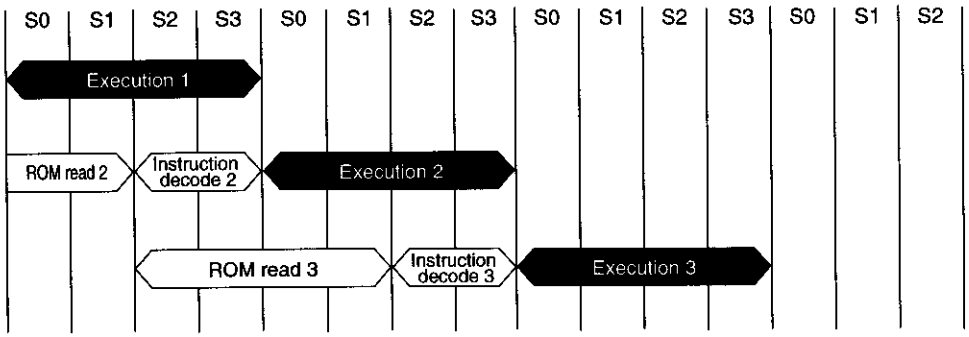


Figure 2-2-3 Basic Timing (No Queue Wait)

2-3 Pointer Register

2-3-1 Overview

The pointer register consists of four 16-bit registers: the instruction pointer (IP), stack pointer (SP), and data pointers (XP and YP).
A register is called a "pointer" when its contents are used to specify a memory address.

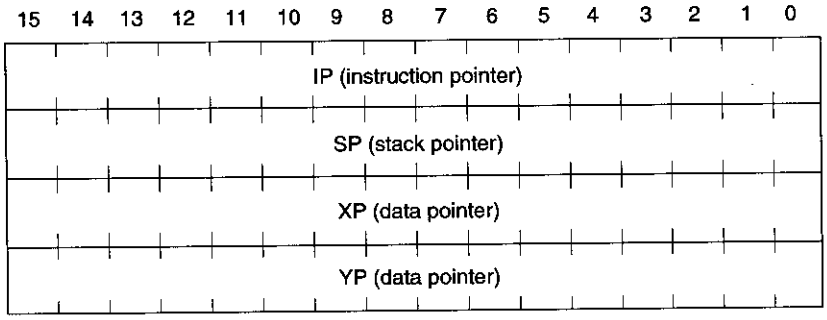


Figure 2-3-1 Pointer Register Configuration

2-3-2 Instruction Pointer (IP)

The instruction pointer (IP) is a 16-bit register that indicates the execution address of an instruction stored in ROM. This register controls the order of execution of instructions in the program memory.

2-3-3 Stack Pointer (SP)

The stack pointer (SP) indicates the address of the stack area that saves data using part of RAM.

The stack area holds the execution address of the next instruction (or the next address after a branch source address) in the case of an interrupt or CALL instruction, data saved/restored by the PUSH/POP instructions, and so on.

The data stored in the event of an interrupt consists of three bytes: the upper and lower address bytes of the instruction pointer when the interrupt is generated, and the flag status byte. The SP is decremented by 3.

The data stored in the case of a CALL instruction consists of two bytes: the upper and lower address bytes of the IP (the next address after the branch source address). The SP is decremented by 2.

The number of bytes of data stored when a PUSH instruction is executed depends on the operand.

① PUSH imm, PUSH (da)

The data stored is one-byte immediate data or the byte specified by a direct address. The SP is decremented by 1.

② PUSH XP, PUSH YP

The data stored is the two bytes of XP or YP. The SP is decremented by 2.

The stack pointer is set to X'0100' by hardware in a reset, and therefore the necessary SP value must be set again taking account of the amount of stack space required for interrupts, subroutine nesting, and PUSH/POP instructions.

When the stack area is used to write data, the SP is decremented before the write. When the stack area is used to read data, the SP is incremented after the read. The stack pointer is 16 bits long, and moves up and down within a 64-Kbyte area.

[☞ Figure 2-3-2]

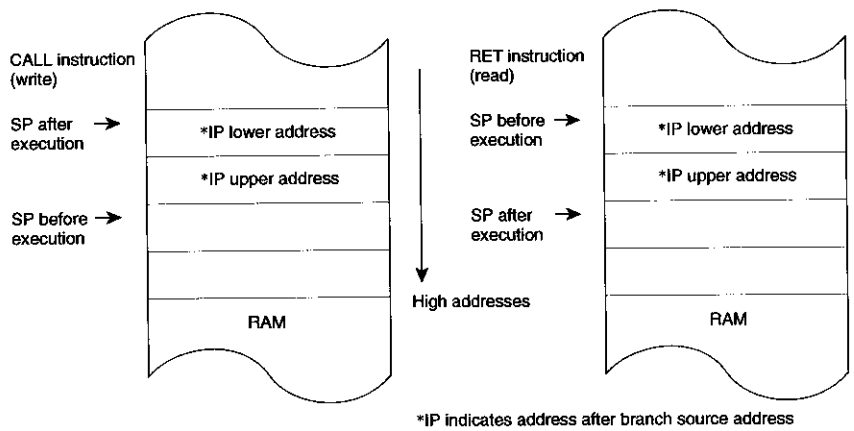


Figure 2-3-2 Stack Pointer (SP) Operation

2-3-4 Data Pointers (XP, YP)

The data pointers (XP, YP) are 16-bit data RAM address pointers.

In direct address mode (DF=1), in the case of a double-operand instruction the upper 8 bits of XP and the destination address are added together to indicate a RAM address, and the upper 8 bits of YP and the source address are added together to indicate a RAM address.

With other instructions, the upper 8 bits of XP and the destination address are added together to indicate a RAM address. [☞ Figure 2-3-3]

With register indirect addressing, the RAM address indicated by XP/YP is accessed.

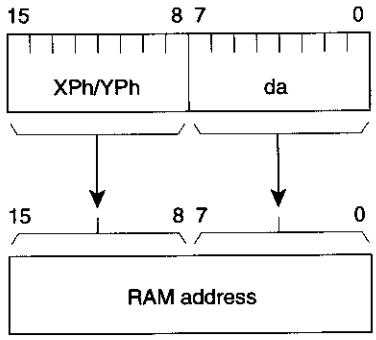


Figure 2-3-3 RAM Address Modification by Data Pointer

2-4 Arithmetic and Logic Block

2-4-1 Arithmetic and Logic Unit (ALU)

ALU
(Arithmetic Logical Unit)

The arithmetic and logic unit (ALU) performs an arithmetic operation (addition/subtraction, multiplication/division, decimal addition, increment/decrement, or compare) or a logic operation (AND, OR, XOR, complement, rotate, or nibble swap) on data input from the data bus to two 8-bit latches, and outputs the result to the data bus via an output latch.

2-5 Flag Block

2-5-1 Overview

The flag status byte consists of a 3-bit register comprising the carry flag (CF) and zero flag (ZF) (operation flags) and the direct flag (DF) (address control flag), and a control section. [☞ Figure 2-5-1]
The flags automatically record various information relating to the result of an operation. The flag status byte is saved to the stack area when an interrupt processing is executed, and restored on return from the interrupt.

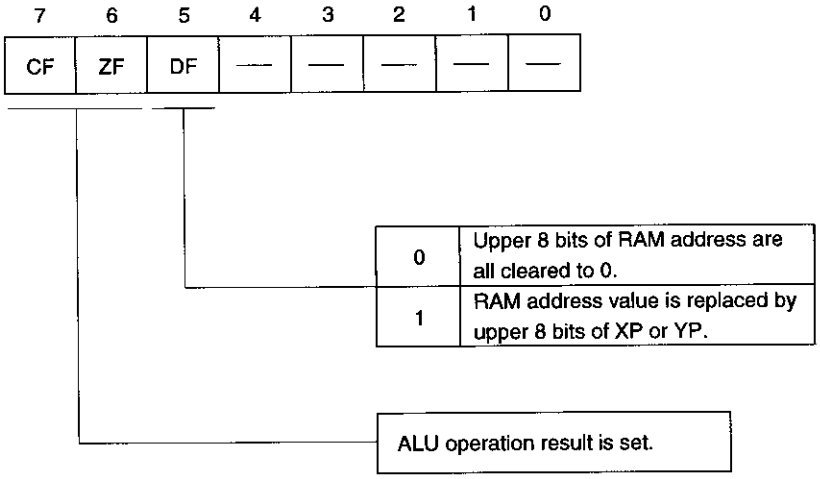


Figure 2-5-1 Flag Status Byte

2-5-2 Carry Flag (CF)

The carry flag is set when the ALU operation result overflows or underflows, and reset otherwise.
This bit can also be manipulated by the SET and CLR instructions.

2-5-3 Zero Flag (ZF)

The zero flag is set when the ALU operation result is 0, and reset otherwise.

2-5-4 Direct Flag (DF)

This is an address control flag, controlling the RAM address value in direct addressing mode. It is set and reset by instructions (SET DF, CLR DF). In a CPU reset, it is reset to 0 by hardware.
The direct flag controls direct addresses as shown below.

In direct addressing mode, a RAM address value is specified directly as the operation code operand value.

DF = 0: The upper 8 bits of the address are processed as 0.
DF = 1: The upper 8 bits of the XP or YP address are processed as the upper 8 bits of the RAM address.

- *Specifiable RAM address range*
  DF = 0: X'0'~X'FF'
  DF = 1: X'0'~X'FFFF'

2-6 Data Memory Block (RAM)

2-6-1 Overview

RAM: Random Access Memory

RAM is memory in which information can be written to and read from any address. MN187XX23 microcomputers have from 512 to 1024 bytes of RAM, depending on the model. The RAM is memory-mapped, and data memory (RAM), special registers, and I/O ports are all located in the RAM memory space. The special registers and I/O ports are located in the area starting at address X'000', and the remaining area is used as data RAM and the stack area.

In the reset state the stack pointer is set to X'100'. It must therefore be initialized to the required value by the program.

2-6-2 RAM Memory Space

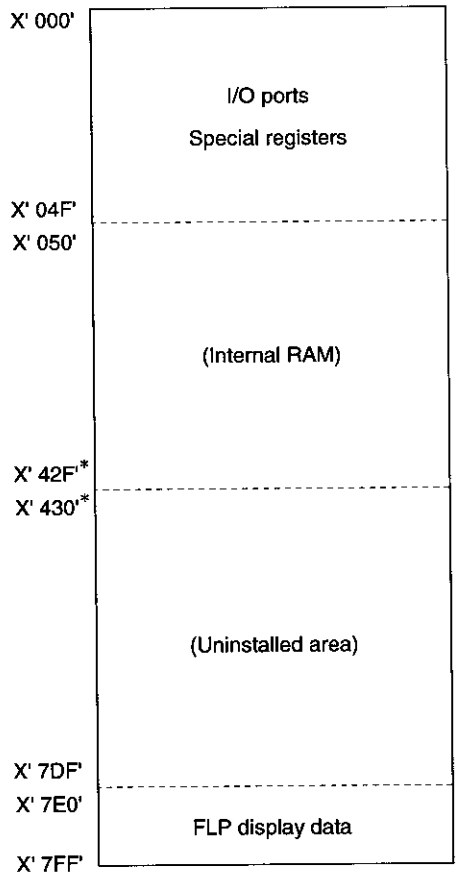


Figure 2-6-1 MN1873223 RAM Address Space

* *As the RAM size varies according to the model, the last address of the internal RAM area also varies as follows:*

MN1872423:	*X'22F'*
MN1874023:	*X'42F'*
MN1874823:	*X'42F'*
MN187P6423:	*X'7CF'*
EP1876423:	*X'2AF'*

X' 000'	PORT0	PORT1	PORT2	PORT3	PORT4	PORT5	PORT6	PORT7
X' 008'	PORT8	PORT9	PFSR0	PFSR1	PFSR2	DNDR	FCMR	—
X' 010'	IRQM	—	IF0	IF1	IE0	IE1	CPUM	—
X' 018'	P0DIR	P1DIR	P2DIR	P3DIR	P4DIR	P9DIR	—	—
X' 020'	SIBUF0	SIBUF1	SIM0	SIM1	SBC0	SBC1	—	RMC
X' 028'	TC0/BC0	TM0	TC1/BC1	TM1	TM2	TC3/BC3	ADC	ADBUF
X' 030'	PWMC	PWMHR	PWMLR	PWMHL	PWMLL	SIBUF2	SIM2	SBC2
X' 038'	ADTC	ADTB	TAPL	TAPH	SAPL	SAPH	DAPL	DAPH
X' 040'	P0RON	P1RON	P2RON	P3RON	P4RON	P9RON	—	—
X' 048'	TC4/BC4	TM4	TC5/BC5	TM5	—	—	PWMR2	—
X' 050' ↓ X' 428'*1				Internal RAM area				
X' 430'*1 ↓ X' 7DF'				Uninstalled area				
X' 7E0' ↓ X' 7FF'				FLP display data*2				

— *indicates uninstalled RAM. (R/W prohibited)*

Figure 2-6-2 MN1873223 RAM Address Space

*1 *As the RAM size varies according to the model, the last address of the internal RAM area also varies as follows:*
MN1872423: X'22F'
MN1874023: X'42F'
MN1874823: X'42F'
MN187P6423: X'7CF'
EP1876423: X'2AF'

2 If the area from X'7E0' to X'7FF' is not used for FLP display, it can be used as ordinary RAM.

2-7 Special Register Block

2-7-1 Overview

In the MN187XX23, addresses X'000' to X'04F' are allocated as the special register area.

2-7-2 List of Special Registers

Table 2-7-1 Special Registers (1/3)

Register Symbol	Name	RAM Address	Reference Page
PORT0	Port 0	X'000'	80
PORT1	Port 1	X'001'	80
PORT2	Port 2	X'002'	80
PORT3	Port 3	X'003'	80
PORT4	Port 4	X'004'	80
PORT5	Port 5	X'005'	80
PORT6	Port 6	X'006'	80
PORT7	Port 7	X'007'	80
PORT8	Port 8	X'008'	80
PORT9	Port 9	X'009'	80
PFSR0	Port/FLP select register 0	X'00A'	164
PFSR1	Port/FLP select register 1	X'00B'	165
PFSR2	Port/FLP select register 2	X'00C'	166
DNDR	Digit number/dimmer register	X'00D'	167
FCMR	FLP clock mode register	X'00E'	168
IRQM	Interrupt mode control register	X'010'	58
IF0	Interrupt request flag register 0	X'012'	59
IF1	Interrupt request flag register 1	X'013'	59
IE0	Interrupt enable flag register 0	X'014'	61
IE1	Interrupt enable flag register 1	X'015'	63
CPUM	CPU mode register	X'016'	66
P0DIR	Port 0 direction control register	X'018'	80
P1DIR	Port 1 direction control register	X'019'	80
P2DIR	Port 2 direction control register	X'01A'	80
P3DIR	Port 3 direction control register	X'01B'	80

Table 2-7-1 Special Registers (2/3)

Register Symbol	Name	RAM Address	Reference Page
P4DIR	Port 4 direction control register	X'01C'	80
P9DIR	Port 9 direction control register	X'01D'	80
SIBUF0	Transmit/receive shift buffer 0	X'020'	143
SIBUF1	Transmit/receive shift buffer 1	X'021'	143
SIM0	Serial interface mode register 0	X'022'	147
SIM1	Serial interface mode register 1	X'023'	148
SBC0	Serial interface bit counter 0	X'024'	144
SBC1	Serial interface bit counter 1	X'025'	145
RMC	Remote control/noise filter control register	X'027'	202, 213
TC0/BC0	Programmable timer counter 0 (timer latch 0/binary counter 0)	X'028'	116
TM0	Timer mode register 0	X'029'	119
TC1/BC1	Programmable timer counter 1 (timer latch 1/binary counter 1)	X'02A'	116
TM1	Timer mode register 1	X'02B'	120
TM2	Timer mode register 2	X'02C'	121
TC3/BC3	Programmable timer counter 3 (timer latch 3/binary counter 3)	X'02D'	117
ADC	A/D control register	X'02E'	194
ADBUF	AD buffer	X'02F'	195
PWMC	PWM control register	X'030'	186
PWMHR	PWM0 data register upper 7 bits	X'031'	187
PWMLR	PWM0 data register lower 7 bits	X'032'	187
PWMHL	PWM0 data latch upper 7 bits	X'033'	187
PWMLL	PWM0 data latch lower 7 bits	X'034'	187
SIBUF2	Transmit/receive shift buffer 2	X'035'	143
SIM2	Serial interface mode register 2	X'036'	149
SBC2	Serial interface bit counter 2	X'037'	146
ADTC	Automatic data transfer control register	X'038'	227
ADTB	Transfer byte counter	X'039'	228
TAPL	Data transfer target address register lower 8 bits	X'03A'	229
TAPH	Data transfer target address register upper 8 bits	X'03B'	229
SAPL	Data transfer source address register lower 8 bits	X'03C'	229
SAPH	Data transfer source address register upper 8 bits	X'03D'	229
DAPL	Data transfer destination address register lower 8 bits	X'03E'	230
DAPH	Data transfer destination address register upper 8 bits	X'03F'	230
P0RON	Port 0 pull-up control register	X'040'	80

Table 2-7-1 Special Registers (3/3)

Register Symbol	Name	RAM Address	Reference Page
P1RON	Port 1 pull-up control register	X'041'	80
P2RON	Port 2 pull-up control register	X'042'	80
P3RON	Port 3 pull-up control register	X'043'	80
P4RON	Port 4 pull-up control register	X'044'	80
P9RON	Port 9 pull-up control register	X'045'	80
TC4/BC4	Programmable timer counter 4 (timer latch 4/binary counter 4)	X'048'	117
TM4	Timer mode register 4	X'049'	122
TC5/BC5	Programmable timer counter 5 (timer latch 5/binary counter 5)	X'04A'	118
TM5	Timer mode register 5	X'04B'	123
PWMR2	PWM2 data register	X'04E'	188

2-8 Interrupt Controller

2-8-1 Overview

The interrupt controller interrupts the flow of the executing program in response to an interrupt request, saves the program status at the point of interruption on the stack, and controls the start of execution of an interrupt handler according to the interrupt source.

With the MN187XX23, the user can use 15 kinds of interrupt sources except a reset. [☞ Figure 2-8-1]

Vector addressing is used to specify the start address of an interrupt handler, and the user can freely specify vector addresses set in ROM in accordance with the various interrupt sources. [☞ Table 2-8-1]

2-8-2 Interrupt Sources and Interrupt Vector Addresses

Interrupt mode control register (IRQM: X'010' R/W)
Bit 4: IEMASK
(interrupt enable mask flag)

An interrupt is accepted by the interrupt controller only when both the interrupt request flag (IF) and interrupt enable flag (IE) for that particular interrupt are set while the interrupt enable mask flag (IEMASK) is cleared.

When an interrupt is accepted, execution of the interrupt handler is started. However, while an interrupt handler is current being executed, unless multiple interrupts are permitted by the program, handling of the newly accepted interrupt is deferred until the currently executing interrupt handler completes.

[☞ Figure 2-8-2]

If a number of interrupts are accepted simultaneously, they are handled in order according to the priority levels determined by hardware. [☞ Table 2-8-1]

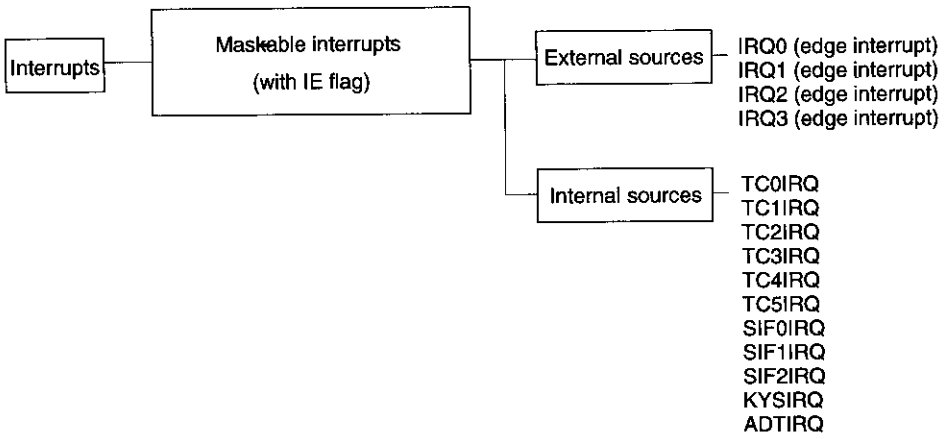


Figure 2-8-1 Interrupt Sources

Table 2-8-1 Interrupt Handling Vector Addresses

Interrupt Source		Vector Address	Priority
(Reset CPU)	(RESET)	X'0000'	High
External signal interrupt (edge input)	(IRQ0)	X'0004'	
External signal interrupt (edge input)	(IRQ1)	X'0006'	
External signal interrupt (edge input)	(IRQ2)	X'0008'	
External signal interrupt (edge input)	(IRQ3)	X'000A'	
Timer 0 interrupt	(TC0IRQ)	X'000C'	
Timer 1 interrupt	(TC1IRQ)	X'000E'	
Timer 2 interrupt	(TC2IRQ)	X'0010'	
Timer 3 interrupt	(TC3IRQ)	X'0012'	
Timer 4 interrupt	(TC4IRQ)	X'0014'	
Timer 5 interrupt	(TC5IRQ)	X'0016'	
Serial interface 0 interrupt	(SIF0IRQ)	X'0018'	
Serial interface 1 interrupt	(SIF1IRQ)	X'001A'	
Serial interface 2 interrupt	(SIF2IRQ)	X'001C'	
Key-scan interrupt	(KYSIRQ)	X'001E'	
Automatic data transfer interrupt	(ADTIRQ)	X'0020'	
Reserved interrupt	(RSVIRQ)	X'0022'	Low

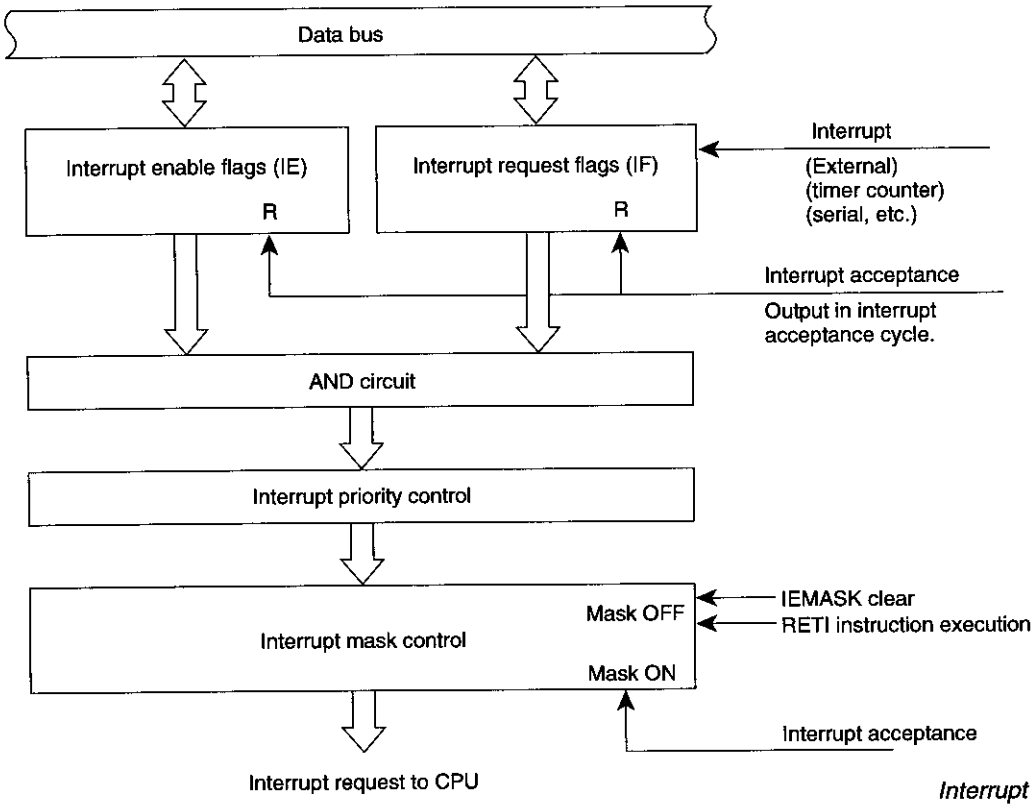


Figure 2-8-2 Interrupt Control Block

Interrupt mode control register
(IRQM: X'010' R/W)
Bit 4: IEMASK
(interrupt enable mask flag)

2-8-3 Interrupt Acceptance

The interrupt acceptance operation begins when a serial or other interrupt is generated, and ends with a branch to the start of the interrupt handler.

[☞ Figure 2-8-3]

When an interrupt is generated, the interrupt request flag (IF) for the corresponding interrupt level is first set. If the interrupt enable flag (IE) for the level corresponding to the interrupt request flag (IF) is set at this time, the generated interrupt receives the right to be accepted. If no other interrupt handling is being performed, the interrupt with the highest priority level of those generated is accepted. While other interrupt handling is in progress, unless multiple interrupts are permitted, the new interrupt is accepted after the current interrupt handling is completed.

Interrupt acceptance processing involves operations similar to those performed when a CALL instruction is executed. In the interrupt acceptance cycle, the instruction pointer (IP) and flag status byte (FS) are written to ("pushed onto") the stack area in RAM.

The values of status flags CF, ZF, and DF are not changed by interrupt generation.

Next, to determine the start address of the interrupt handler, the predetermined interrupt handler start vector address for the relevant interrupt source is read from a specific address in ROM, and set in the instruction pointer (IP).

The IE and IF flags for the level corresponding to that for which interrupt acceptance is current being performed are to be reset, and IEMASK is to be set, causing hardware masking of the IE flags for all levels and so disabling acceptance of all interrupts.

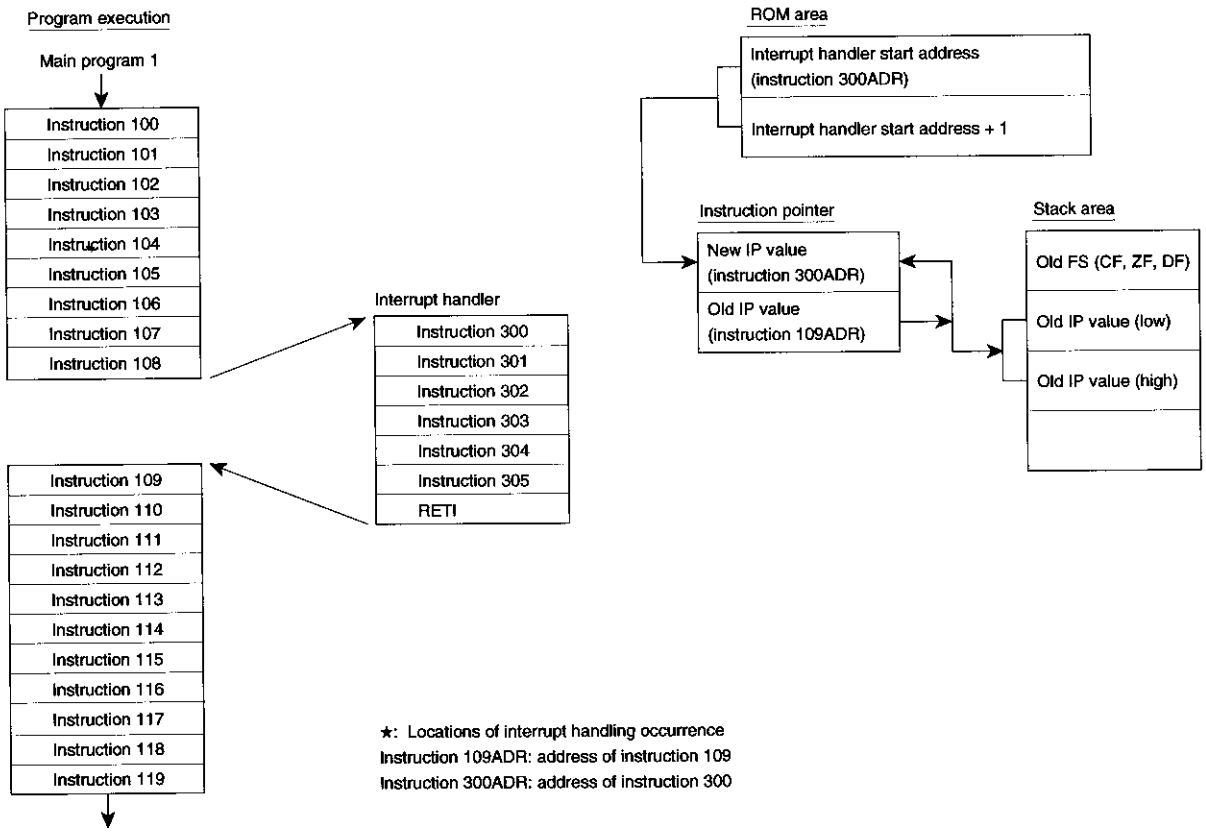


Figure 2-8-3 Interrupt Operation

2-8-4 Return from Interrupt

The interrupt return processing for returning from interrupt handler execution to the main program is performed by the RETI instruction.

The RETI instruction operates in a similar way to the RET instruction (Return) used to return from a subroutine to the main routine. [☞ Figure 2-8-4]
When the RETI instruction is executed, the pre-interrupt instruction pointer (IP) and flag status byte (FS) values previously pushed onto the stack area RAM are restored to the instruction pointer (IP) and flag status byte (FS), and the program flow is restored to its state prior to the interrupt.
The interrupt response time, taking the worst case in which a 10-cycle divide instruction is being executed, is a maximum of 12 machine cycles from interrupt generation until the start of interrupt acceptance, and 18 machine cycles until the branch to the start of the interrupt handler.

Interrupt acceptance in the MN187XX23 can be temporarily disabled either by setting the interrupt enable mask flag (IEMASK) or by clearing the interrupt enable flags (IE). As interrupt handling uses pipelining, whichever method is used it is possible that an interrupt may be accepted after the disabling instruction is executed.

1. Using interrupt mask flag
   If an interrupt is accepted immediately after execution of the instruction that sets the interrupt mask flag, the set interrupt mask flag will be cleared.
   ● Remedy
   Monitor the interrupt mask flag after setting it to confirm that it is actually set.

```
loop1   set     (IRQM) IEMASK
        tlbz    (IRQM) IEMASK, $loop1
```

2. Using interrupt enabling
   If an interrupt is accepted immediately after execution of the instruction that clears interrupt enabling, and the interrupt enable flag is set during interrupt handling, the cleared flag will be set again.
   ● Remedy (in case of IRQ0)
   Monitor the interrupt enable flag after clearing it to confirm that it is actually cleared.

```
loop1   clr     (IE0) IEIRQ0
        tlbnz   (IE0) IEIRQ0, $loop1
```

When the RETI instruction is executed, IEMASK is reset and the hardware interrupt enable flags (IE) are unmasked. Consequently, all interrupt enable flags (IE) that were set immediately before interrupt acceptance become valid. However, the IE flag for the interrupt level currently being handled is cleared in the interrupt acceptance cycle, and so does not become valid unless set again in the interrupt handler.

Interrupt mode control register (IRQM: X'010' R/W)
Bit 4: IEMASK
(interrupt enable mask flag)

2-8-5 Stack Operation in Case of Interruption

The stack level at the time of interrupt acceptance and interrupt return varies by only the 3 bytes of the instruction pointer (IP) and flag status byte (FS) that are pushed or popped.

With a normal interrupt, the instruction pointer (IP) and flag status byte (FS) are pushed, and therefore a 3-byte stack area is needed. Consequently, the stack pointer (SP) value is incremented by 3 on return from the interrupt. [☞ Figure 2-8-4]

In the interrupt acceptance cycle, first the stack pointer (SP) is decremented, then the 16-bits of the instruction pointer (IP) and the 8 bits of the flag status byte (FS) are written to the stack area RAM. The stack pointer (SP) is therefore decremented by 3 relative to its value before the interrupt.

Interrupt return is performed by means of the RETI instruction. The RETI instruction restores the flag status byte (FS) and instruction pointer (IP), which were pushed onto the stack area RAM in the interrupt acceptance cycle, to their original values. After the value in the RAM indicated by the stack pointer (SP) has been read, the stack pointer (SP) is incremented.

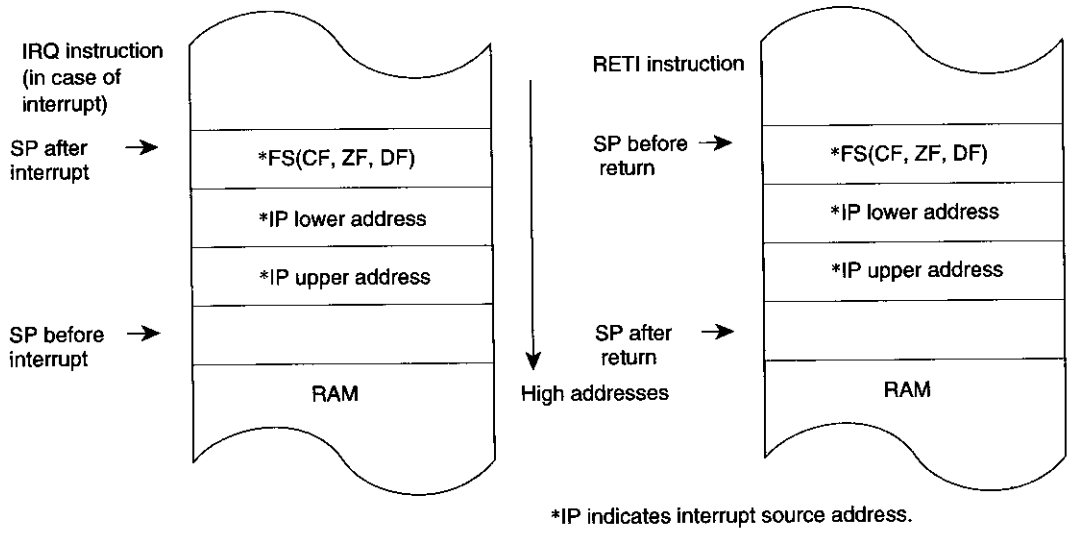


Figure 2-8-4 Stack Pointer (SP) Operation in Case of Interrupt

2-8-6 Interrupt Enable Hardware Mask

A hardware mask function is provided which disables all levels of interrupts when the interrupt acceptance cycle is started, to prevent further interrupt acceptance.

In the interrupt handling cycle, the interrupt enable mask flag (IEMASK) is set and interrupt enable flags (IE) are cleared to disable new interrupt acceptance.

Interrupt mode control register
(IRQM: X'010' R/W)
Bit 4: IEMASK

Hardware masking of interrupts is released on return from the interrupt (execution of the RETI instruction), and can also be released by resetting IEMASK. Multiple interruption is not enabled unless IEMASK is reset. [☞ Figure 2-8-5]

IRQ: External interrupt
TCIRQ: Timer interrupt
IFIRQ: External interrupt request flag
IEIRQ: External interrupt enable flag
IFTC: Timer interrupt request flag
IETC: Timer interrupt enable flag

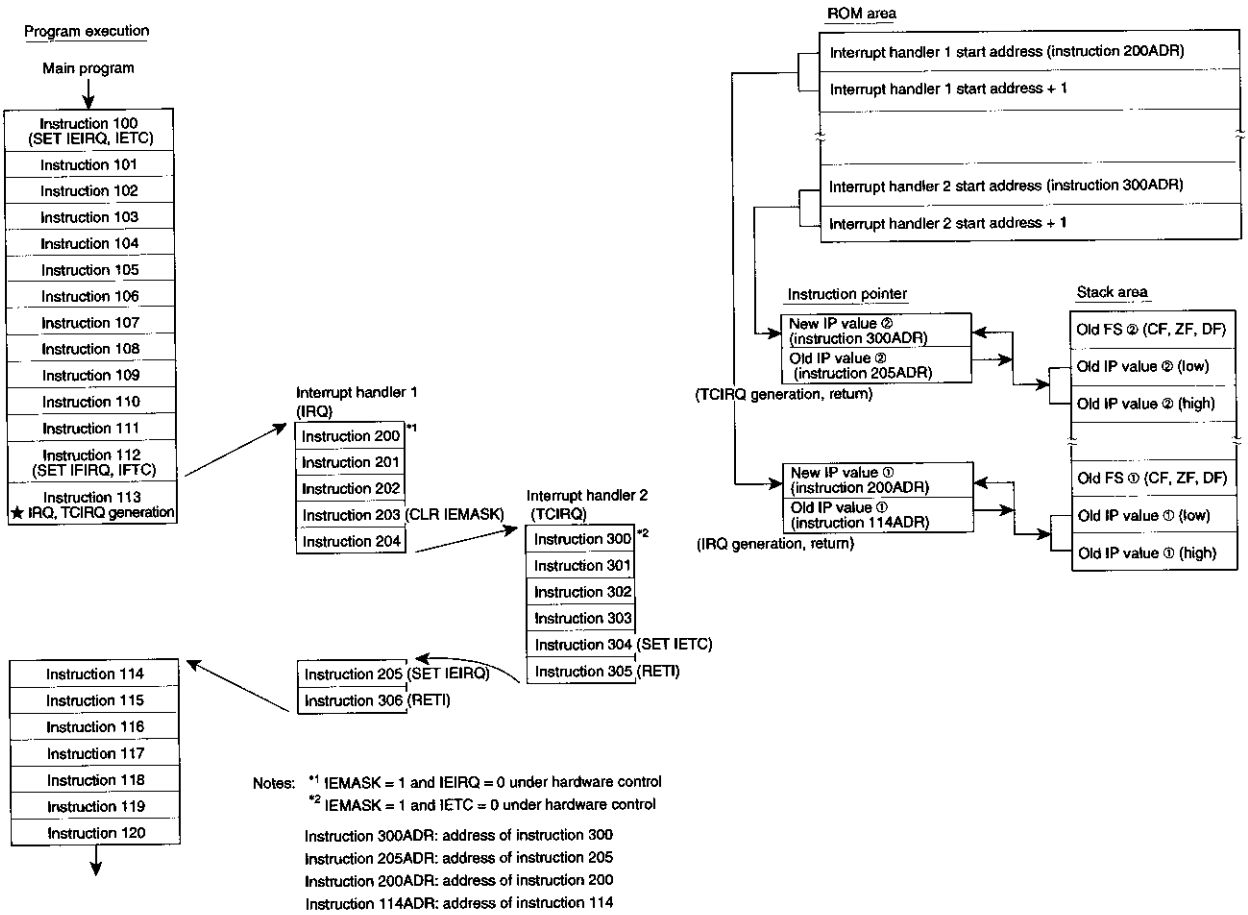


Figure 2-8-5 Multiple Interrupt Operation

2-8-7 Interrupt Control Registers

The MN187XX23 has the following control registers. Interrupt control, interrupt request flags, and interrupt enable flags are all forms of register control.

[☞ Table 2-8-2]

Table 2-8-2 Interrupt Control Registers

Register Abbreviation	RAM Address	R/W	Register Name
IRQM	X'010'	R/W	Interrupt mode control register
IF0	X'012'	R/W	Interrupt request flag register 0
IF1	X'013'	R/W	Interrupt request flag register 1
IE0	X'014'	R/W	Interrupt enable flag register 0
IE1	X'015'	R/W	Interrupt enable flag register 1

These registers are described in detail below.

(1) Interrupt mode control register (IRQM)

■ IRQM

The interrupt mode control register (IRQM) is allocated to RAM address X'010', and performs interrupt control.

● EDGEDIR0~3

Bits 0 to 3 specify the active edge of the external interrupt signals (IRQ0~3). When one of these bits is modified, the interrupt enable flag (IF) may be set; therefore, the interrupt enable flag (IE) should be masked before modifying these bits. These bits are cleared to 0 by a reset.

● IEMASK

Bit 4 specifies whether acceptance of all interrupts is enabled. It masks all interrupt enable flags (IE) and temporarily disables acceptance of all interrupts. It is set to 1 by interrupt acceptance. To enable multiple interrupts, this bit should be cleared by an instruction in the interrupt handler. The IEMASK bit is cleared to 0 by the RETI instruction or a reset.

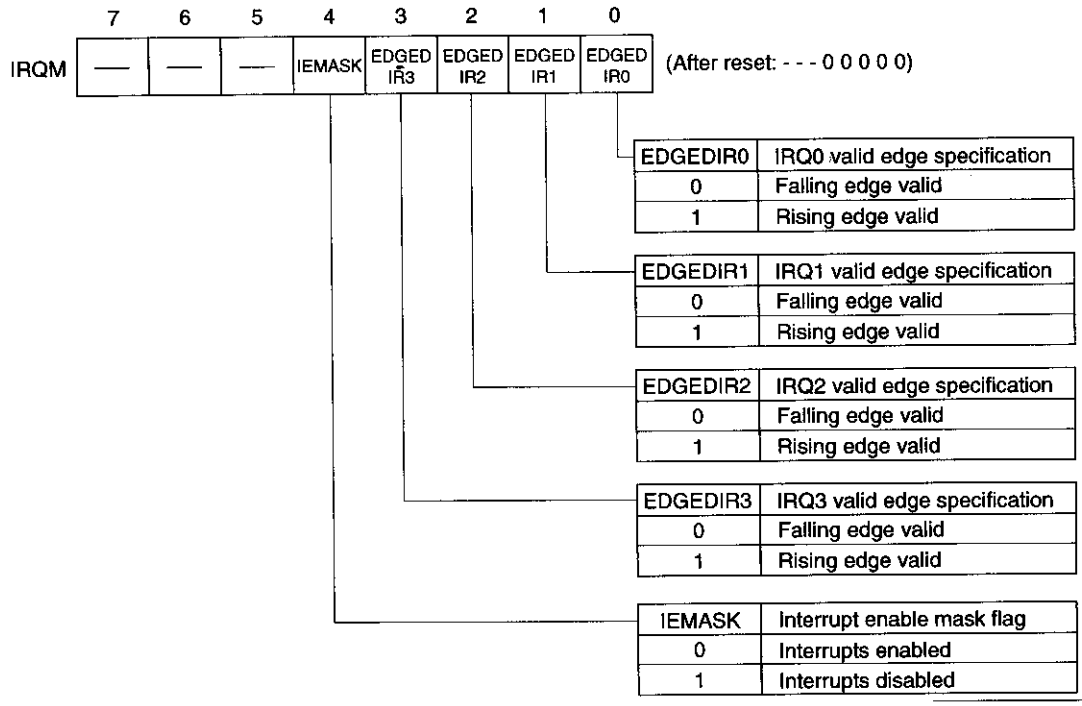


Figure 2-8-6 Interrupt Mode Control Register (IRQM: X'010' R/W))

— indicates an uninstalled bit. An undefined value is returned if read.

(2) Interrupt request flag registers (IF)

The interrupt request flag registers (IF0, IF1) are set by generation of an interrupt.

■ Interrupt request flag register 0 (IF0)

IF0 bits are set to 1 by generation of an interrupt, and upon reset or interrupt acceptance, the interrupt request flag for the accepted interrupt only is cleared to 0. All bits are cleared to 0 by a reset. [☞ Figure 2-8-7]

■ Interrupt request flag register 1 (IF1)

IF1 bits are set to 1 by generation of an interrupt, and upon reset or interrupt acceptance, the interrupt request flag for the accepted interrupt only is cleared to 0. All bits are cleared to 0 by a reset. [☞ Figure 2-8-8]

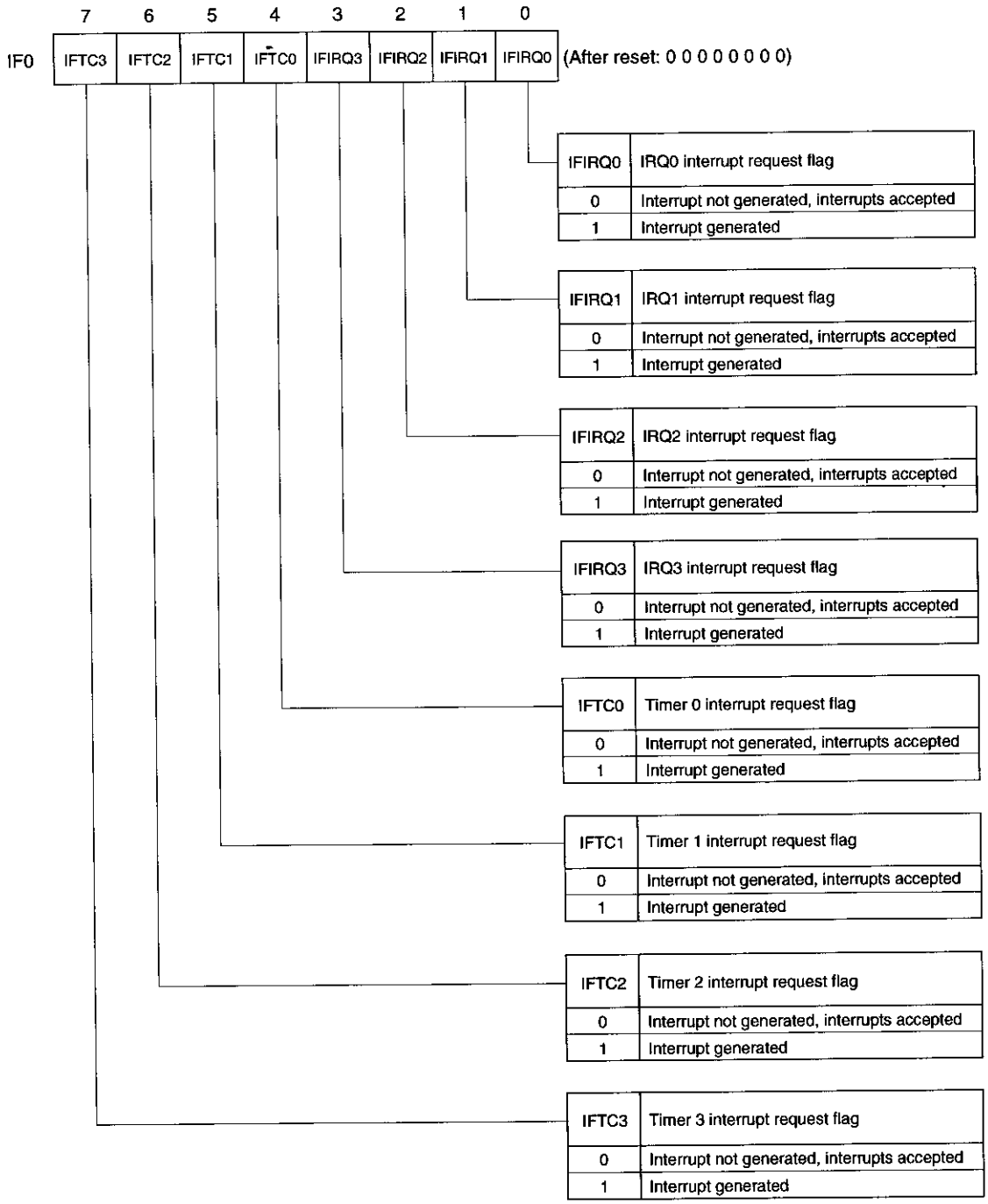


Figure 2-8-7 Interrupt Request Flag Register 0 (IF0: X'012' R/W)

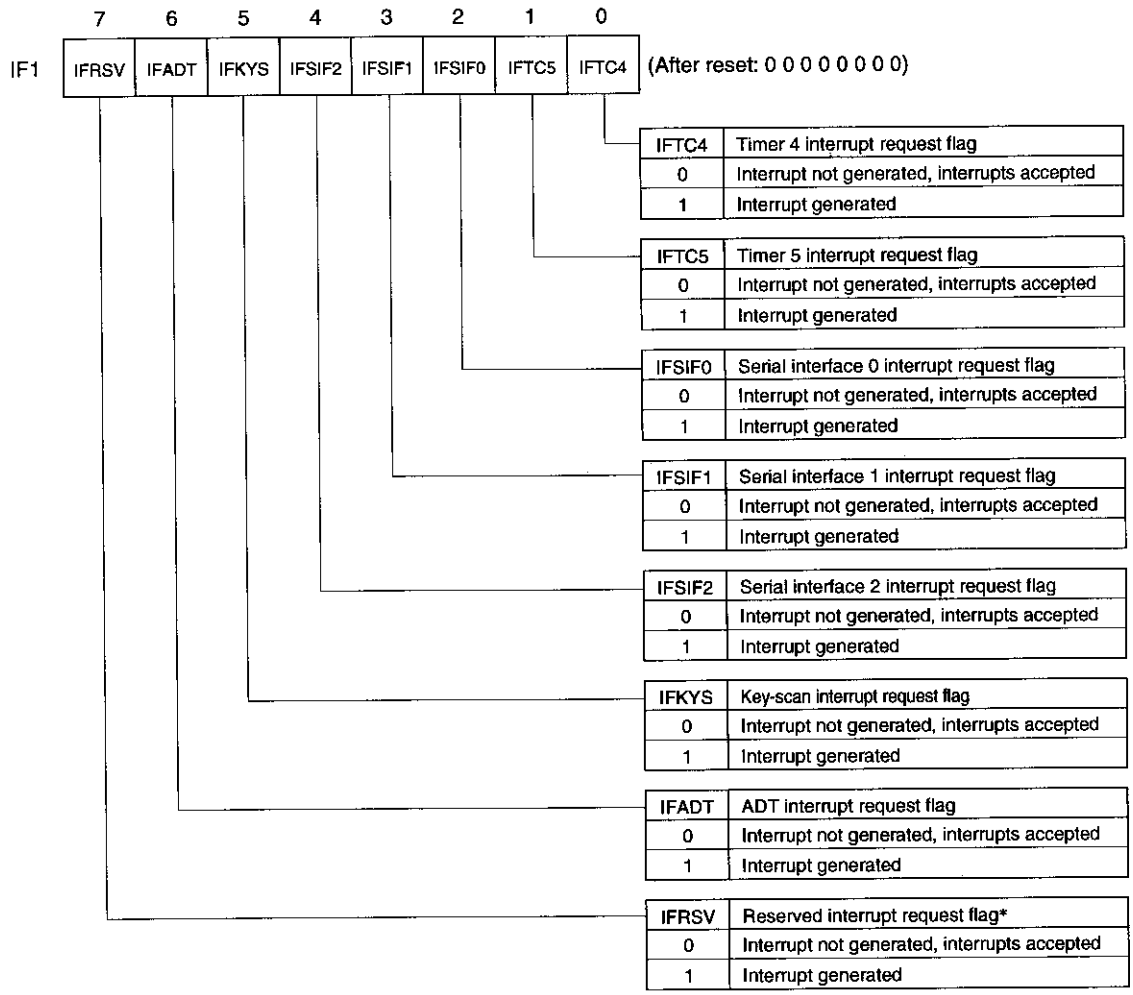


Figure 2-8-8 Interrupt Request Flag Register 1 (IF1: X'013' R/W)

(3) Interrupt enable flag registers (IE)

The interrupt enable flag registers (IE0, IE1) control interrupt enabling.

■ Interrupt enable flag register 0 (IE0)

IE0 specifies whether interrupt request acceptance is enabled.

When interrupts are to be accepted, the IEMASK bit in the interrupt mode control register (IRQM) must be cleared to 0.

When an interrupt is generated, the interrupt enable flag for the accepted interrupt only is cleared to 0.

All bits are cleared to 0 by a reset. [☞ Figure 2-8-9]

* *The reserved interrupt (RSV) is reserved for future addition of an interrupt source. However, if 1 is written to both IFRSV and IERSV, the program will branch to the address indicated by vector address X'0022'.*

Interrupt mode control register (IRQM: X'010' R/W)
Bit 4: IEMASK

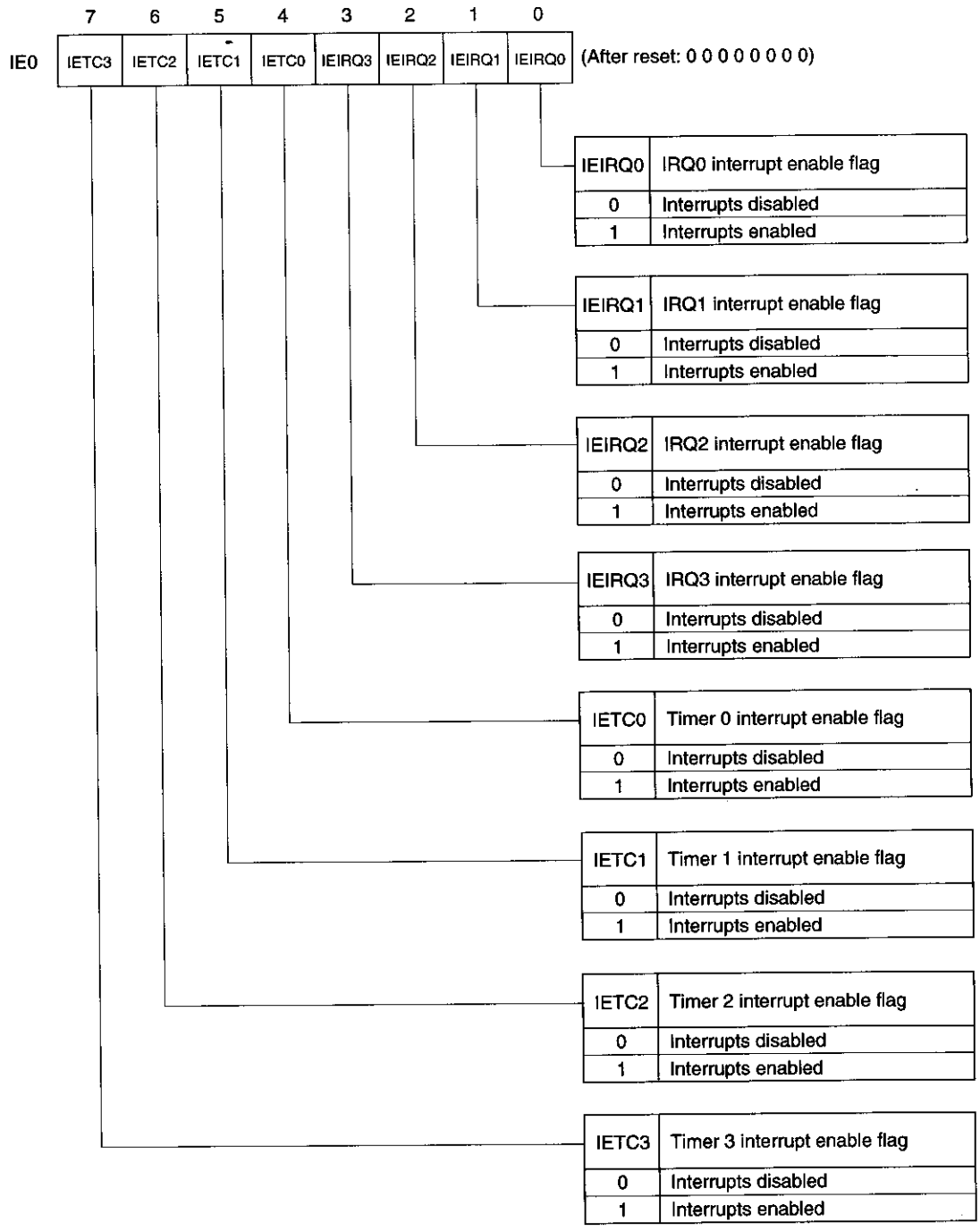


Figure 2-8-9 Interrupt Enable Flag Register 0 (IE0: X'014' R/W)

■ Interrupt enable flag register 1 (IE1)

IE1 specifies whether individual interrupt request acceptance is enabled.

When interrupts are to be accepted, the IEMASK bit in the interrupt mode control register (IRQM) must be cleared to 0.

When an interrupt is generated, the interrupt enable flag for the accepted interrupt only is cleared to 0.

All bits are cleared to 0 by a reset. [☞ Figure 2-8-10]

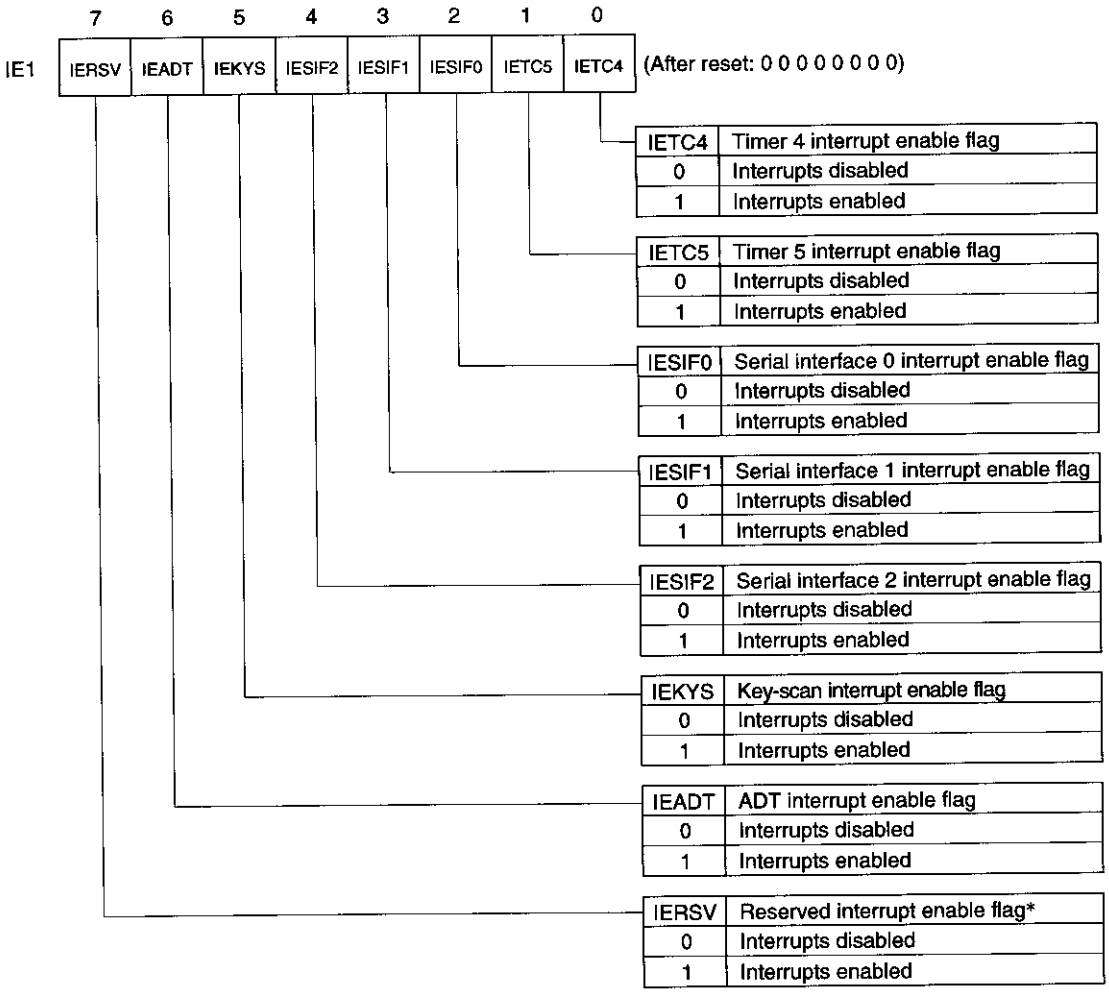
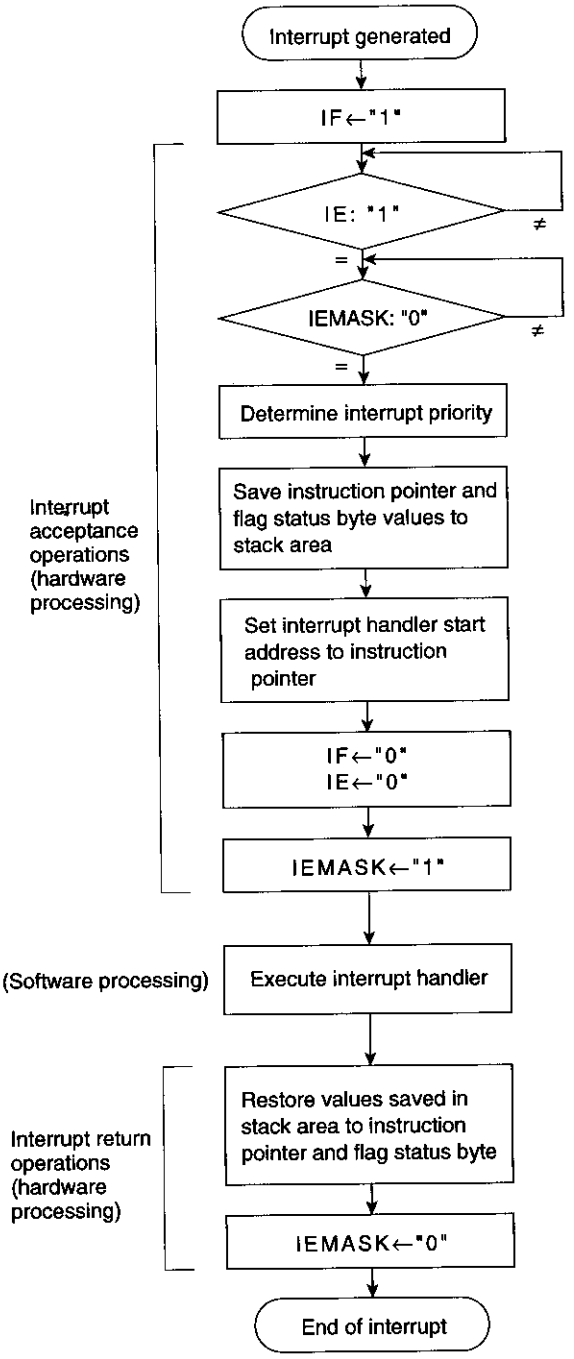


Figure 2-8-10 Interrupt Enable Flag Register 1 (IE1: X'015' R/W)

* *The reserved interrupt (RSV) is reserved for future addition of an interrupt source. However, if 1 is written to both IFRSV and IERSV, the program will branch to the address indicated by vector address X'0022'.*

2-8-8 Outline Interrupt Flowchart



IF (interrupt request flag) and IE (interrupt enable flag) must both be set.

Determination of whether an interrupt handler is executing. If so, the interrupt is held pending until the handler complete. If IEMASK (interrupt enable mask flag) is 1, an interrupt handler is executing.

If interrupts are generated simultaneously, the highest-priority interrupt is selected according to the interrupt priorities predetermined by hardware.
Saved to the stack area, to be restored on completion of the interrupt.

The interrupt handler vector address is fetched from ROM and set to the instruction pointer.

The accepted interrupt IF (interrupt request flag) and IE (interrupt enable flag) are reset.

IEMASK (interrupt enable mask flag) is set and IE (interrupt enable flags) are masked by hardware to disable acceptance of other interrupts.

Execution of the interrupt handler is started, and ends with an RETI instruction. To enable the relevant interrupt again, IF is set to 1 in the interrupt handler.

Execution of the RETI instruction restores the values saved in the stack area to the instruction pointer and flag status byte, and restores the pre-interrupt program flow.

IEMASK (interrupt enable mask flag) is reset, and all interrupts accepted directly before this interrupt are enabled. However, since the IE (interrupt enable flag) for the interrupt now completed was cleared during acceptance, this interrupt cannot be accepted. (If necessary, the interrupt enable flag should be set.)

Figure 2-8-11 Outline Flowchart of Processing from Generation to Termination of Interrupt

2-8-9 Example of Interrupt Vector Address Setup Program Coding

A sample program for setting up vector addresses is shown below.

```
        absolute    0           ;Start at absolute address 0
reset   equ         *           ;Reset processing start address specification
        dw          a(start)    ;Reservation of start vector address storage area
        dw          a(dummy)    ;Reservation of dummy process vector address storage area
        dw          a(irq0)     ;Reservation of external signal interrupt 0 vector address storage area
        dw          a(irq1)     ;Reservation of external signal interrupt 1 vector address storage area
        dw          a(irq2)     ;Reservation of external signal interrupt 2 vector address storage area
        dw          a(irq3)     ;Reservation of external signal interrupt 3 vector address storage area
        dw          a(tc0irq)   ;Reservation of timer 0 interrupt vector address storage area
        dw          a(tc1irq)   ;Reservation of timer 1 interrupt vector address storage area
        dw          a(tc2irq)   ;Reservation of timer 2 interrupt vector address storage area
        dw          a(tc3irq)   ;Reservation of timer 3 interrupt vector address storage area
        dw          a(tc4irq)   ;Reservation of timer 4 interrupt vector address storage area
        dw          a(tc5irq)   ;Reservation of timer 5 interrupt vector address storage area
        dw          a(sif0irq)  ;Reservation of serial interface 0 interrupt vector address storage area
        dw          a(sif1irq)  ;Reservation of serial interface 1 interrupt vector address storage area
        dw          a(sif2irq)  ;Reservation of serial interface 2 interrupt vector address storage area
        dw          a(kysirq)   ;Reservation of key-scan interrupt vector address storage area
        dw          a(adtirq)   ;Reservation of automatic data transfer interrupt vector address storage area
        dw          a(rsvirq)   ;Reservation of reserved interrupt vector address storage area
dummy   equ         reti        ;Dummy processing
start   equ         *           ;Start address specification
```

Figure 2-8-12 Example of Interrupt Vector Address Setup Program

In this example, since no interrupt vector is allocated to X'0002' and X'0003', the reti instruction address is written as dummy processing. Also, X'0022' and X'0023' are allocated as reserved interrupt addresses, and therefore the reti instruction address is written as dummy processing.

2-9 Clock Controller

2-9-1 Overview

The MN187XX23 has two operating modes, NORMAL and standby. Standby mode further consists of two modes, HALT and STOP. In standby mode, power consumption can be reduced.

* *The initial values of the OSCM1 and OSCM0 bits depend on the mask option.*

[☞ "2-9-4 Switching Oscillation Mode"]

2-9-2 CPU Mode Register (CPUM)

■ CPUM

This register specifies the CPU's operating mode.

CPUM (After reset: - 0 - - * * 0 0)

7	6	5	4	3	2	1	0
—	CLKSEL	—	—	OSCM1	OSCM0	HALTM	STOPM

STOPM	STOP mode
0	NOP
1	All oscillation stopped

HALTM	HALT mode
0	NOP
1	HALT mode

OSCM1	OSCM0	Oscillation control		
			fosc	fx
0	0	Normal	○	△
	1	IDLE	△	○
1	0	Cannot be used		
	1	SLOW	×	○

○: Used for system clock

△: Oscillation is performed but not used for system control

×: Oscillation stopped

CLKSEL	Clock select
0	System clock is supplied to peripheral circuits
1	1/2 system clock is supplied to peripheral circuits

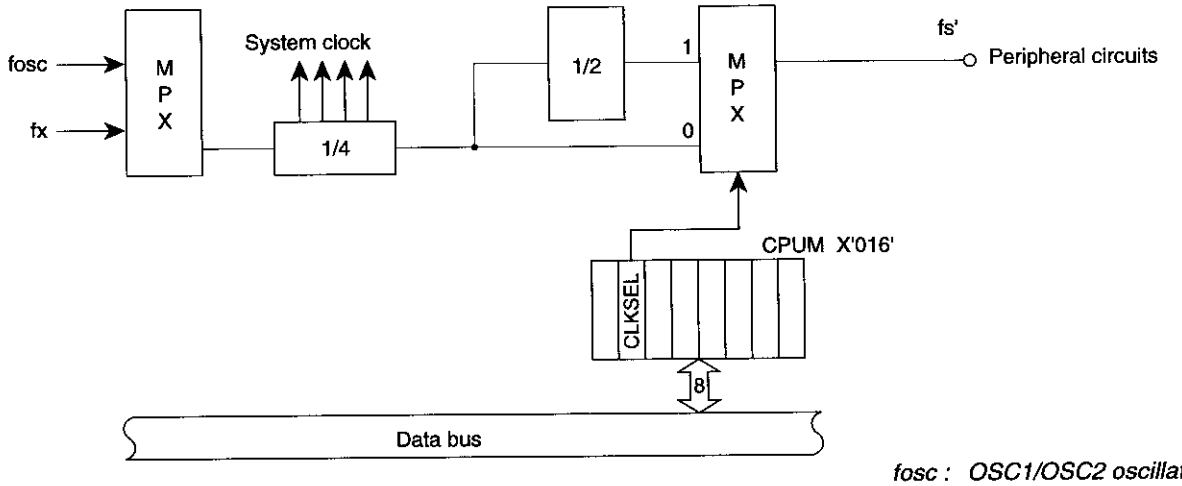
Figure 2-9-1 CPU Mode Register (CPUM: X'016' R/W)

In a reset, the STOPM bit, HALTM bit, and CLKSEL bit are cleared to 0, and the OSCM1 and OSCM0 bits are set to 00 or 11 depending on the mask option.

The STOPM bit and HALTM bit flags are cleared by an interrupt request.

When the CLKSEL bit is set to 1, 1/2 system clock frequency is supplied to the peripheral circuits.

Here, fs is 1/4fosc or 1/4fx when the CLKSEL bit is 0, and 1/8fosc or 1/8fx when the CLKSEL bit is 0. [☞ Figure 2-9-1]



fosc : OSC1/OSC2 oscillation frequency

fx : XI/XO oscillation frequency

fs : System clock

Figure 2-9-2 System Clock Generation Block

2-9-3 STOP and HALT Modes

In addition to the three operating modes shown in figure 2-9-4, the MN187XX23 has two standby modes, STOP and HALT, for reducing power consumption.

In STOP mode both fosc and fx oscillation stops, and the CPU halts with its internal state maintained. Recovery from STOP mode is by means of an interrupt. In this case the interrupt enable flag (IE) for the interrupt used for recovery must be set to the enable state beforehand. Recovery from STOP mode includes an oscillation stabilization wait. [☞ Figure 2-9-4]

The STOP and HALT instructions must be used for STOP and HALT mode control; CPU mode register (CPUM) bits cannot be manipulated directly. At this time, XPh and DF should both be cleared to 0.

CPU mode register
(CPUM: X'016' R/W)
Bit 3: OSCM1
Bit 2: OSCM0

Table 2-9-1 shows the state of the MN187XX23 in STOP and HALT modes. In both modes, the serial interface and timers can operate. Both modes can be exited by an IRQ0, IRQ1, IRQ2, or IRQ3 external interrupt, a reset, a serial interface or a timer.

HALT mode is one of three kinds—HALT0, HALT1, or HALT2—according to the value of the OSCM1 and OSCM0 bits in the CPU mode register (CPUM). In all three modes, the CPU's internal state is maintained. Recovery from HALT mode is by means of an interrupt in the same way as for STOP mode. There is no oscillation stabilization wait in recovery from HALT mode.

Table 2-9-1 States in STOP and HALT Modes

Mode		HALT0 OSCM1,0="00" HALTM="1"	HALT1 OSCM1,0="01" HALTM="1"	HALT2 OSCM1,0="11" HALTM="1"	STOP STOPM="1"
OSC1/OSC2		Active	Active	Stopped	Stopped
XI/XO		Active	Active	Active	Stopped
CPU		×	×	×	×
RAM		Retained	Retained	Retained	Retained
Ports		Retained	Retained	Retained	Retained
Interrupts/special functions	IRQ0	○	○	○	○
	IRQ1	○*1	○*1	○*1	○*1
	IRQ2	○*1	○*1	○*1	○*1
	IRQ3	○	○	○	○
	Timer 0	○	○	○	△*2
	Timer 1	○	○	○	×
	Timer 2	○	○	○	×
	Timer 3	○	○	○	△
	Timer 4	○	○	○	△*2
	Timer 5	○	○	○	×
	Serial interface 0	○	○	○	△
	Serial interface 1	○	○	○	△
	Serial interface 2	○	○	○	△
	FLP	×	×	×	×
	A/D converter	×	×	×	×
	Automatic data transfer	×	×	×	×

○: Active

×: Stopped

△: Operates only in case of external clock or event counter.

*1: Does not operate when a noise filter is used.

*2: Does not operate when synchronous mode is used, even with event counting.

2-9-4 Switching Oscillation Mode

The MN187XX23 has two on-chip clock pulse generators, for fosc and fx. The system clock is derived by dividing the fosc or fx clock by 4. Switching of the oscillation mode, stopping oscillation, and HALT mode control are performed using the STOPM, HALTM, OSCM1, and OSCM0 bits in the CPU mode register (CPUM).

Figure 2-9-4 shows the transition state diagram for system clock switchover.

In the MN187XX23, the initial value of the OSCM1 and OSCM0 bits after a reset can be changed by means of a mask option (type A or B). [☞ Figure 2-9-4]

CPU mode register
(CPUM: X'016' R/W)
Bit 3: OSCM1
Bit 2: OSCM0
Bit 1: HALTM
Bit 0: STOPM

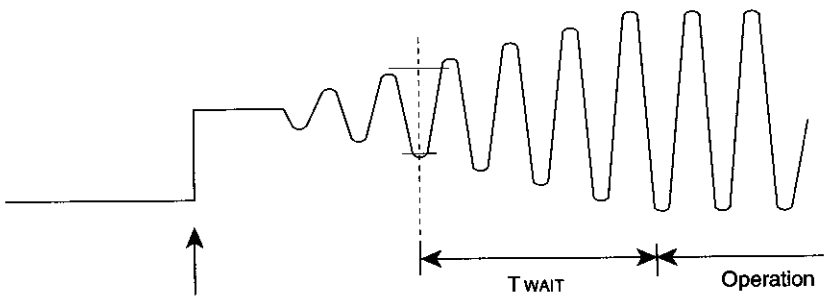
When type A is selected, the OSCM1 and OSCM0 bits in the CPU mode register (CPUM) are both cleared to 0, and operation starts from a reset cycle in NORMAL mode.

When type B is selected, the OSCM1 and OSCM0 bits in the CPU mode register (CPUM) are both set to 1, and operation starts from a reset cycle in SLOW mode.

In either case, when the system clock is switched from fx to fosc, or from fosc to fx, the switch to NORMAL or SLOW mode should be after allowing for the oscillation stabilization wait time in IDLE mode.

The guideline wait time is 1/fx in a transition from NORMAL to SLOW mode, and the time required for the fosc to stabilize in a transition from SLOW to NORMAL mode. [☞ Figure 2-9-3]

In a reset, and when recovering from STOP mode, a hardware wait is generated automatically until clock oscillation stabilizes.
The wait time is a count of 2^{13} clock pulses after clock generation starts and its amplitude reaches the level required to drive a counter.



$T_{WAIT}=2^{13}\times Ts$ Ts=1/fs=system clock cycle

=1.0 s (32kHz) / 7.8ms (4.19MHz) / 3.9ms (8.38MHz)

Figure 2-9-3 Wait at Start of Oscillation

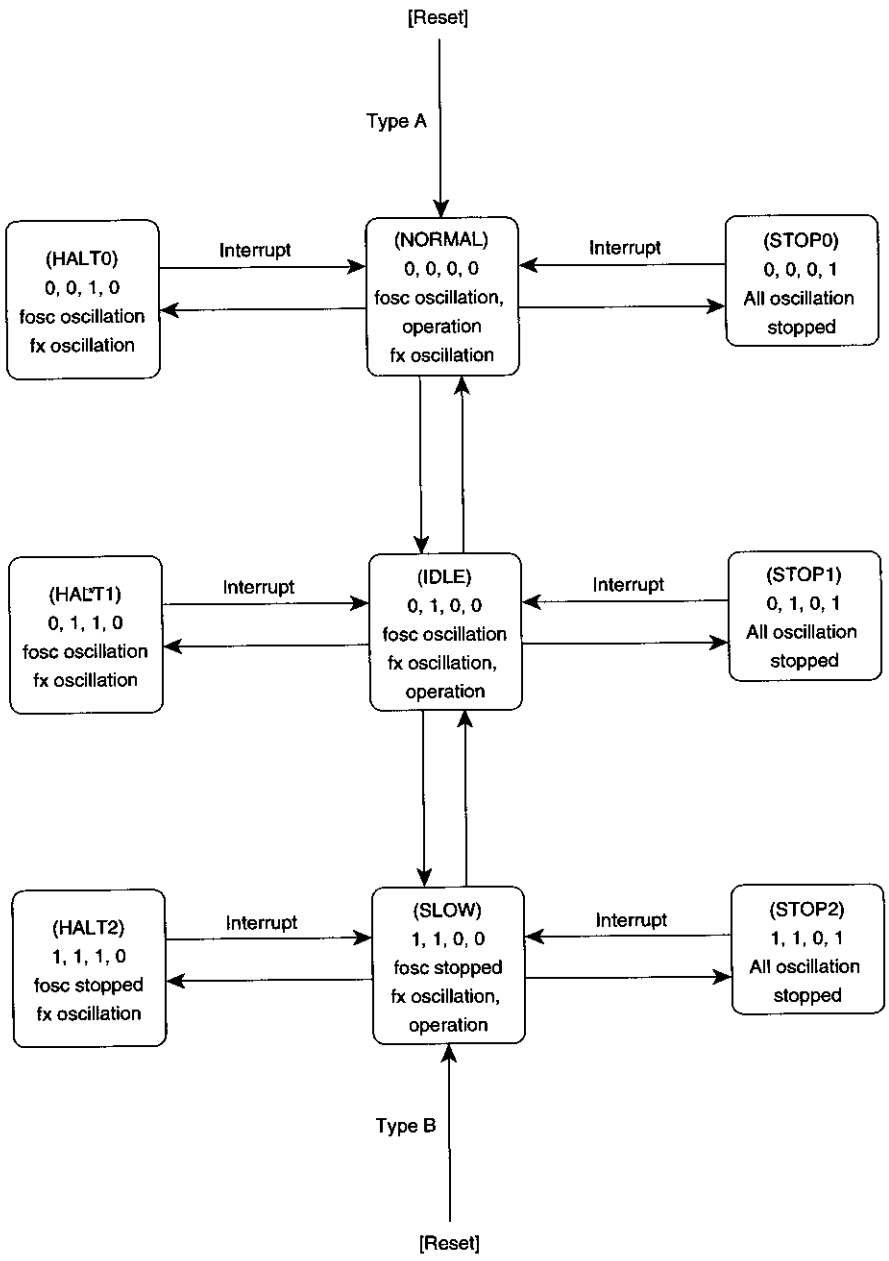


Figure 2-9-4 System Clock State Transition Diagram

Either type A or type B can be selected by means of a mask option. In a reset, an oscillation stabilization wait is provided by hardware.

"0, 1, 1, 0" in the figure on the left indicates the value of the following CPU mode register (CPUM) bits in left-to-right order: OSCM1=0, OSCM0=1, HALTM=1, STOPM=0.

2-10 Reset

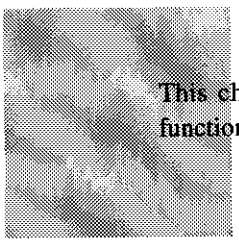
The MN187XX23 carries out the following processing in a reset cycle.

1. Clears the carry flag (CF), zero flag (ZF), and direct flag (DF).
2. Clears the data pointers (XP, YP).
3. Sets X'100' to the stack pointer (SP).
4. Clears the flags in the special registers (see the descriptions of the individual registers).
5. Sets the reset vector address to the instruction pointer (IP).

The reset pin is an open-drain input/output pin, and reset cycle operations are performed by means of an instruction that causes low-level output to P07. Input to the reset pin becomes valid on a low-level edge. The number of clock pulses equivalent to the oscillation stabilization wait time ($2^{13} \times 1/fs$) are counted after the reset signal goes high, then the internal reset is released.

Chapter 3 I/O Port Functions

3



This chapter describes the MN187XX23's I/O port functions.

3-1 Overview

The MN187XX23 has a total of 73 I/O port pins in 10 ports, P0 to P9, including ports shared with special-function pins.
Each I/O port is assigned to RAM address space, and I/O port operations can be performed in byte, nibble, and bit units in the same way as RAM operations.

Each I/O port is assigned to an address with the same number as the port: thus P0 is assigned to RAM address X'000', P1 to RAM address X'001', and so on.

Input/output control is performed by port 0~4 and 9 direction control registers (P0DIR~P4DIR, P9DIR). A "1" setting specifies output mode and a "0" setting specifies input mode for the corresponding port pin.
Switching of pins shared with I/O port and special-function pins is performed by means of special-function register settings. Input/output direction control by the port 0~4 and 9 direction control registers is also valid when special function use is selected, and therefore these settings must make provision for the input/output direction of the special-function pins.
In a reset, the port 0~4 and 9 direction control registers are cleared to 0, and the port/special-function switching bits are set to port mode, so that all I/O port (special-function-shared) pins are initialized as input ports. All port output latches are also cleared to 0.

Port 0 direction control register (P0DIR: X'018' R/W)

Port 1 direction control register (P1DIR: X'019' R/W)

Port 2 direction control register (P2DIR: X'01A' R/W)

Port 3 direction control register (P3DIR: X'01B' R/W)

Port 4 direction control register (P4DIR: X'01C' R/W)

Port 9 direction control register (P9DIR: X'01D' R/W)

1. *The special-function pin corresponding to each port pin is shown in parentheses.*
2. *In the I/O column, I/O indicates dual input/output, O output-only, and I input-only.*

Port 0 direction control register (P0DIR: X'018' R/W)

Serial interface mode register 0 (SIM0: X'022' R/W)
Bit 5: SBTC0
Bit 4: SBOC0

Serial interface mode register 1 (SIM1: X'023' R/W)
Bit 5: SBTC1
Bit 4: SBOC1

Port 0 pull-up control register (P0RON: X'040' R/W)

Port 1 direction control register (P1DIR: X'019' R/W)

Port 1 pull-up control register (P1RON: X'041' R/W)

Port 2 direction control register (P2DIR: X'01A' R/W)

Port 2 pull-up control register (P2RON: X'042' R/W)

Table 3-1-1 I/O Ports (1/3)

Port Name	Pin Name	I/O	Description
Port 0 (P0)	P00 ($\overline{SBO0}$)	I/O	8-bit general-purpose input/output port. P00 to P06 are CMOS 3-state input/output ports, and P07 is an N-channel open-drain input/output port. Ports P00 to P06 can be set as input or output bit-wise by the port 0 direction control register (P0DIR). [Pins P00 ($\overline{SBO0}$), P02 (SBT0), P03 ($\overline{SBO1}$), and P05 (SBT1) can be set as N-channel open-drain input/output ports by bits SBOC0, SBTC0, SBOC1, and SBTC1, respectively, in the serial interface mode registers (SIM0, SIM1).] P00 to P06 are also controlled by the port 0 direction control register (P0DIR) when their special-function mode is selected. P07 is shared with the reset pin, and the LSI is reset by a low-level input or low-level output. Use of a pull-up resistor can be selected for ports P00, P01, P03, P04, and P06 with the port 0 pull-up control register (P0RON). [☞ "5-4-4 Serial Interface Mode Registers (SIM)," "3-2-3 Direction Control Registers (DIR)"]
	P01 ($\overline{SBI0}$)	I/O	
	P02 (SBT0)	I/O	
	P03 ($\overline{SBO1}$)	I/O	
	P04 ($\overline{SBI1}$)	I/O	
	P05 (SBT1)	I/O	
	P06 (PSBT0)	I/O	
	P07 ($\overline{RST}$)	I/O	
Port1 (P1)	P10 (TCIO0)	I/O	8-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 1 direction control register (P1DIR). Individual P1 pins are also controlled by the port 1 direction control register (P1DIR) when their special-function mode is selected. Use of a pull-up resistor can be selected for each port 1 pin with the port 1 pull-up control register (P1RON). [☞ "4-10-3 Timer Mode Registers (TM)," "7-5-2 PWM Control Register (PWMC)," "3-2-3 Direction Control Registers (DIR)"]
	P11 (TCO1)	I/O	
	P12 (TCI3)	I/O	
	P13 (TCIO4)	I/O	
	P14 (TCO35)	I/O	
	P15 (RMOUT)	I/O	
	P16 (PWM1)	I/O	
	P17 (PWM2)	I/O	
Port2 (P2)	P20 (ADIN0)	I/O	8-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 2 direction control register (P2DIR). Individual P2 pins are also controlled by the port 2 direction control register (P2DIR) when their special-function mode is selected. Use of a pull-up resistor can be selected for each port 2 pin with the port 2 pull-up control register (P2RON). [☞ "8-3-2 A/D Control Register (ADC)," "3-2-3 Direction Control Registers (DIR)"]
	P21 (ADIN1)	I/O	
	P22 (ADIN2)	I/O	
	P23 (ADIN3)	I/O	
	P24 (ADIN4)	I/O	
	P25 (ADIN5)	I/O	
	P26 (ADIN6)	I/O	
	P27 (ADIN7)	I/O	

Table 3-1-1 I/O Ports (2/3)

Port Name	Pin Name	I/O	Description
Port 3 (P3)	P30 (IRQ0) P31 (IRQ1) P32 (IRQ2) P33 (IRQ3) P34 (PWM0)	I/O I/O I/O I/O I/O	5-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 3 direction control register (P3DIR). P3 pins are also controlled by the port 3 direction control register (P3DIR) when their special-function mode is selected. Use of a pull-up resistor can be selected for each port 3 pin with the port 3 pull-up control register (P3RON). [☞ "7-5-2 PWM Control Register (PWMC)," "3-2-3 Direction Control Registers (DIR)"]
Port 4 (P4)	P40 P41 P42 P43 P44 P45 P46 P47	I/O I/O I/O I/O I/O I/O I/O I/O	8-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 4 direction control register (P4DIR). Use of a pull-down resistor between the pin and the VSS pin can be selected for each port 4 pin with the port 4 pull-down control register (P4RON).
Port 5 (P5)	P50 (SEG0) P51 (SEG1) P52 (SEG2) P53 (SEG3) P54 (SEG4) P55 (SEG5) P56 (SEG6) P57 (SEG7)	O O O O O O O O	8-bit high-voltage break down output port. Output structure is P-channel open-drain type. Port 5 pins incorporate a pull-down resistor between the pin and the VPP pin. When the FLP display circuit is operating, these pins can be used as segment output pins. [☞ "6-3-2 (3) Port/FLP Select Register 2 (PFSR2)"]
Port 6 (P6)	P60 (SEG8) P61 (SEG9) P62 (SEG10) P63 (SEG11) P64 (SEG12) P65 (SEG13) P66 (SEG14) P67 (SEG15)	O O O O O O O O	8-bit high-voltage break down output port. Output structure is P-channel open-drain type. Use of a pull-down resistor between the pin and the VPP pin can be selected for each port 6 pin with a mask option. When the FLP display circuit is operating, these pins can be used as segment output pins. [☞ "6-3-2 (1) Port/FLP Select Register 0 (PFSR0)"]

Port 3 direction control register (P3DIR: X'01B' R/W)

Port 3 pull-up control register (P3RON: X'043' R/W)

Port 4 direction control register (P4DIR: X'01C' R/W)

Port 4 pull-down control register (P4RON: X'044' R/W)

Table 3-1-1 I/O Ports (3/3)

Port Name	Pin Name	I/O	Description
Port 7 (P7)	P70 (DGT7)	O	8-bit high-voltage break down output port. Output structure is P-channel open-drain type. Port 7 pins incorporate a pull-down resistor between the pin and the VPP pin. When the FLP display circuit is operating, these pins can be used as digit output pins. [☞ "6-3-2 (3) Port/FLP Select Register 2 (PFSR2)"]
	P71 (DGT6)	O	
	P72 (DGT5)	O	
	P73 (DGT4)	O	
	P74 (DGT3)	O	
	P75 (DGT2)	O	
	P76 (DGT1)	O	
	P77 (DGT0)	O	
Port 8 (P8)	P80 (DGT15)	O	8-bit high-voltage break down output port. Output structure is P-channel open-drain type. Use of a pull-down resistor between the pin and the VPP pin can be selected for each port 8 pin with a mask option. When the FLP display circuit is operating, these pins can be used as digit output pins. [☞ "6-3-2 (2) Port/FLP Select Register 1 (PFSR1)"]
	P81 (DGT14)	O	
	P82 (DGT13)	O	
	P83 (DGT12)	O	
	P84 (DGT11)	O	
	P85 (DGT10)	O	
	P86 (DGT9)	O	
	P87 (DGT8)	O	
Port 9 (P9)	P90 (BUZZER)	I/O	4-bit CMOS 3-state input/output port. Input or output can be set bit-wise by the port 9 direction control register (P9DIR). Pins P91 ($\overline{SBO2}$) and P93 (SBT2) can be set as N-channel open-drain input/output ports by bits SBOC2 and SBTC2, respectively, in the serial interface mode registers (SIM2). Individual port 9 pins are also controlled by the port 9 direction control register (P9DIR) when their special-function mode is selected. Use of a pull-up resistor can be selected for P91 and P92 with the port 9 pull-up control register (P9RON). [☞ "4-10-3 Timer Mode Registers (TM)," "3-2-3 Direction Control Registers (DIR)"]
	P91 ($\overline{SBO2}$)	I/O	
	P92 ($\overline{SBI2}$)	I/O	
	P93 (SBT2)	I/O	

Port 9 direction control register (P9DIR: X'01D' R/W)

Serial interface mode register 2 (SIM2: X'036' R/W)
Bit 5: SBTC2
Bit 4: SBOC2

Port 9 pull-up control register (P9RON: X'045' R/W)

3-2 Port Control Registers

3-2-1 Overview

Twenty-two register are used to control the MN187XX23's I/O ports: ports (PORT0~PORT9), port direction control registers (P0DIR~P4DIR, P9DIR) which control the input/output direction, port pull-up control registers (P0RON~P3RON, P9RON), and a port pull-down control register (P4RON). These registers are all located in RAM space.

Table 3-2-1 I/O Port Control Registers

Register Abbreviation	RAM Address	R/W	Register Name
PORT0	X'000'	R/W	Port 0
PORT1	X'001'	R/W	Port 1
PORT2	X'002'	R/W	Port 2
PORT3	X'003'	R/W	Port 3
PORT4	X'004'	R/W	Port 4
PORT5	X'005'	W	Port 5
PORT6	X'006'	W	Port 6
PORT7	X'007'	W	Port 7
PORT8	X'008'	W	Port 8
PORT9	X'009'	R/W	Port 9
P0DIR	X'018'	R/W	Port 0 direction control register
P1DIR	X'019'	R/W	Port 1 direction control register
P2DIR	X'01A'	R/W	Port 2 direction control register
P3DIR	X'01B'	R/W	Port 3 direction control register
P4DIR	X'01C'	R/W	Port 4 direction control register
P9DIR	X'01D'	R/W	Port 9 direction control register
P0RON	X'040'	R/W	Port 0 pull-up control register
P1RON	X'041'	R/W	Port 1 pull-up control register
P2RON	X'042'	R/W	Port 2 pull-up control register
P3RON	X'043'	R/W	Port 3 pull-up control register
P4RON	X'044'	R/W	Port 4 pull-up control register
P9RON	X'045'	R/W	Port 9 pull-up control register

	7	6	5	4	3	2	1	0	
PORT0	P07	P06	P05	P04	P03	P02	P01	P00	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT1	P17	P16	P15	P14	P13	P12	P11	P10	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT2	P27	P26	P25	P24	P23	P22	P21	P20	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT3	—	—	—	P34	P33	P32	P31	P30	(After reset: - - - 0 0 0 0 0 Output latch)
PORT4	P47	P46	P45	P44	P43	P42	P41	P40	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT5	P57	P56	P55	P54	P53	P52	P51	P50	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT6	P67	P66	P65	P64	P63	P62	P61	P60	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT7	P77	P76	P75	P74	P73	P72	P71	P70	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT8	P87	P86	P85	P84	P83	P82	P81	P80	(After reset: 0 0 0 0 0 0 0 0 Output latch)
PORT9	—	—	—	—	P93	P92	P91	P90	(After reset: - - - - 0 0 0 0 Output latch)
P0DIR	—	P06ON	P05ON	P04ON	P03ON	P02ON	P01ON	P00ON	(After reset: - 0 0 0 0 0 0 0)
P1DIR	P17ON	P16ON	P15ON	P14ON	P13ON	P12ON	P11ON	P10ON	(After reset: 0 0 0 0 0 0 0 0)
P2DIR	P27ON	P26ON	P25ON	P24ON	P23ON	P22ON	P21ON	P20ON	(After reset: 0 0 0 0 0 0 0 0)
P3DIR	—	—	—	P34ON	P33ON	P32ON	P31ON	P30ON	(After reset: - - - 0 0 0 0 0)
P4DIR	P47ON	P46ON	P45ON	P44ON	P43ON	P42ON	P41ON	P40ON	(After reset: 0 0 0 0 0 0 0 0)
P9DIR	—	—	—	—	P93ON	P92ON	P91ON	P90ON	(After reset: - - - - 0 0 0 0)
P0RON	—	P06RON	—	P04RON	P03RON	—	P01RON	P00RON	(After reset: - 0 - 0 0 - 0 0)
P1RON	P17RON	P16RON	P15RON	P14RON	P13RON	P12RON	P11RON	P10RON	(After reset: 0 0 0 0 0 0 0 0)
P2RON	—	—	—	—	—	—	P2HRON	P2LRON	(After reset: - - - - - - 0 0)
P3RON	—	—	—	—	P33RON	P32RON	P31RON	P30RON	(After reset: - - - - 0 0 0 0)
P4RON	P47RON	P46RON	P45RON	P44RON	P43RON	P42RON	P41RON	P40RON	(After reset: 0 0 0 0 0 0 0 0)
P9RON	—	—	—	—	—	P92RON	P91RON	—	(After reset: - - - - - 0 0 -)

— indicates an uninstalled bit. These bits cannot be written, and always return 1 if read.

Figure 3-2-1 Port Control Registers

3-2-2 Port Read/Write Operations

When data is written to a port address, the contents are written into the corresponding port output latch.

When a port is read, in the case of an input/output port, whether the output latch or pin value is read depends on the instruction. For example, with the SET and CLR instructions, the output latch value is read. This function enables any output latch bit manipulation to be performed without regard to the pin value.

For details of which value, output latch or pin, is read by which instructions, see the "in" section in the "Instruction List" and the "MN1870 Series Instruction Manual."

3-2-3 Direction Control Registers

The port 0~4 and 9 direction control registers (P0DIR~P4DIR, P9DIR) control the input/output direction of input/output ports bit by bit. A "0" setting specifies input mode for a port, and a "1" setting specifies output mode.

In a reset, all port output latches are cleared to 0, and the direction control registers are also cleared to 0, so that I/O ports are set to input mode (output high-impedance). [☞ Figure 3-2-1]

Port 0 direction control register (P0DIR: X'018' R/W)

Port 1 direction control register (P1DIR: X'019' R/W)

Port 2 direction control register (P2DIR: X'01A' R/W)

Port 3 direction control register (P3DIR: X'01B' R/W)

Port 4 direction control register (P4DIR: X'01C' R/W)

Port 9 direction control register (P9DIR: X'01D' R/W)

3-2-4 Pull-Up Control Registers and Pull-Down Control Register

Port 0 pins P00, P01, P03, P04, and P06, port 1 pins P10~P17, port 2 pins P20~P27, port 3 pins P30~P33, and port 9 pins P91 and P92 incorporate a program-controllable pull-up resistor between the pin and VDD. Port 4 pins P40~P47 incorporate a program-controllable pull-down resistor between the pin and VSS.

The pull-up resistors and pull-down resistors are turned on and off by means of the pull-up control registers (P0RON, P1RON, P2RON, P3RON, P9RON) and the pull-down control register (P4RON). A "0" setting turns the corresponding port's pull-up resistor or pull-down resistor off, and a "1" setting turns it on.

The on/off status of the pull-up resistors or pull-down resistors of ports 0, 1, 3, 4, and 9 is controlled bit by bit. For port 2, pull-up resistor on/off control is performed in 4-bit units.

In a reset, all pull-up control register and pull-down control register bits are cleared to 0, so that all pull-up resistors and pull-down resistors are off after a reset.

Port 0 pull-up control register (P0RON: X'040' R/W)

Port 1 pull-up control register (P1RON: X'041' R/W)

Port 2 pull-up control register (P2RON: X'042' R/W)

Port 3 pull-up control register (P3RON: X'043' R/W)

Port 4 pull-down control register (P4RON: X'044' R/W)

Port 9 pull-up control register (P9RON: X'045' R/W)

When an I/O port is read, in the case of an input/output port, whether the output latch or pin value is read depends on the instruction. That is, the TERM signal in table 3-3-1 is either 0 or 1 depending on the instruction

3-3 I/O Port Structure and Functions

Table 3-3-1 I/O Port Structures and Functions (1/7)

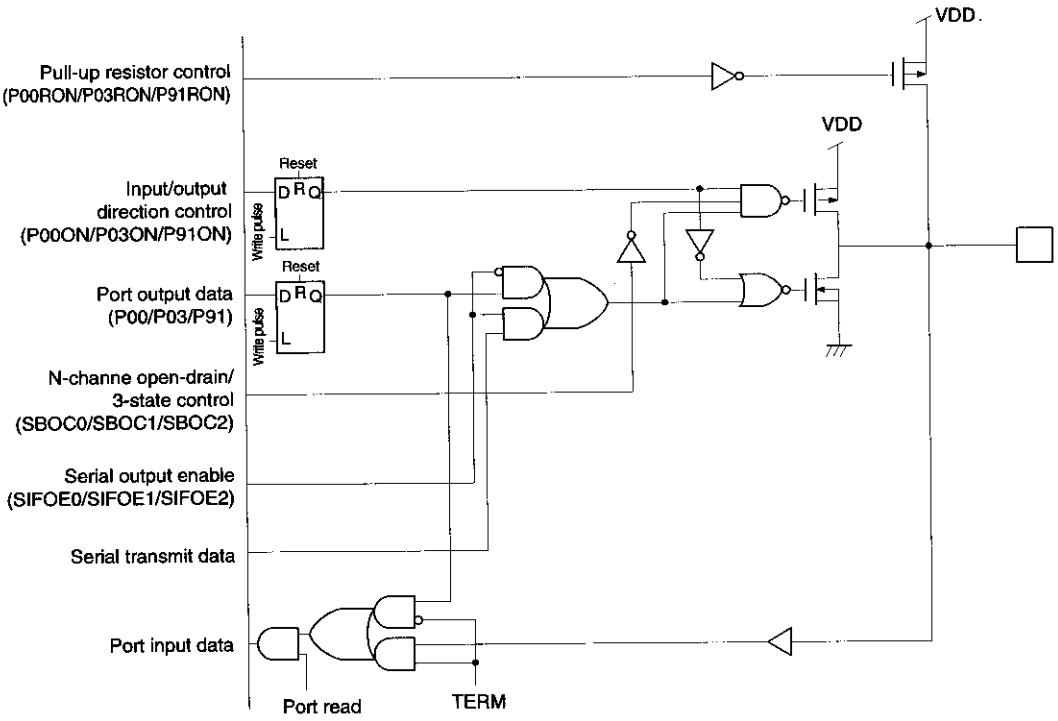
Pin Names	Structure
P00 ($\overline{SBO0}$) P03 ($\overline{SBO1}$) P91 ($\overline{SBO2}$)	 <Legend> SBOC0/SBOC1/SBOC2: Bit 4 of serial interface mode register 0/1/2 (SIM0/SIM1/SIM2) SIFOE0/SIFOE1/SIFOE2: Bit 7 of serial interface mode register 0/1/2 (SIM0/SIM1/SIM2)

Table 3-3-1 I/O Port Structures and Functions (2/7)

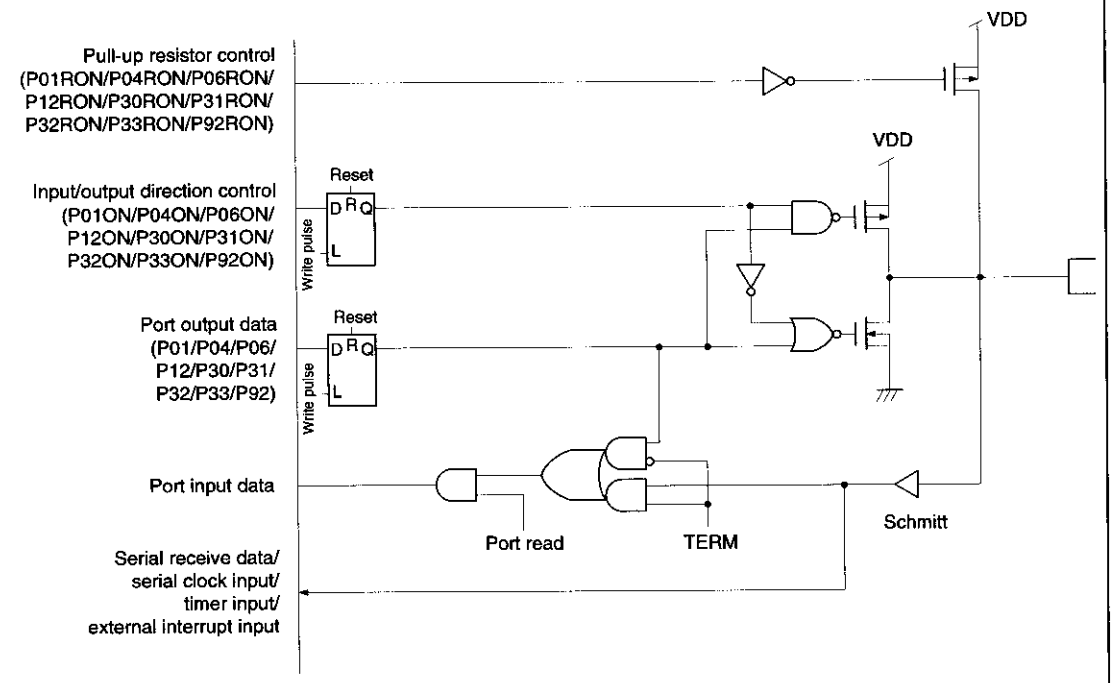
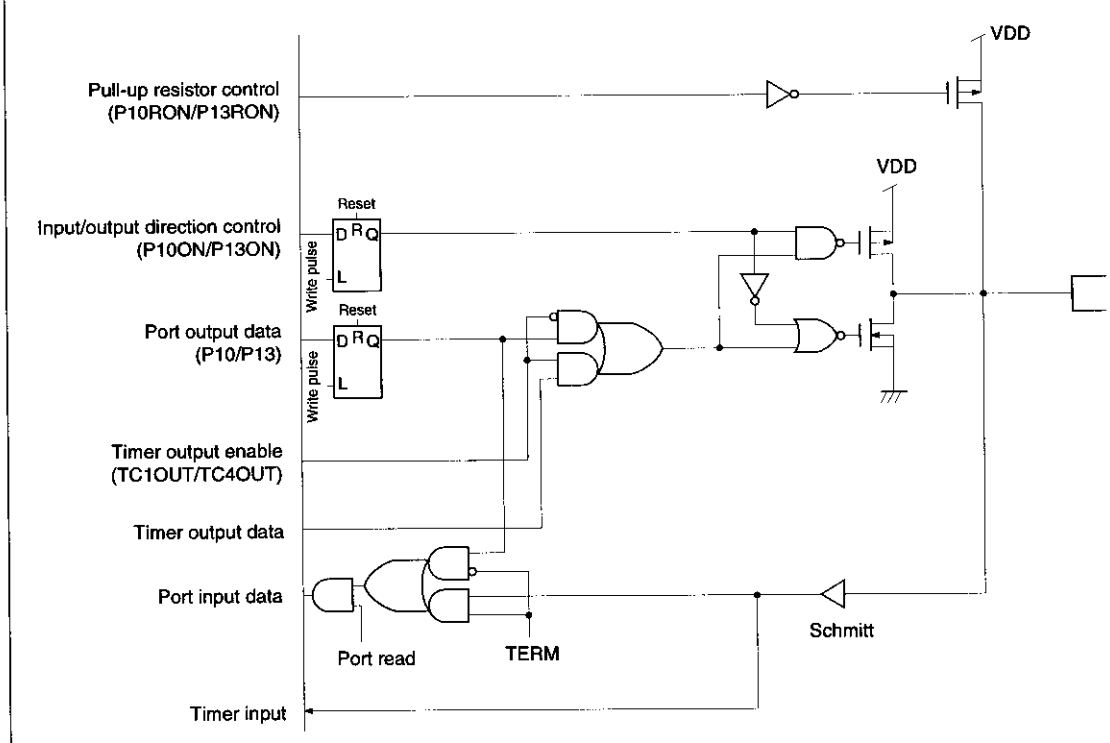
Pin Names	Structure
P01 ($\overline{SBI0}$) P04 ($\overline{SBI1}$) P06 (PSBT0) P12 (TCI3) P30 (IRQ0) P31 (IRQ1) P32 (IRQ2) P33 (IRQ3) P92 ($\overline{SBI2}$)	
P10 (TCIO0) P13 (TCIO4)	 <Legend> TC0OUT: Bit 5 of timer mode register 0 (TM0) TC4OUT: Bit 3 of timer mode register 4 (TM4)

Table 3-3-1 I/O Port Structures and Functions (3/7)

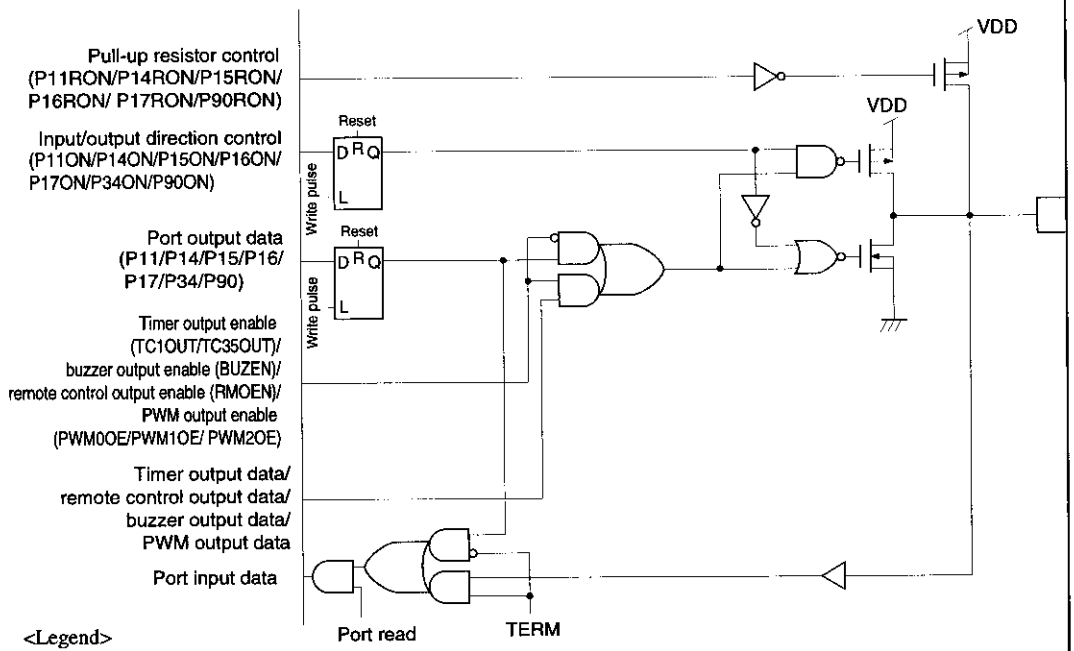
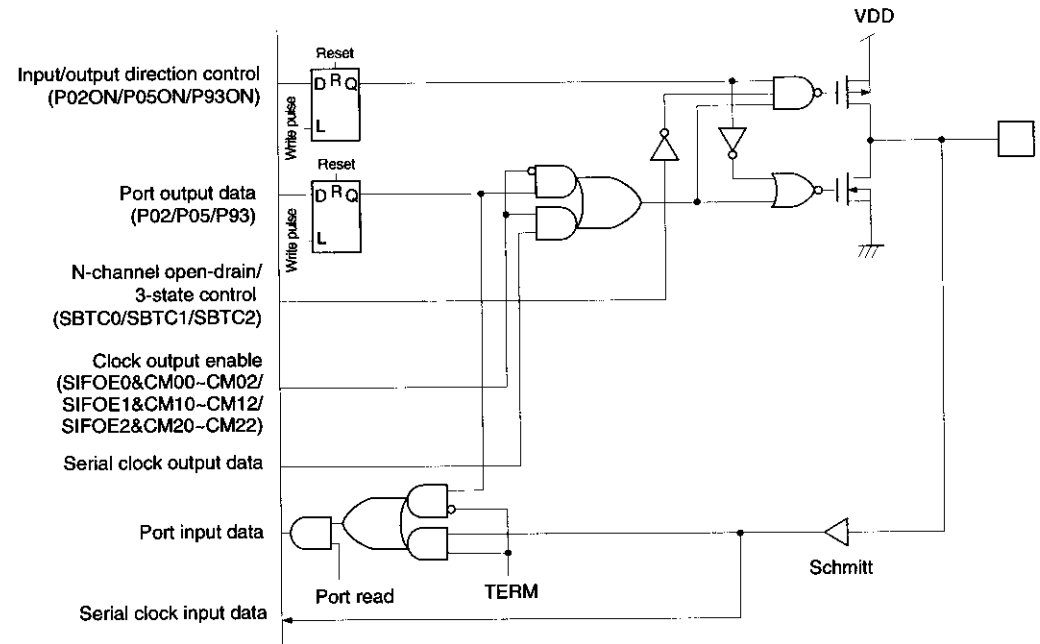
Pin Names	Structure
P11 (TCO1) P14 (TCO35) P15 (RMOUT) P16 (PWM1) P17 (PWM2) P34 (PWM0) P90 (BUZZER)	 <Legend> TC1OUT: Bit 3 of timer mode register 1 (TM1) TC35OUT: Bit 6 of timer mode register 5 (TM5) RMOEN: Bit 4 of remote control control register (RMC) BUZEN: Bit 5 of timer mode register 2 (TM2) PWM0OE: Bit 0 of PWM control register (PWMC) PWM1OE: Bit 1 of PWM control register (PWMC) PWM2OE: Bit 2 of PWM control register (PWMC)
P02 (SBT0) P05 (SBT1) P93 (SBT2)	 <Legend> SIFOE0 & CM00~CM02: Bits 7, 0~2 of serial interface mode register 0 (SIM0) SIFOE1 & CM10~CM12: Bits 7, 0~2 of serial interface mode register 1 (SIM1) SIFOE2 & CM20~CM22: Bits 7, 0~2 of serial interface mode register 2 (SIM2)

Table 3-3-1 I/O Port Structures and Functions (4/7)

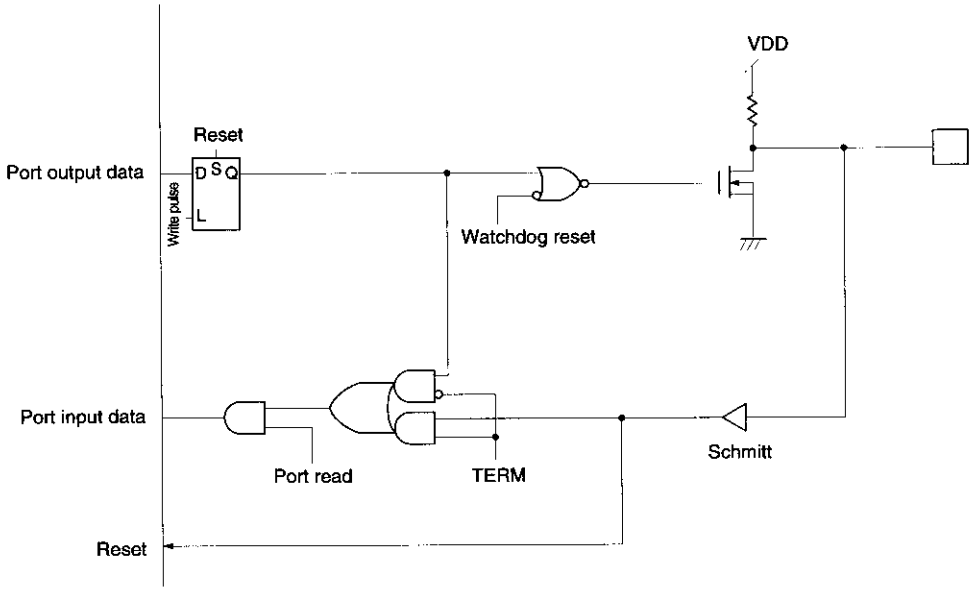
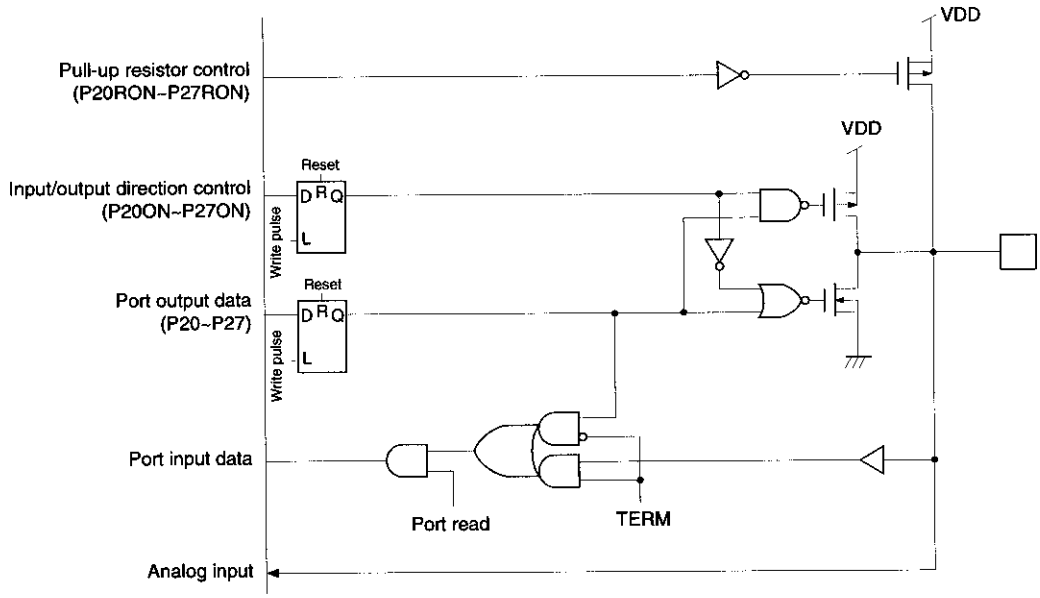
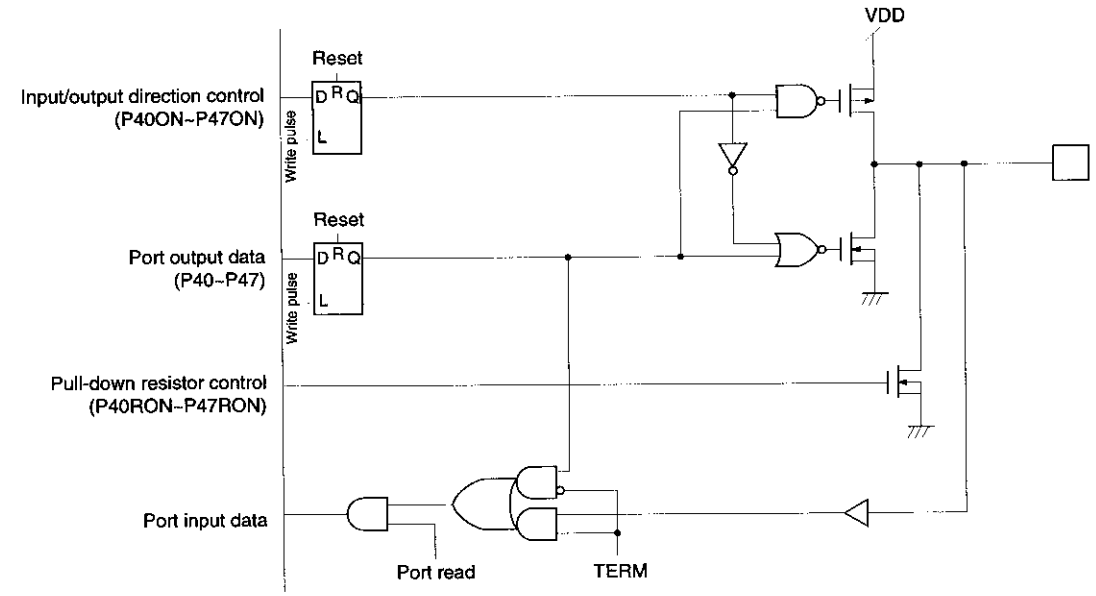
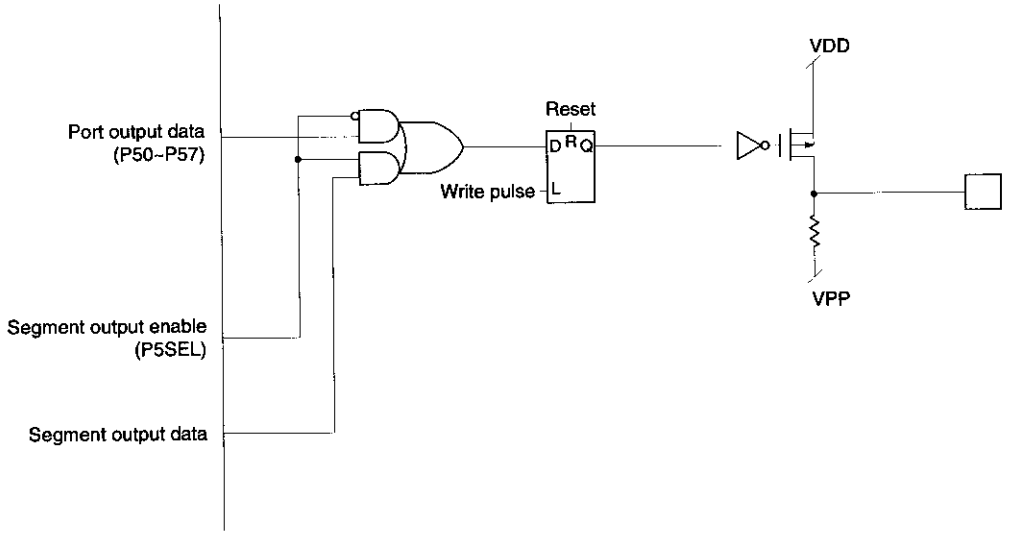
Pin Names	Structure
P07 ($\overline{RST}$)	
P20 (ADIN0) ~ P27 (ADIN7)	

Table 3-3-1 I/O Port Structures and Functions (5/7)

Pin Names	Structure
P40~P47	
P50 (SEG0) P57 (SEG7)	

<Legend>

P5SEL: Bit 0 of port/FLP select register 2 (PFSR2)

Table 3-3-1 I/O Port Structures and Functions (6/7)

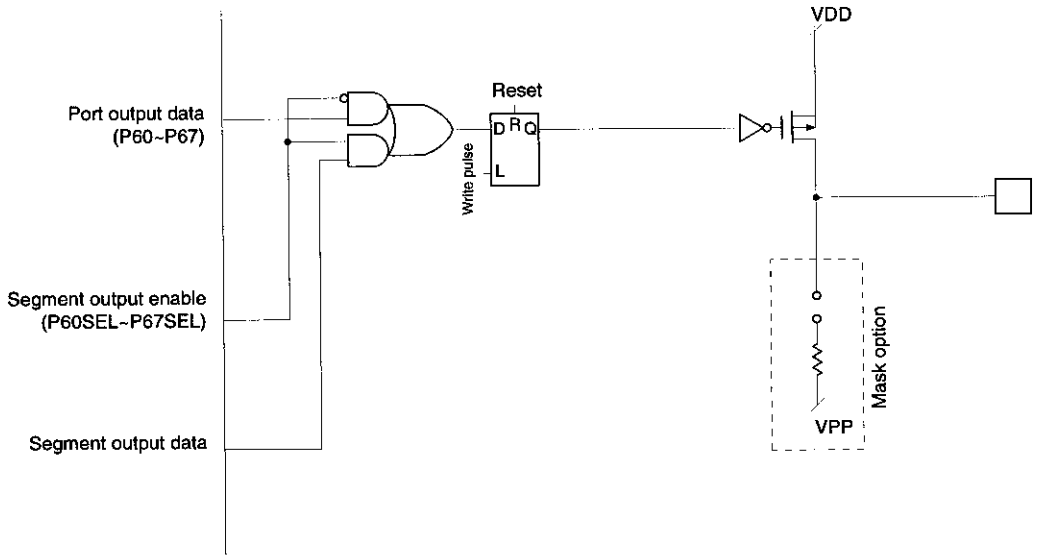
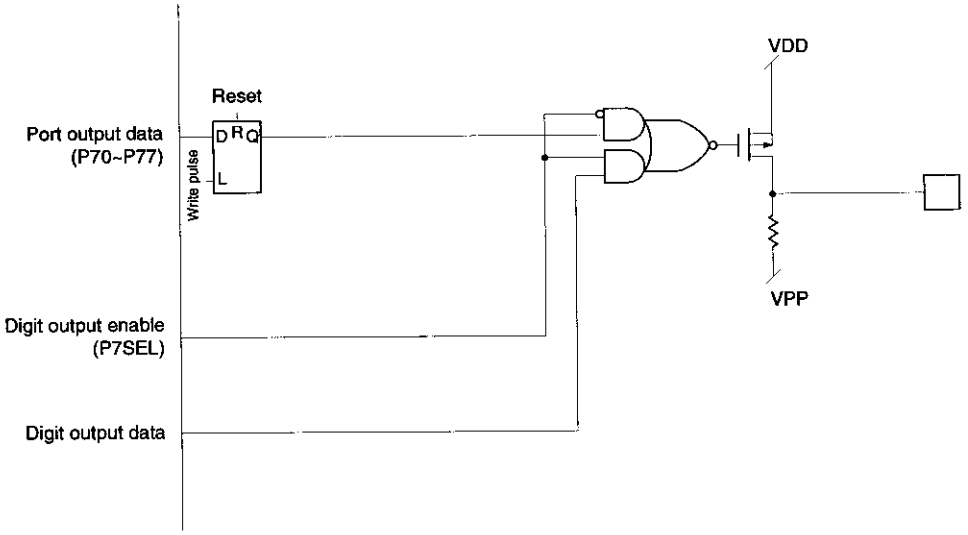
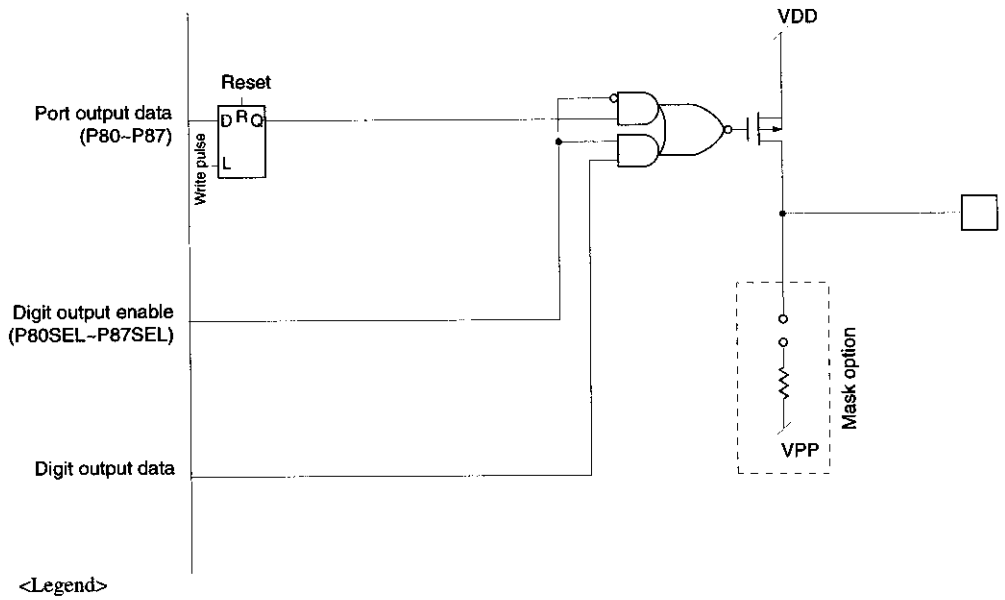
Pin Names	Structure
P60 (SEG8) P67 (SEG15)	 <Legend> P60SEL~P67SEL: Bits 0~7 of port/FLP select register 0 (PFSR0)
P70 (DGT7) P77 (DGT0)	 <Legend> P7SEL: Bit 1 of port/FLP select register 2 (PFSR2)

Table 3-3-1 I/O Port Structures and Functions (7/7)

Pin Names	Structure
P80 (DGT15) P87 (DGT8)	 <Legend> P80SEL~P87SEL: Bits 0~7 of port/FLP select register 1 (PFSR1)

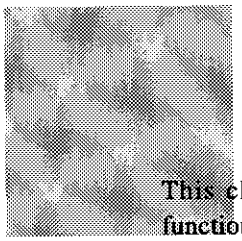
3-4 Handling of Unused Pins

Unused pins should be handled as shown below.

Input/output pins: Pins whose input/output direction can be set with the port 0~4 and 9 direction control registers should be set to input mode and pulled up or down by inserting a resistor of several tens of kΩ.

In the case of pins for which a pull-up or pull-down resistor can be selected with the pull-up control register or pull-down control register, unused pins can also be handled by using a pull-up or pull-down resistor.

Chapter 4 Timer Functions



This chapter describes the MN187XX23's timer functions (timers 0 to 5, watchdog timer, and buzzer output circuit).

4-1 Overview

The MN187XX23 has six 8-bit timers, a watchdog timer and a buzzer output circuit.

Timer 0, 1, 3, 4, and 5 are reload timers, and timer 2 is a time-base counter.

Table 4-1-1 Overview of Timer Functions

	Reload Function	Event Count	Timer Output	Synchro-nous Count	Counter Stop	Counter Reset	Synchro-nous Count	Count Clock
Timer 0	○	○	○	○	○	—	—	Choice of 16 clocks: (fs, fx, timer 2 internal output, external input) × (1/1, 1/2^2, 1/2^4, 1/2^6)
Timer 1 *	○	—	○	—	○	—	—	Choice of 5 clocks: fosc, fs/2^4, fs/2^6, fs/2^8, timer 0 internal output
Timer 2	—	—	—	—	○	○	—	Choice of fx/2^7 or fs/2^{12}
Timer 3	○	○	—	—	○	—	○	Choice of 4 clocks: fs, fs/2^2, fs/2^4, external input
Timer 4	○	○	○	○	○	—	—	Choice of 4 clocks: fosc, fs/2^2, fs/2^4, external input
Timer 5	○	—	○	—	○	—	○	Choice of 12 clocks: (fx, fs, timer 4 internal output) × (1/1, 1/2^2, 1/2^4, 1/2^6)
Watchdog timer	———				○	○	—	Generates system reset in fs/2^{16} cycle
Buzzer output circuit	———		○	—	○	○	—	Outputs fs/2^8 or fs/2^9 cycle waveform

When a signal divided from the system clock, fs,, is used as a count clock by a timer, PWM0 should be made operational by setting the PWMOE bit to 1 in the PWM control register (PWMC: X'030' R/W).

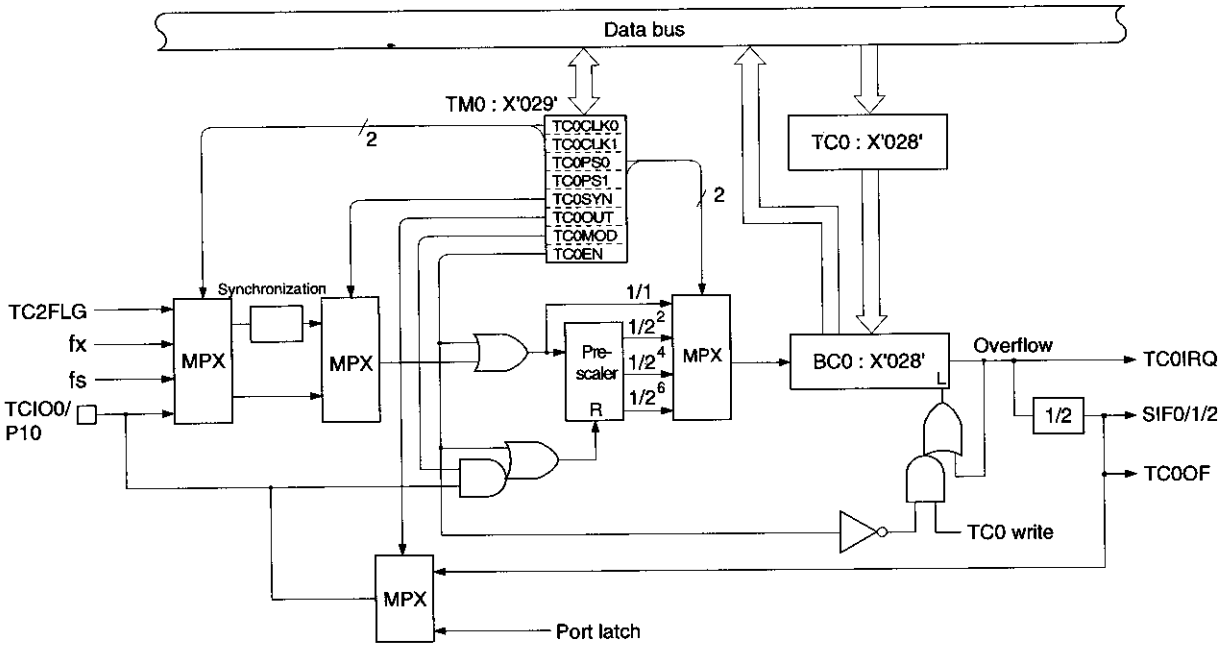


Figure 4-1-1 Timer 0 Block Diagram

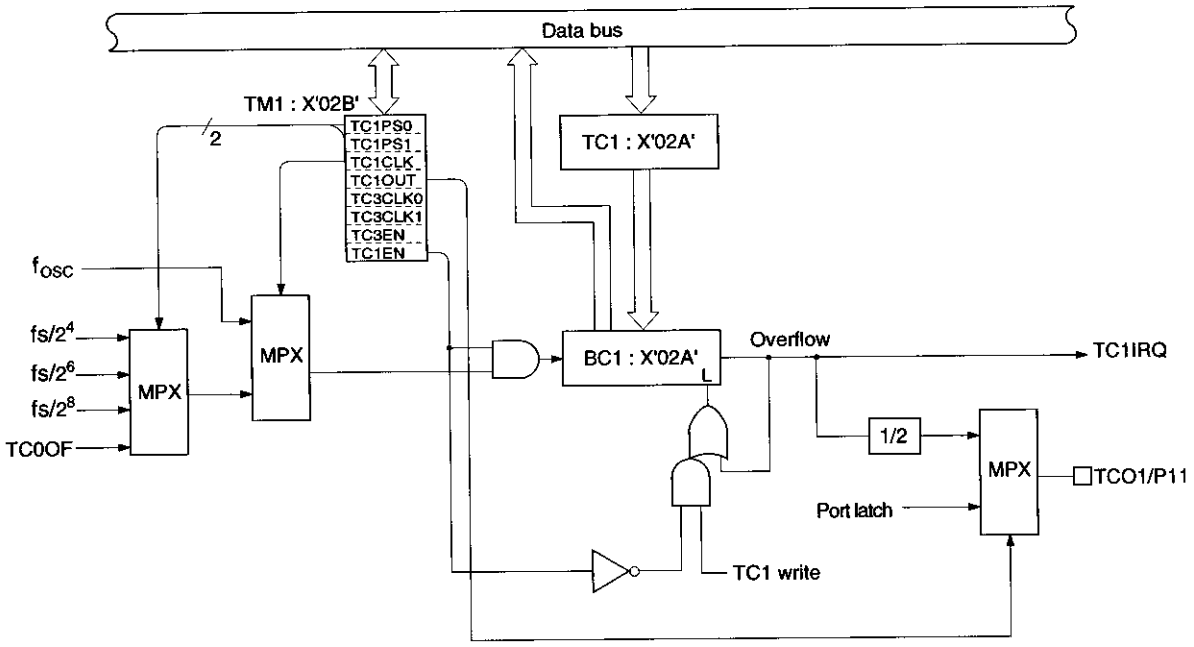


Figure 4-1-2 Timer 1 Block Diagram

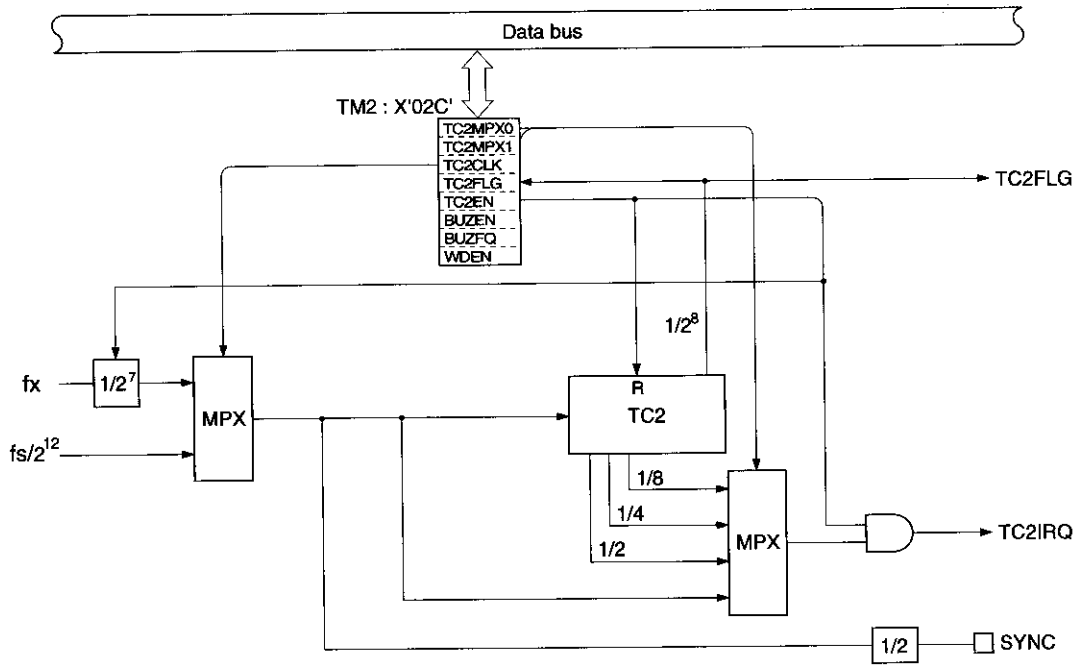


Figure 4-1-3 Timer 2 Block Diagram

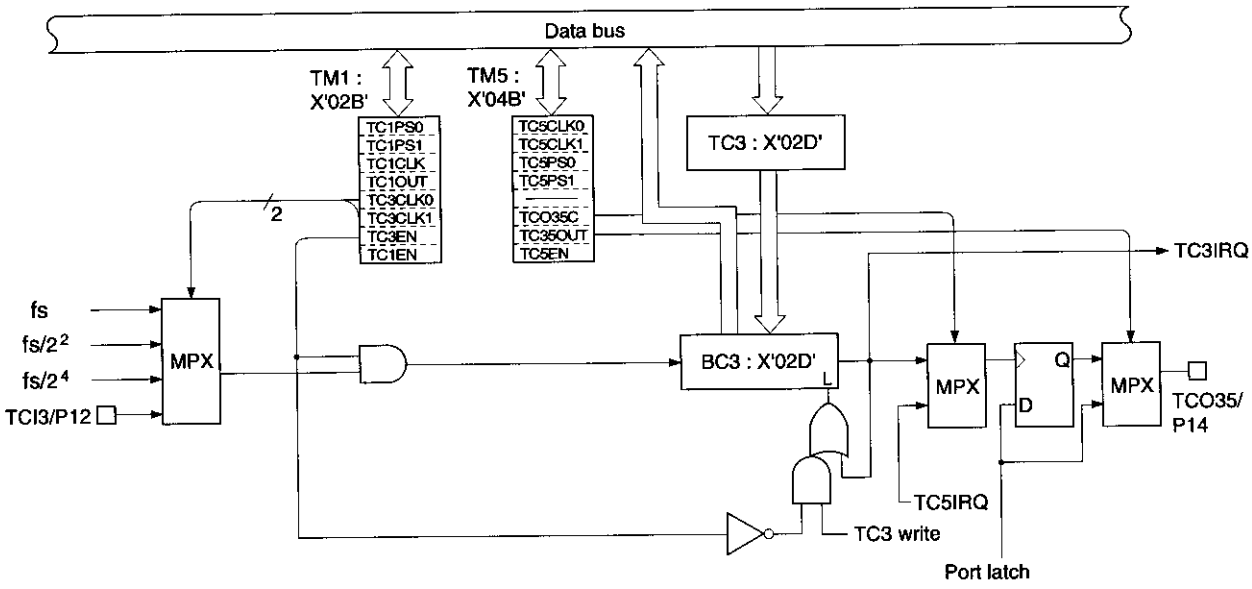


Figure 4-1-4 Timer 3 Block Diagram

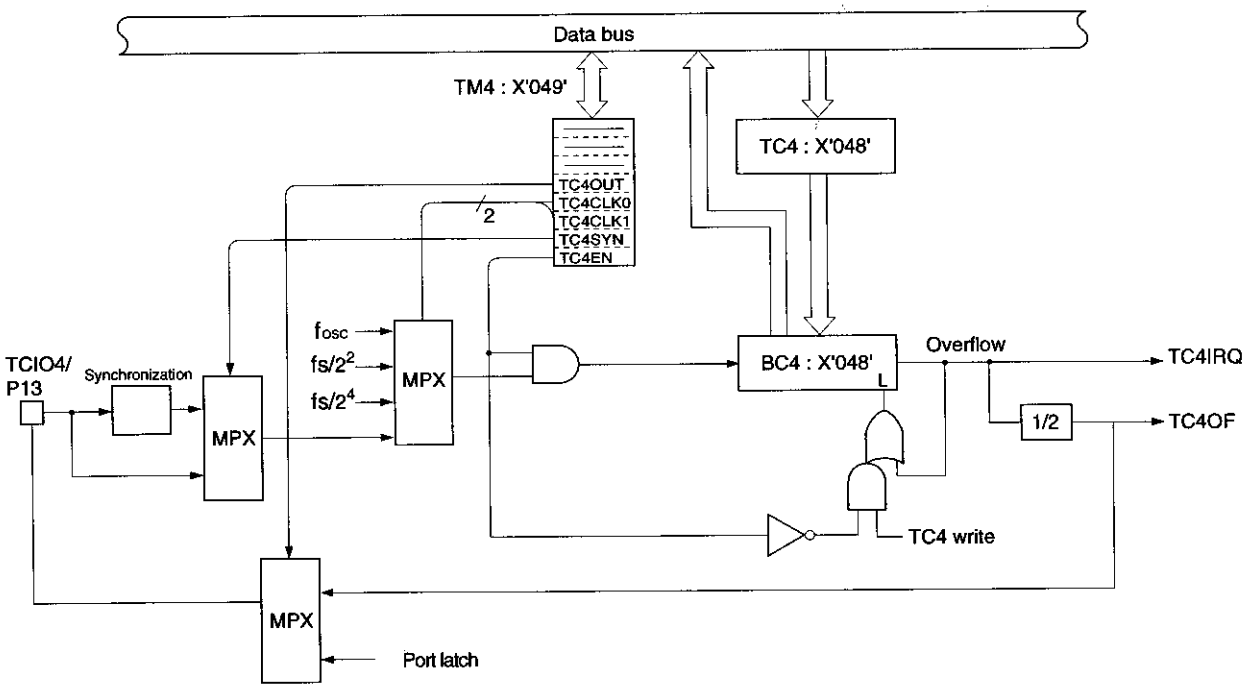


Figure 4-1-5 Timer 4 Block Diagram

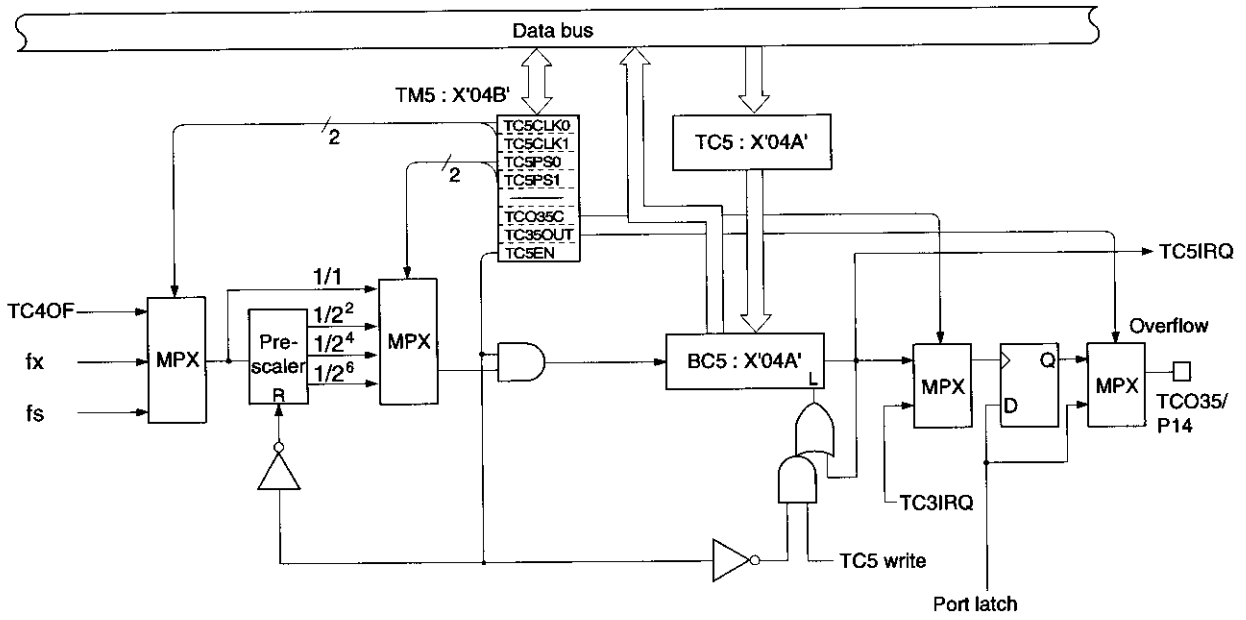


Figure 4-1-6 Timer 5 Block Diagram

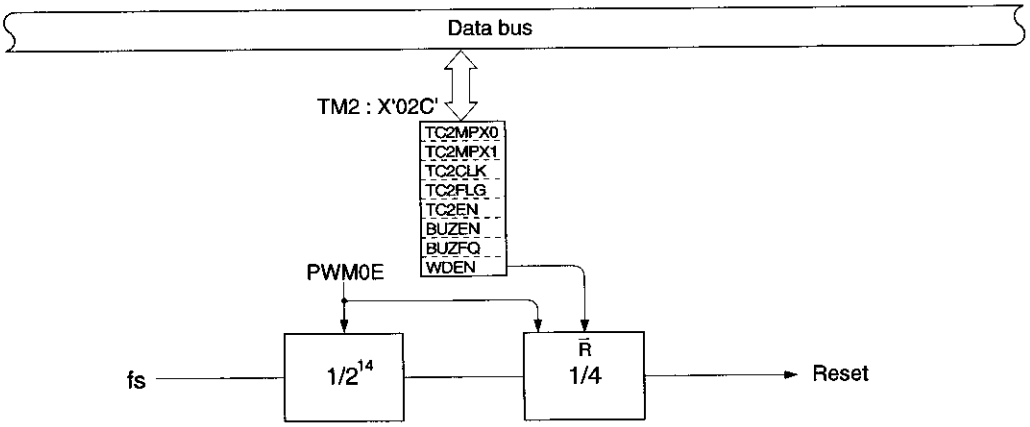


Figure 4-1-7 Watchdog Timer Block Diagram

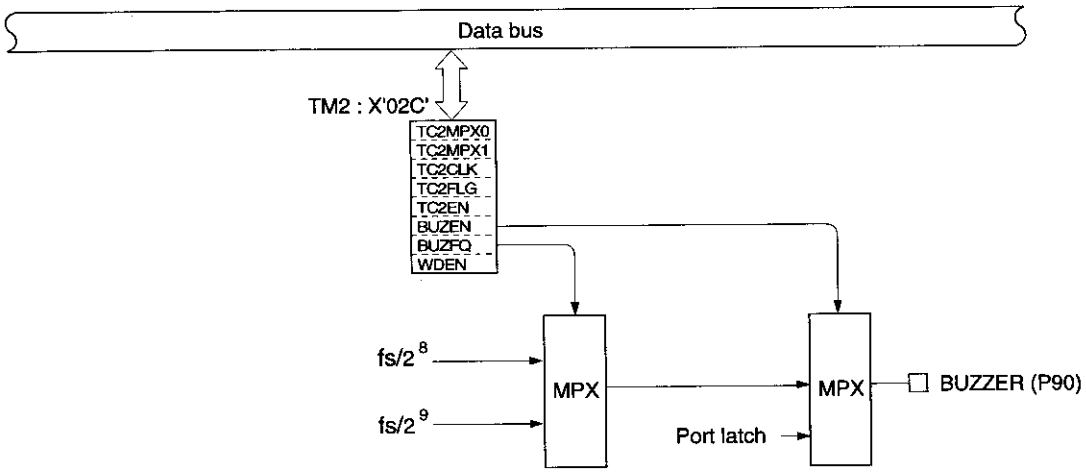


Figure 4-1-8 Buzzer Output Circuit Block Diagram

<Legend>

fx: XI, XO oscillator source oscillation frequency clock (normally 32.768kHz)

fs: Clock derived by scaling OSC1, OSC2 oscillator source oscillation frequency (fosc) / XI, XO oscillator source oscillation frequency (fx) clock by 1/4 or 1/8 (scaling factor selected by CLKSEL bit)

TCIO0/P10: Timer 0 event input/timer 0 output

TC0OF: Signal with frequency of 1/2 timer 0 overflow cycle. Can be used as timer 1 clock source.

TC0IRQ: Timer 0 interrupt signal. Generated by BC0 overflow.

SIF0/1/2: Signal with frequency of 1/2 timer 0 overflow cycle. Can be used as serial interface 0/1/2 transfer clock.

TC1IRQ: Timer 1 interrupt signal. Generated by BC1 overflow.

TC2FLG: Realtime clock flag, reflected in bit 3 of TM2 (normally set for 1 Hz repeat cycle). This signal can be used as timer 0 clock source.

TC2IRQ: Timer 2 interrupt signal. Selection of 4 interrupt cycles by means of TC2MPX1 and TC2MPX0 bits.

TC3IRQ: Timer 3 interrupt signal. Generated by BC3 overflow.

TC4OF: Signal with frequency of 1/2 timer 4 overflow cycle. Can be used as timer 5 clock source.

TC4IRQ: Timer 4 interrupt signal. Generated by BC4 overflow.

TC5IRQ: Timer 5 interrupt signal. Generated by BC5 overflow.

4-2 Timer 0

4-2-1 Overview

Timer 0 is a 6-bit prescaler + 8-bit programmable counter with event counting capability, controlled by timer mode register 0 (TM0).

Timer mode register 0 (TM0: X'029' R/W)
Bit 4: TC0SYN
Bit 3: TC0PS1
Bit 2: TC0PS0
Bit 1: TC0CLK1
Bit 0: TC0CLK0

■ Clock Source

The timer 0 count clock is controlled by (1) prescaler clock source selection and (2) prescaler scaling factor selection. Timer 0 has a built-in synchronization circuit, and can be operated in synchronization with the CPU's system clock.

[☞ "4-10-3 (1) Timer Mode Register 0 (TM0)"]

(1) Prescaler clock source

The prescaler clock source is specified by the TC0CLK1 and TC0CLK0 bits in timer mode register 0 (TM0). Any of four clocks can be selected: fx, fs, an external clock (TCIO0 pin), or timer 2 internal output (TC2FLG).

(2) Prescaler scaling factor

The prescaler scaling factor is specified by the TC0PS1 and TC0PS0 bits in timer mode register 0 (TM0). Any of four scaling factors can be selected: 1/1, 1/4, 1/16, or 1/64 of the prescaler input.

(3) Synchronization circuit

Asynchronous or synchronous mode is selected with the TC0SYN bit in timer mode register 0 (TM0). When synchronous mode is selected, timer 0 counts in synchronization with the CPU's system clock. In synchronous mode, therefore, the correct value is always returned when binary counter 0 (BC0) is read, even if event counting is selected. When an event count is performed in asynchronous mode, it may not always be possible to read the correct value, depending on the read timing.

Timer latch 0/binary counter 0 (TC0/BC0: X'028' R/W)

■ Timer Latch 0/Binary Counter 0 (TC0/BC0)

Timer latch 0 (TC0) and binary counter 0 (BC0) are assigned to the same address (X'028'). Binary counter 0 (BC0) is incremented by operating clock input, and its value can be read at any time by reading address X'028'. When address X'028' is written to, that value is written into timer latch 0 (TC0).

Timer 0 cannot be used in synchronous mode when it is used in STOP mode, or when an external clock with a frequency higher than fs/2 is used as its source.

4-2-2 Timer 0 Operation

(1) Reload operation

When a value is written to timer latch 0 (TC0) while the TC0EN bit is cleared to 0 in timer mode register 0 (TM0), the same value is simultaneously loaded into binary counter 0 (BC0).

When the TC0EN bit in timer mode register 0 (TM0) is set to 1, binary counter 0 (BC0) starts counting. When BC0 overflows, an interrupt is generated and the IFTC0 bit is set in interrupt request flag register 0 (IF0). At the same time, the timer latch 0 (TC0) value is loaded into binary counter 0 (BC0), and the count continues from that value.

When address X'028' is read, the value in binary counter 0 (BC0) is returned.

Timer 0 has a count stop function, whereby the count is stopped when the TC0EN bit in timer mode register 0 (TM0) is cleared to 0. The prescaler is cleared at the same time.

When a value is written to timer latch 0 (TC0) while the TC0EN bit is set to 1 in timer mode register 0 (TM0), that value is not loaded into binary counter 0 (BC0) immediately. The timer latch 0 (TC0) value is loaded into binary counter 0 (BC0) when overflow occurs while the TC0EN bit is set to 1.

(2) Pulse width measurement function

Setting the TC0MOD bit to 1 in timer mode register 0 (TM0) enables the low-level width of pulses input from the TCIO0 pin to be measured. In pulse width measurement mode, binary counter 0 (BC0) counts while the TCIO0 pin input is low, and stops counting, retaining the count value, when the input goes high. The prescaler is cleared at this time. Long low-level pulse measurement can be performed easily by using the timer 0 interrupt (TC0IRQ).

Timer mode register 0
(TM0: X'029' R/W)
Bit 7: TC0EN
Bit 6: TC0MOD
Bit 4: TC0SYN
Bit 1: TC0CLK1
Bit 0: TC0CLK0

Timer latch 0/binary counter 0
(TC0/BC0: X'028' R/W)

Interrupt request flag register 0
(IF0: X'012' R/W)
Bit 4: IFTC0

(3) Event count function

When the TC0CLK1 and TC0CLK0 bits are set to 00 in timer mode register 0 (TM0), timer 0 functions as an event counter. Binary counter 0 (BC0) is incremented on detection of a negative edge in the TCIO0 pin input. Clearing the TC0SYN bit to 0 in timer mode register 0 (TM0) enables the count to be performed in synchronization with the CPU's system clock.

[☞ Clock Source in "4-2-1 Overview"]

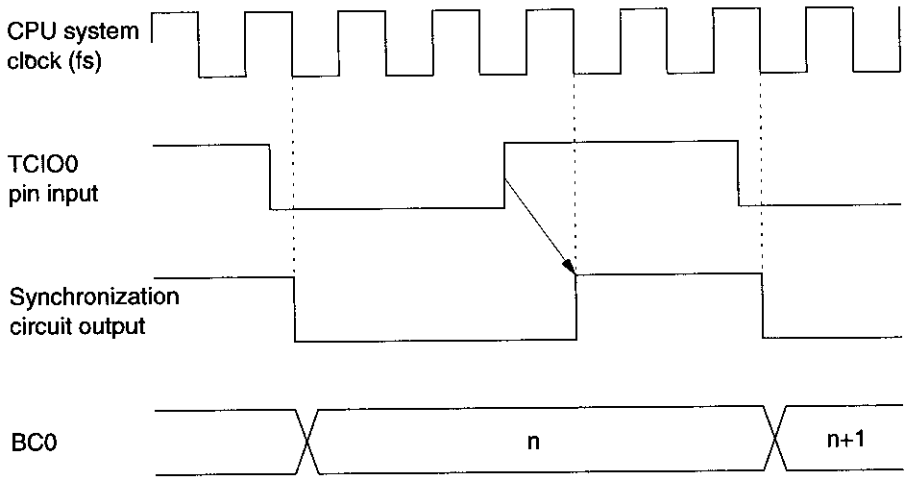


Figure 4-2-1 Timer 0 Event Count Timing (Synchronous Mode)

(4) Serial clock generation

A clock (SIF0/1/2) with 1/2 the frequency of timer 0 overflow can be used as the serial interface 0/1/2 transfer clock.

[☞ SIM0, SIM1, and SIM2 in "5-4-4 Serial Interface Mode Registers (SIM)"]

(5) Timer 1 input clock generation

A clock (TC0OF) with 1/2 the frequency of timer 0 overflow can be used as the timer 1 clock source. [☞ "4-3 Timer 1"]

4-2-3 Timer 0 Interrupt

When timer 0 overflows (the binary counter 0 (BC0) value changes from X'FF' to X'00'), the IFTC0 bit is set to 1 in interrupt request flag register 0 (IF0). If the IETC0 bit in interrupt enable flag register 0 (IE0) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

Interrupt enable flag register 0
(IE0: X'014' R/W)
Bit 4: IETC0

4-3 Timer 1

4-3-1 Overview

Timer 1 is an 8-bit programmable counter, controlled by timer mode register 1 (TM1). The timer 1 carry can be scaled by 1/2 (1:1 duty) and output from the TCO1 pin.

Timer mode register 1
(TM1: '02B' R/W)
Bit 2: TC1CLK
Bit 1: TC1PS1
Bit 0: TC1PS0

■ Clock Source

Timer 1 uses fs/16, fs/64, fs/256, timer 0 internal output (TC0OF), or fosc directly as the count clock. The TC1CLK bit in timer mode register 1 (TM1) specifies whether or not the source oscillation frequency (fosc) is used as the count clock, and the fs scaling factor is selected by bits TC1PS1 and TC1PS0 in timer mode register 1 (TM1). [☞ "4-10-3 (2) Timer Mode Register 1 (TM1)"]

■ Timer Latch 1/Binary Counter 1 (TC1/BC1)

Timer latch 1 (TC1) and binary counter 1 (BC1) are assigned to the same address (X'02A'). Binary counter 1 (BC1) is incremented by operating clock input, and its value can be read at any time by reading address X'02A'. When address X'02A' is written to, that value is written into timer latch 1 (TC1).

Timer latch 1/binary counter 1
(TC1/BC1: X'02A' R/W)

4-3-2 Timer 1 Operation

Interrupt request flag register 0 (IF0: X'012' R/W) Bit 5: IFTC1

(1) Reload operation

When a value is written to timer latch 1 (TC1) while the TC1EN bit is cleared to 0 in timer mode register 1 (TM1), the same value is simultaneously loaded into binary counter 1 (BC1).

When the TC1EN bit in timer mode register 1 (TM1) is set to 1, binary counter 1 (BC1) starts counting. When BC1 overflows, an interrupt is generated and the IFTC1 bit is set in interrupt request flag register 0 (IF0). At the same time, the timer latch 1 (TC1) value is loaded into binary counter 1 (BC1), and the count continues from that value.

When address X'02A' is read, the value in binary counter 1 (BC1) is returned.

Timer 1 has a count stop function, whereby the count is stopped when the TC1EN bit in timer mode register 1 (TM1) is cleared to 0.

When a value is written to timer latch 1 (TC1) while the TC1EN bit is set to 1 in timer mode register 1 (TM1), that value is not loaded into binary counter 1 (BC1) immediately. The timer latch 1 (TC1) value is loaded into binary counter 1 (BC1) when overflow occurs while the TC1EN bit is set to 1.

(2) Timer output function

When the TC1OUT bit is set to 1 in timer mode register 1 (TM1), a pulse with a frequency of 1/2 the binary counter 1 (BC1) overflow (1:1 duty) is output from the TCO1 pin.

Timer mode register 1 (TM1: X'02B' R/W) Bit 3: TC1OUT Bit 7: TC1EN

Interrupt request flag register 0 (IE0: X'014' R/W) Bit 5: IETC1

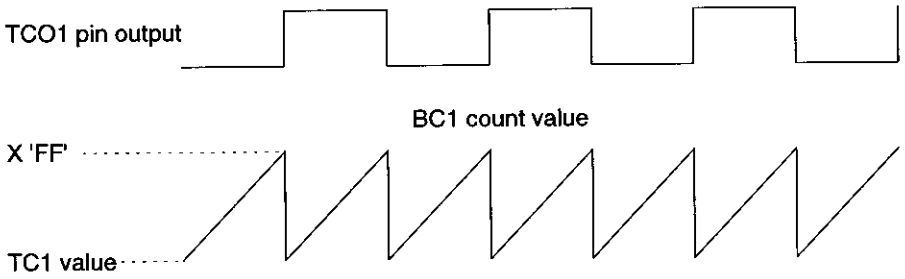


Figure 4-3-1 Timer 1 Output Waveform

4-3-3 Timer 1 Interrupt

When timer 1 overflows (the BC1 value changes from X'FF' to X'00'), the IFTC1 bit is set to 1 in interrupt request flag register 0 (IF0). If the IETC1 bit in interrupt enable flag register 0 (IE0) is 1 at this time, an interrupt request is issued to the CPU.

[☞ "2-8 Interrupt Controller"]

4-4 Timer 2

4-4-1 Overview

Timer 4 is an 8-bit binary counter with a reset function for time base use, controlled by timer mode register 2 (TM2).

■ Clock Source

Either $fs/2^{12}$ or $fx/2^{7}$ can be selected by the TC2CLK bit in timer mode register 2 (TM2). Normally, the setting is made so that a 3.9ms clock is input to timer latch 2 (TC2). The relation between the clock source and the fosc frequency, fx frequency, and CPU mode register (CPUM) settings is shown below.

Timer mode register 2
(TM2: X'02C' R/W)
Bit 4: TC2EN
Bit 3: TC2FLG
Bit 2: TC2CLK
Bit 1: TC2MPX1
Bit 0: TC2MPX0

CPU mode register
(CPUM: X'016' R/W)

Table 4-4-1 Timer 2 Clock Sources

CPUM			fosc=4.19MHz	fosc=8.38MHz	fx=32.768kHz
Bit 6	Bit 3	Bit 2			
CLKSEL	OSCM1	OSCM0			
0	0	0	○	△	○
		1	×	×	○
	1	0	×	×	×
		1	×	×	○
1	0	0	△	○	○
		1	×	×	○
	1	0	×	×	×
		1	×	×	○

<Legend>
○: Setting at which $fs/2^{12}$ or $fx/2^{7}$ is 3.9ms
△: Setting which can be used for TC2 clock source but at which $fs/2^{12}$ is not 3.9ms
×: Setting which cannot be used for TC2 clock source

4-4-2 Timer 2 Operation

(1) Timer 2 interrupt cycle control function

By means of bits TC2MPX1 and TC2MPX0 in timer mode register 2 (TM2), an interrupt generation cycle of 3.9ms, 7.8ms, 15.6ms, or 31.2ms can be selected (in the case of a 3.9ms TC2 input clock).

(2) Realtime clock flag

The value of the MSB of timer latch 2 (TC2) [1 Hz repeat frequency, 1:1 duty (with a 3.9ms TC2 input clock)] is reflected in the TC2FLG bit in timer mode register 2 (TM2), and can be read at all times. Also, this signal can be used as the timer 0 clock source.

Timer 2 has a reset function enabling error to be minimized in realtime clock setting. When the TC2EN bit is cleared to 0 in timer mode register 2 (TM2), timer stops counting and at the same time, timer latch 2 (TC2) is cleared.

(3) SYNC output

An $fs/2^{13}$ or $fx/2^{8}$ clock is output from the SYNC pin. Switching between the $fs/2^{13}$ and $fx/2^{8}$ clock sources is performed by means of the TC2CLK bit in timer mode register 2 (TM2). When $fx/2^{8}$ is selected, the TC2EN bit in timer mode register 2 (TM2) should be set to 1. If $fx/2^{8}$ is selected while the TC2EN bit is cleared to 0, the clock will not be output and the output value will be undefined.

4-4-3 Timer 2 Interrupt

When a timer 2 interrupt is generated at the interval set by the TC2MPX1 and TC2MPX0 bits in timer mode register 2 (TM2), the IFTC2 bit is set to 1 in interrupt request flag register 0 (IF0). If the IETC2 bit in interrupt enable flag register 0 (IE0) is 1 at this time, an interrupt request is issued to the CPU.

[☞ "2-8 Interrupt Controller"]

Interrupt request flag register 0
(IF0: X'012' R/W)
Bit 6: IFTC2

Interrupt enable flag register 0
(IE0: X'014' R/W)
Bit 6: IETC2

4-5 Timer 3

4-5-1 Overvie

Timer 3 is an 8-bit programmable counter with event counting capability, controlled by timer mode register 1 (TM1).

Timer mode register 1
(TM1: '02B' R/W)
Bit 5: TC3CLK1
Bit 4: TC3CLK0

■ Clock Source

Timer 3 uses fs, fs/4, fs/16, or TCI3 pin input directly as the count clock. The clock source is selected by bits TC3CLK1 and TC3CLK0 in timer mode register 1 (TM1).

[☞ "4-10-3 (2) Timer Mode Register 1 (TM1)"]

■ Timer Latch 3/Binary Counter 3 (TC3/BC3)

Timer latch 3 (TC3) and binary counter 3 (BC3) are assigned to the same address (X'02D'). Binary counter 3 (BC3) is incremented by operating clock input, and its value can be read at any time by reading address X'02D'. When address X'02D' is written to, that value is written into timer latch 3 (TC3).

Timer latch 3/binary counter 3
(TC3/BC3: X'02D' R/W)

4-5-2 Timer 3 Operation

Interrupt request flag register 0
(IF0: X'012' R/W)
Bit 7: IFTC3

Interrupt enable flag register 0
(IE0: X'014' R/W)
Bit 7: IETC3

(1) Reload operation

When a value is written to timer latch 3 (TC3) while the TC3EN bit is cleared to 0 in timer mode register 1 (TM1), the same value is simultaneously loaded into binary counter 3 (BC3).

When the TC3EN bit in timer mode register 1 (TM1) is set to 1, binary counter 3 (BC3) starts counting. When BC3 overflows, an interrupt is generated and the IFTC3 bit is set in interrupt request flag register 1 (IF1). At the same time, the timer latch 3 (TC3) value is loaded into binary counter 3 (BC3), and the count continues from that value.

When address X'02D' is read, the value in binary counter 3 (BC3) is returned.

When a value is written to timer latch 3 (TC3) while the TC3EN bit is set to 1 in timer mode register 1 (TM1), that value is not loaded into binary counter 3 (BC3) immediately. The timer latch 3 (TC3) value is loaded into binary counter 3 (BC3) when overflow occurs while the TC3EN bit is set to 1.

Timer 3 has a count stop function, whereby the count is stopped when the TC3EN bit in timer mode register 1 (TM1) is cleared to 0.

(2) Event count function

When the TC3CLK1 and TC3CLK0 bits are set to "11" in timer mode register 1 (TM1), timer 3 functions as an event counter. Binary counter 3 (BC3) is incremented on detection of a negative edge in the TCIO3 pin input.

(3) Synchronous output function

Timer 3 and timer 5 have a synchronous output function that outputs the port 1 bit 4 output latch value from the TCO35 pin (P14) when the timer overflows. When the TC35OUT bit is set to 1 in timer mode register 5 (TM5), the synchronous output function is enabled and the P14 (TCO35) pin functions as the TCO35 pin. Either timer 3 or timer 5 overflow can be selected as the synchronous output trigger by means of the TCO35C bit in timer mode register 5 (TM5) (when TCO35C=1, synchronous output triggered by timer 3 overflow is selected).

When synchronous output is performed, bit 4 of the port 1 direction control register (P1DIR) should be set to 1.

[☞ "4-10-3 (5) Timer Mode Register 5," "3-2-2 Direction Control Register"]

4-5-3 Timer 3 Interrupt

When timer 3 overflows (the BC3 value changes from X'FF' to X'00'), the IFTC3 bit is set to 1 in interrupt request flag register 0 (IF0). If the IETC3 bit in interrupt enable flag register 0 (IE0) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

4-6 Timer 4

4-6-1 Overview

Timer 4 is an 8-bit programmable counter with event counting capability, controlled by timer mode register 4 (TM4). The timer 4 carry can be scaled by 1/2 (1:1 duty) and output from the TCIO4 pin.

■ Clock Source

Timer 4 uses fosc, fs/4, fs/16, or TCIO4 pin input directly as the count clock. The clock source is selected by bits TC4CLK0 and TC4CLK1 in timer mode register 4 (TM4).

Timer 4 has a built-in synchronization circuit, and external input can be synchronized with the CPU's system clock. Asynchronous or synchronous mode is selected with the TC4SYN bit in timer mode register 4 (TM4). When synchronous mode is selected, timer 4 counts in synchronization with the CPU's system clock. In synchronous mode, therefore, the correct value is always returned when binary counter 4 (BC4) is read, even if event counting is selected. When an event count is performed in asynchronous mode, it may not always be possible to read the correct value, depending on the read timing.

[☞ "4-10-3 (5) Timer Mode Register 4 (TM4)"]

Timer 4 cannot be used in synchronous mode when it is used in STOP mode, or when an external clock with a frequency higher than fs/2 is used as its source

■ Timer Latch 4/Binary Counter 4 (TC4/BC4)

Timer latch 4 (TC4) and binary counter 4 (BC4) are assigned to the same address (X'048'). Binary counter 4 (BC4) is incremented by operating clock input, and its value can be read at any time by reading address X'048'. When address X'048' is written to, that value is written into timer latch 4 (TC4) and also into binary counter 4 (BC4).

When a value is written to timer latch 4 (TC4) while the TC4EN bit is set to 1 in timer mode register 4 (TM4), that value is not loaded into binary counter 4 (BC4) immediately. The timer latch 4 (TC4) value is loaded into binary counter 4 (BC4) when overflow occurs while the TC4EN bit is set to 1.

4-6-2 Timer 4 Operation

(1) Reload operation

When a value is written to timer latch 4 (TC4) while the TC4EN bit is cleared to 0 in timer mode register 4 (TM4), the same value is simultaneously loaded into binary counter 4 (BC4). When the TC4EN bit in timer mode register 4 (TM4) is set to 1, binary counter 4 (BC4) starts counting. When BC4 overflows, an interrupt is generated and the IFTC4 bit is set in interrupt request flag register 1 (IF1). At the same time, the timer latch 4 (TC4) value is loaded into binary counter 4 (BC4), and the count continues from that value.

When address X'048' is read, the value in binary counter 4 (BC4) is returned.

Timer 4 has a count stop function, whereby the count is stopped when the TC4EN bit in timer mode register 4 (TM4) is cleared to 0.

(2) Timer output function

When the TC4OUT bit is set to "1" in timer mode register 4 (TM4), a pulse with a frequency of 1/2 the binary counter 4 (BC4) overflow (1:1 duty) is output from the TCIO4 pin.

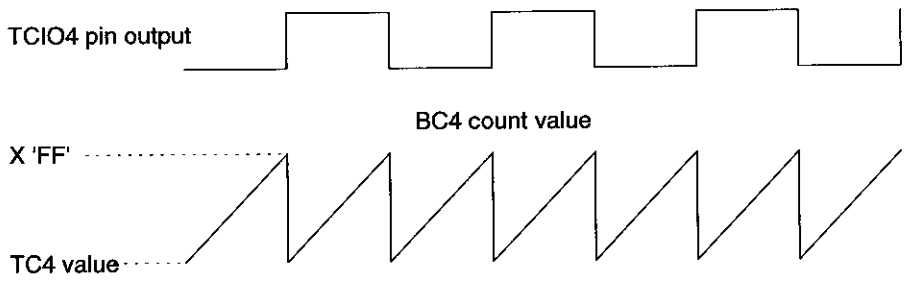


Figure 4-6-1 Timer 4 Output Waveform

(3) Event count operation

When the TC4CLK1 and TC4CLK0 bits are set to "11" in timer mode register 4 (TM4), timer 4 functions as an event counter. Binary counter 4 (BC4) is incremented on detection of a negative edge in the TCIO4 pin input.

Clearing the TC4SYN bit to 0 in timer mode register 4 (TM4) enables the count to be performed in synchronization with the CPU's system clock.

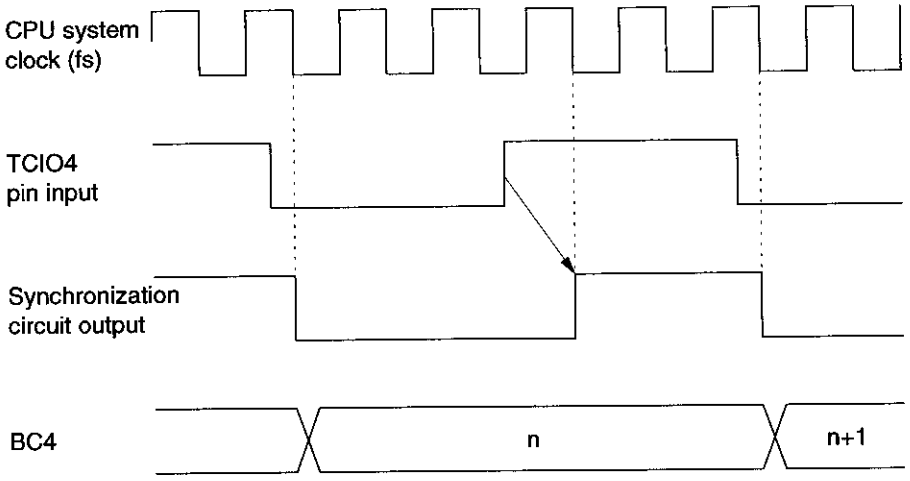


Figure 4-6-2 Timer 4 Event Count Timing (Synchronous Mode)

(4) Timer 5 input clock/remote control carrier generation

A clock (TC4OF) with 1/2 the frequency of timer 0 overflow can be used as the timer 5 count clock. In remote control output, TC4OF can be used as the remote control carrier.

[☞ "Chapter 10 Remote Control Transmission"]

4-6-3 Timer 4 Interrupt

When timer 4 overflows (the binary counter 4 (BC4) value changes from X'FF' to X'00'), the IFTC4 bit is set to 1 in interrupt request flag register 1 (IF1). If the IETC4 bit in interrupt enable flag register 1 (IE1) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 0: IFTC4

Interrupt enable flag register 1
(IE1: X'015' R/W)
Bit 0: IETC4

4-7 Timer 5

4-7-1 Overview

Timer 5 is a 6-bit prescaler + 8-bit programmable counter with event counting capability, controlled by timer mode register 5 (TM5).

■ Clock Source

The timer 5 count clock is controlled by (1) prescaler clock source selection and (2) prescaler scaling factor selection.

[☞ "4-10-3 (6) Timer Mode Register 5 (TM5)"]

(1) Prescaler clock source

The prescaler clock source is specified by the TC5CLK1 and TC5CLK0 bits in timer mode register 5 (TM5). Any of three clocks can be selected: fx, fs, or timer 4 internal output (TC4OF).

(2) Prescaler scaling factor

The prescaler scaling factor is specified by the TC5PS1 and TC5PS0 bits in timer mode register 5 (TM5). Any of four scaling factors can be selected: 1/1, 1/4, 1/16, or 1/64 of the prescaler input.

■ Timer Latch 5/Binary Counter 5 (TC5/BC5)

Timer latch 5 (TC5) and binary counter 5 (BC5) are assigned to the same address (X'04A'). Binary counter 5 (BC5) is incremented by operating clock input, and its value can be read at any time by reading address X'04A'. When address X'04A' is written to, that value is written into timer latch 5 (TC5) and also into binary counter 5 (BC5).

4-7-2 Timer 5 Operation

When a value is written to timer latch 5 (TC5) while the TC5EN bit is set to 1 in timer mode register 5 (TM5), that value is not loaded into binary counter 5 (BC5) immediately. The timer latch 5 (TC5) value is loaded into binary counter 5 (BC5) when overflow occurs while the TC5EN bit is set to 1.

(1) Reload operation

When a value is written to timer latch 5 (TC5) while the TC5EN bit is cleared to 0 in timer mode register 5 (TM5), the same value is simultaneously loaded into binary counter 5 (BC5). When the TC5EN bit in timer mode register 5 (TM5) is set to 1, binary counter 5 (BC5) starts counting. When BC5 overflows, an interrupt is generated and the IFTC5 bit is set in interrupt request flag register 1 (IF1). At the same time, the timer latch 5 (TC5) value is loaded into binary counter 5 (BC5), and the count continues from that value.

When address X'04A' is read, the value in binary counter 5 (BC5) is returned.

Timer 5 has a count stop function, whereby the count is stopped when the TC5EN bit in timer mode register 5 (TM5) is cleared to 0. The prescaler is cleared at the same time.

(2) Synchronous output function

Timer 3 and timer 5 have a synchronous output function that outputs the port 1 bit 4 output latch value from the TCO35 pin (P14) when the timer overflows. When the TC35OUT bit is set to 1 in timer mode register 5 (TM5), the synchronous output function is enabled and the P14 (TCO35) pin functions as the TCO35 pin. Either timer 3 or timer 5 overflow can be selected as the synchronous output trigger by means of the TCO35C bit in timer mode register 5 (TM5) (when TCO35C=0, synchronous output triggered by timer 5 overflow is selected).

When synchronous output is performed, bit 4 of the port 1 direction control register (P1DIR) should be set to 1.

[☞ "4-10-3 (5) Timer Mode Register 5," "3-2-2 Direction Control Register"]

(3) Remote control output function

In remote control output, the timer 5 overflow signal (TC5IRQ) is used as the remote control basic pulse. [☞ "Chapter 10 Remote Control Transmission"]

4-7-3 Timer 5 Interrupt

When timer 5 overflows (the binary counter 5 (BC5) value changes from X'FF' to X'00'), the IFTC5 bit is set to 1 in interrupt request flag register 1 (IF1). If the IETC5 bit in interrupt enable flag register 1 (IE1) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 1: IFTC5

Interrupt enable flag register 1
(IE1: X'015' R/W)
Bit 1: IETC5

4-8 Watchdog Timer

4-8-1 Overview

Timer mode register 2
(TM2: X'02C' R/W)
Bit 7: WDEN

PWM control register
(PWMC: X'030' R/W)
Bit 2: PWM0E

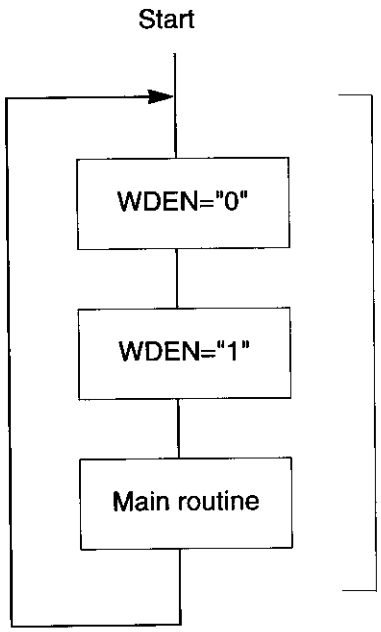
The watchdog timer can be used to detect program runaway, and is controlled by timer mode register 2 (TM2).

The watchdog timer uses the PWM0 14-bit counter for its count operations. Therefore, when the watchdog timer is used, the PWM0E bit in the PWM control register (PWMC) should be set to 1 to select PWM operation. The watchdog timer starts operating when the WDEN bit is set to 1 in timer mode register 2 (TM2), and generates a system reset at an interval of fs/2^{16}.

[☞ "7-5-2 PWM Control Register (PWMC)"]

4-8-2 Watchdog Timer Operation

After the WDEN bit has been set to 1 in timer mode register 2 (TM2), 0 will automatically be output to the $\overline{RST}$ pin, causing a CPU reset, unless the WDEN bit is cleared to 0 within a given time (TWD). An outline program flowchart is shown in figure 4-8-1.



The time required to execute this coding, including interrupt handler execution, must be shorter than the minimum value of TWD. In some cases, WDEN bit resetting/setting may be included at a number of points.

Figure 4-8-1 Example of Use of Watchdog Timer

$$\frac{3 \cdot 2^{14}}{fs} \leq T_{WD} \leq \frac{2^{16}}{fs}$$

The interval at which the WDEN bit in timer mode register 2 (TM2) is reset/set in the program must be smaller than the minimum value of TWD.

When the WDEN bit in timer mode register 2 (TM2) is cleared to 0, the 1/4-scaling counter is cleared but the PWM0 14-bit counter is not affected. Therefore, in the worst case, depending on the timing of watchdog timer clearance, the watchdog reset will be generated at a 3/4 interval. Consequently, when repeating WDEN bit clearing and setting at regular intervals, a cycle not exceeding 3/4 of $fs/2^{16}$ should be used.

4-9 Buzzer Output

4-9-1 Overview

Timer mode register 2
(TM2: X'02C' R/W)
Bit 6: BUZFQ
Bit 5: BUZEN

PWM control register
(PWMC: X'030' R/W)
Bit 2: PWM0E

An fs/2^9 or fs/2^8 square wave can be output at the buzzer output pin (P90/BUZZER). Buzzer output is controlled by timer mode register 2 (TM2). The buzzer output waveform is generated using the PWM0 14-bit counter. Therefore, when buzzer output is used, the PWM0E bit in the PWM control register (PWMC) should be set to 1 to select PWM operation.

[☞ "7-5-2 PWM Control Register (PWMC)"]

4-9-2 Buzzer Output Operation

CPU mode register
(CPUM: X'016' R/W)

After the output frequency has been set by means of the BUZFQ bit in timer mode register 2 (TM2), a 1:1 duty square wave is output at the BUZZER pin by setting the BUZEN bit in timer mode register 2 (TM2) to 1. The buzzer output frequency is selected according to the settings of the BUZFQ bit and CPU mode register (CPUM) bits as shown in the following table.

Table 4-9-1 Buzzer Output Frequencies

CPUM			TM2	System Clock (fs) Source Oscillation	
Bit 3	Bit 2	Bit 6	Bit 6	fosc=4.19MHz	fosc=8.38MHz
OSCM1	OSCM0	CLKSEL	BUZFQ	Buzzer output frequency	
0	0	0	0	2kHz	4kHz
			1	4kHz	8kHz
		1	0	1kHz	2kHz
			1	2kHz	4kHz
	1	0	0		
			1		
		1	0		
			1		
1	0	0	0	Setting prohibited	
			1		
		1	0		
			1		
	1	0	0		
			1		
		1	0		
			1		

: Settings not suitable for buzzer output, since the system clock (fs) is generated using 32kHz source oscillation.

4-10 Timer Control Registers

4-10-1 Overview

Ten register bytes are used to control the MN187XX23's timers: timer mode registers (TM0, TM1, TM2, TM4, TM5) which control timer operation, and timer latches/binary counters (TC0/BC0, TC1/BC1, TC2/BC2, TC4/BC4, TC5/BC5) used for reload value setting and count value reading. These registers are all located in RAM space.

Table 4-10-1 Timer Control Registers

Register Abbreviation	RAM Address	R/W	Register Name
TC0/BC0	X'028'	R*/W	Programmable timer counter 0 (timer latch 0/binary counter 0)
TM0	X'029'	R/W	Timer mode register 0
TC1/BC1	X'02A'	R*/W	Programmable timer counter 1 (timer latch 1/binary counter 1)
TM1	X'02B'	R/W	Timer mode register 1
TM2	X'02C'	R/W	Timer mode register 2
TC3/BC3	X'02D'	R*/W	Programmable timer counter 3 (timer latch 3/binary counter 3)
TC4/BC4	X'048'	R*/W	Programmable timer counter 4 (timer latch 4/binary counter 4)
TM4	X'049'	R/W	Timer mode register 4
TC5/BC5	X'04A'	R*/W	Programmable timer counter 5 (timer latch 5/binary counter 5)
TM5	X'04B'	R/W	Timer mode register 5

R/W: Can be both read and written to.

R*/W: When read, the binary counter value is returned.
When written to, the value is written into the timer latch.

4-10-2 Programmable Timer Counters

Timers 0, 1, 3, 4, and 5 are 8-bit programmable timer counters.
Each programmable timer counter consists of a timer latch (TC) and a binary counter (BC).

(1) Programmable timer counter 0 (TC0/BC0)

* *When read, the binary counter value is returned; when written to, the value is written into the timer latch.*

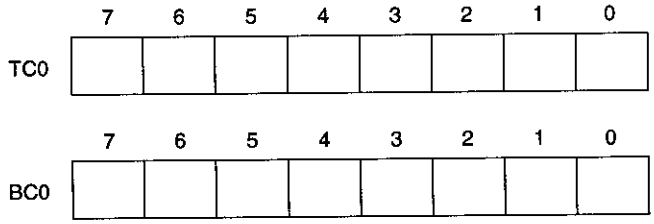


Figure 4-10-1 Programmable Timer Counter 0 (TC0/BC0: X'028' R*/W)

Timer mode register 0 (TM0: X'029' R/W) Bit 7: TC0EN

When a value is written to timer latch 0 (TC0) while the TC0EN bit is cleared to 0 in timer mode register 0 (TM0), the same value is simultaneously loaded into binary counter 0 (BC0).
When the timer is enabled—that is, when the TC0EN bit in timer mode register 0 (TM0) is set to 1—binary counter 0 (BC0) starts counting. When BC0 overflows, the timer latch 0 (TC0) value is loaded into binary counter 0 (BC0), and the count continues from that value. [☞ "4-2-2 (1) Reload operation"]

(2) Programmable timer counter 1 (TC1/BC1)

* *When read, the binary counter value is returned; when written to, the value is written into the timer latch.*

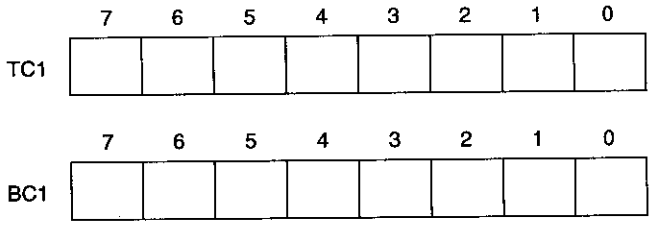


Figure 4-10-2 Programmable Timer Counter 1 (TC1/BC1: X'02A' R*/W)

When a value is written to timer latch 1 (TC1) while the TC0EN bit is cleared to 0 in timer mode register 1 (TM1), the same value is simultaneously loaded into binary counter 1 (BC1).
When the timer is enabled—that is, when the TC1EN bit in timer mode register 1 (TM1) is set to 1—binary counter 1 (BC1) starts counting. When BC1 overflows,

the timer latch 1 (TC1) value is loaded into binary counter 1 (BC1), and the count continues from that value. [☞ "4-3-2 (1) Reload operation"]

(3) Programmable timer counter 3 (TC3/BC3)

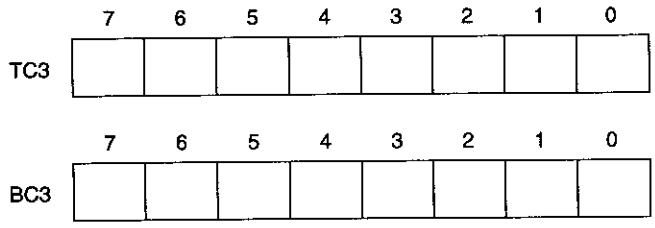


Figure 4-10-3 Programmable Timer Counter 3 (TC3/BC3: X'02D' R*/W)

* *When read, the binary counter value is returned; when written to, the value is written into the timer latch.*

When a value is written to timer latch 3 (TC3) while the TC3EN bit is cleared to 0 in timer mode register 1 (TM1), the same value is simultaneously loaded into binary counter 3 (BC3).

When the timer is enabled—that is, when the TC3EN bit in timer mode register 1 (TM1) is set to 1—binary counter 3 (BC3) starts counting. When BC3 overflows, the timer latch 3 (TC3) value is loaded into binary counter 3 (BC3), and the count continues from that value. [☞ "4-5-2 (1) Reload operation"]

(4) Programmable timer counter 4 (TC4/BC4)

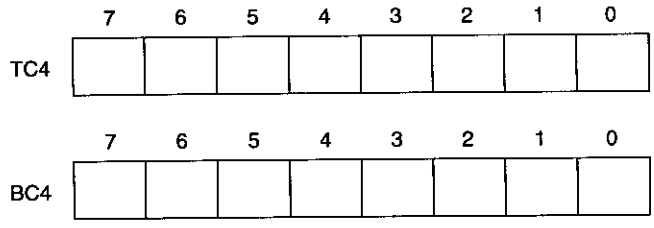


Figure 4-10-4 Programmable Timer Counter 4 (TC4/BC4: X'048' R*/W)

* *When read, the binary counter value is returned; when written to, the value is written into the timer latch.*

When a value is written to timer latch 4 (TC4) while the TC4EN bit is cleared to 0 in timer mode register 4 (TM4), the same value is simultaneously loaded into binary counter 4 (BC4).

When the timer is enabled—that is, when the TC4EN bit in timer mode register 4 (TM4) is set to 1—binary counter 4 (BC4) starts counting. When BC4 overflows, the timer latch 4 (TC4) value is loaded into binary counter 4 (BC4), and the count continues from that value.

(5) Programmable timer counter 5 (TC5/BC5)

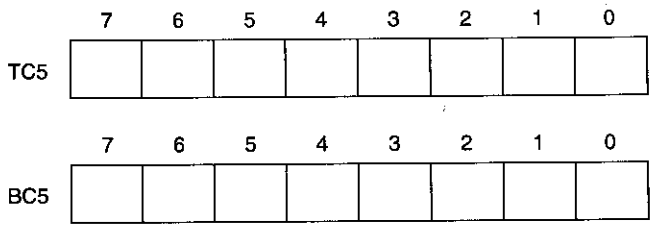


Figure 4-10-5 Programmable Timer Counter 5 (TC5/BC5: X'04A' R*/W)

* *When read, the binary counter value is returned; when written to, the value is written into the timer latch.*

When a value is written to timer latch 5 (TC5) while the TC5EN bit is cleared to 0 in timer mode register 5 (TM5), the same value is simultaneously loaded into binary counter 5 (BC5).

When the timer is enabled—that is, when the TC5EN bit in timer mode register 5 (TM5) is set to 1—binary counter 5 (BC5) starts counting. When BC5 overflows, the timer latch 5 (TC5) value is loaded into binary counter 5 (BC5), and the count continues from that value.

4-10-3 Timer Mode Registers (TM)

The five timer mode registers (TM0, TM1, TM2, TM4, and TM5) are one-byte, read/write registers that control timers 0, 1, 2, 3, 4, and 5, the watchdog timer, and buzzer output.

(1) Timer mode register 0 (TM0)

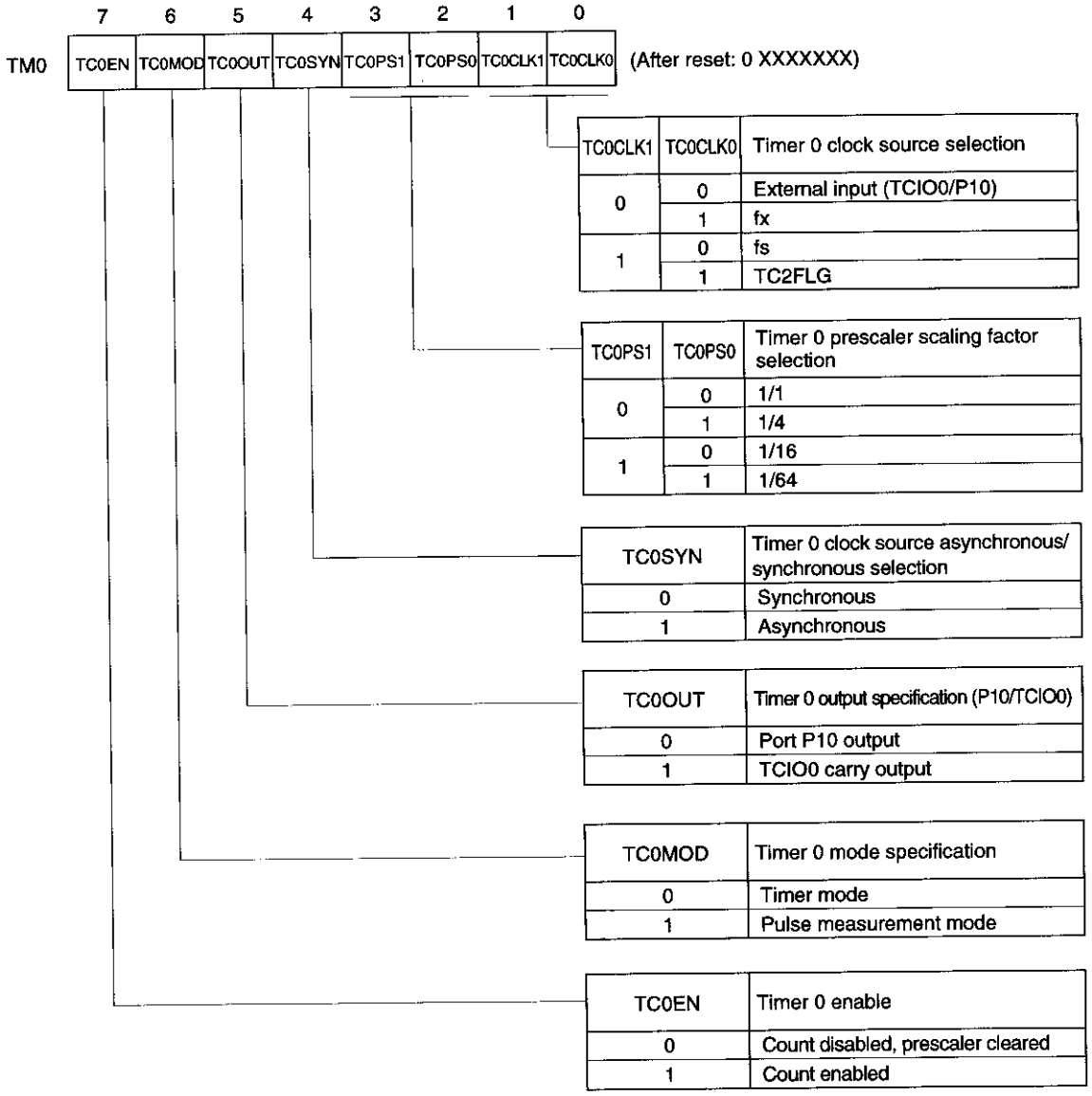
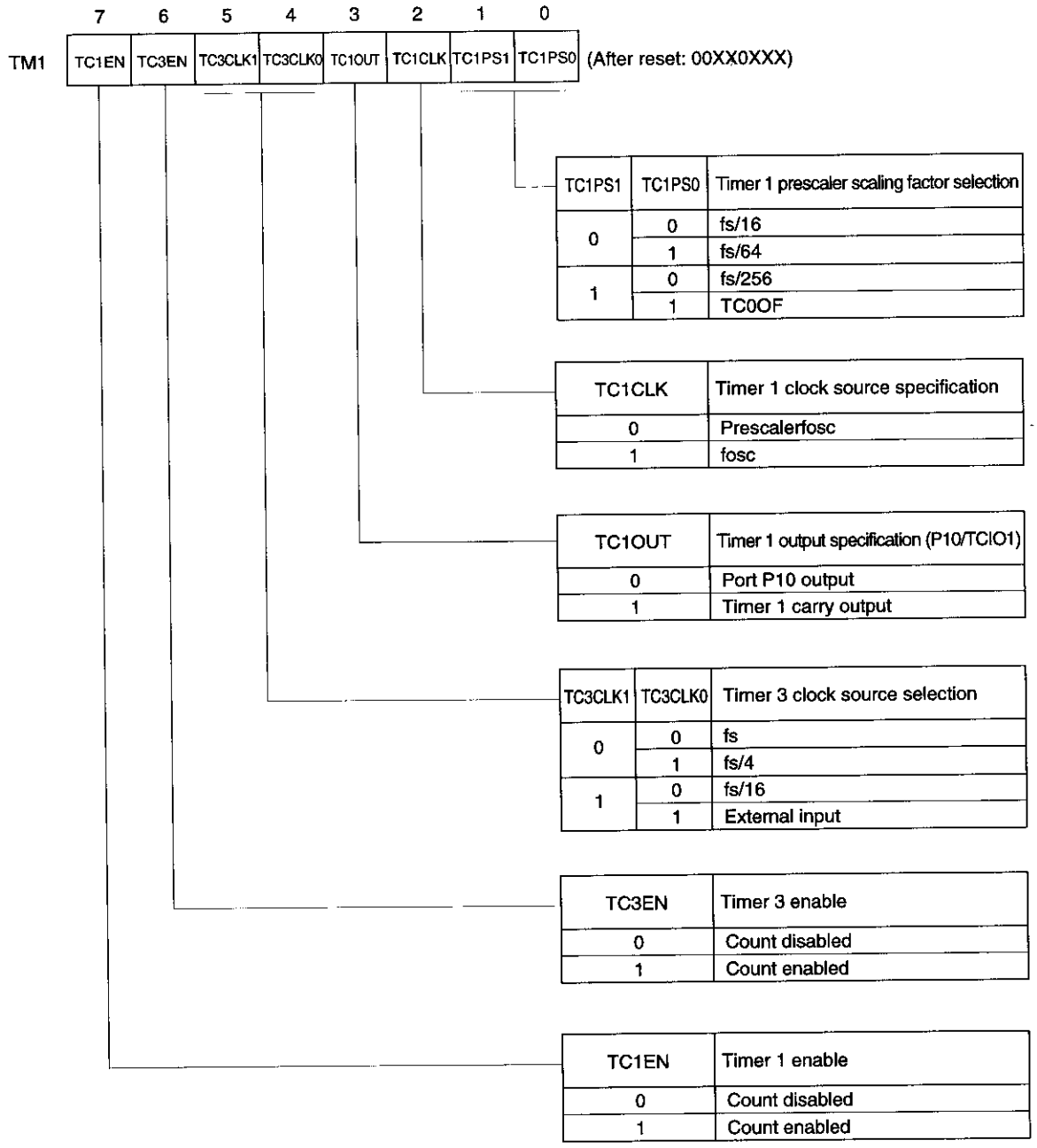


Figure 4-10-6 Timer Mode Register 0 (TM0: X'029' R/W)

In a reset, bit 7 (TC0EN) is cleared to 0 and the other bits are undefined.

If the TC0SYN bit in TM0 is modified when the TC0CLK1 and TC0CLK0 bits in TM0 are set to "10" (fs selected as the clock source), timer 0 may be incremented.

(2) Timer mode register 1 (TM1)



7	6	5	4	3	2	1	0	
TM1	TC1EN	TC3EN	TC3CLK1	TC3CLK0	TC1OUT	TC1CLK	TC1PS1	TC1PS0	(After reset: 00XX0XXX)

TC1PS1	TC1PS0	Timer 1 prescaler scaling factor selection
0	0	fs/16
	1	fs/64
1	0	fs/256
	1	TC0OF

TC1CLK	Timer 1 clock source specification
0	Prescalerfosc
1	fosc

TC1OUT	Timer 1 output specification (P10/TCIO1)
0	Port P10 output
1	Timer 1 carry output

TC3CLK1	TC3CLK0	Timer 3 clock source selection
0	0	fs
	1	fs/4
1	0	fs/16
	1	External input

TC3EN	Timer 3 enable
0	Count disabled
1	Count enabled

TC1EN	Timer 1 enable
0	Count disabled
1	Count enabled

Figure 4-10-7 Timer Mode Register 1 (TM1: X'02B' R/W)

In a reset, bits 3 (TC1OUT), 6 (TC3EN), and 7 (TC1EN) are cleared to 0, and the other bits are undefined.

(3) Timer mode register 2 (TM2)

	7	6	5	4	3	2	1	0	
TM2	WDEN	BUZFQ	BUZEN	TC2EN	TC2FLG	TC2CLK	TC2MPX1	TC2MPX0	(After reset: 0X00X1XX)

TC2MPX1	TC2MPX0	Timer 0 interrupt cycle selection
0	0	$fs/2^{12}$: $fx/2^{7}$
	1	$fs/2^{13}$: $fx/2^{8}$
1	0	$fs/2^{14}$: $fx/2^{9}$
	1	$fs/2^{15}$: $fx/2^{10}$

TC2CLK	Timer 2 clock source selection
0	$fx/2^{7}$
1	$fs/2^{12}$

TC2FLG	Realtime clock flag
1Hz cycle, 1:1 duty	When TC2CLK=0 and fosc=4.19MHz, CLKSEL=0, or fosc=8.38MHz, CLKSEL=1
	When TC2CLK=1 and fx=32kHz

TC2EN	Timer 2 enable
0	Count disabled, counter cleared
1	Count enabled

BUZEN	Buzzer output enabled
0	Disabled (P11 port input/output)
1	Output enabled

BUZFQ	Buzzer cycle selection
0	$fs/2^{9}$
1	$fs/2^{8}$

WDEN	Watchdog timer enable
0	Watchdog timer stopped, 1/4-scaling counter cleared
1	Watchdog timer enabled

Figure 4-10-8 Timer Mode Register 2 (TM2: X'02C' R/W)

In a reset, bit 2 (TC2CLK) is set to 1, bits 4 (TC2EN), 5 (BUZEN), and 7 (WDEN) are cleared to 0, and the other bits are undefined.

(4) Timer mode register 4 (TM4)

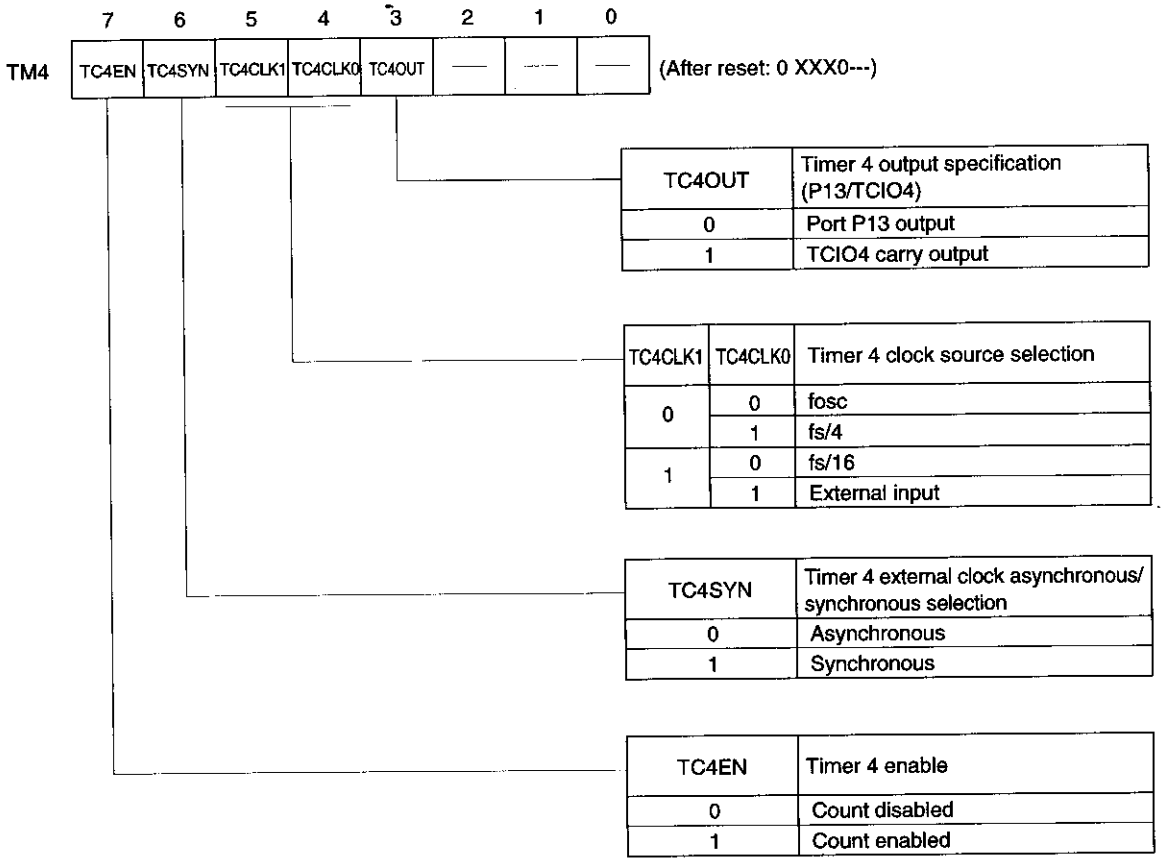


Figure 4-10-9 Timer Mode Register 4 (TM4: X'049' R/W)

In a reset, bits 3 (TC4OUT) and 7 (TC4EN) are cleared to 0, and the other bits are undefined.

(5) Timer mode register 5 (TM5)

	7	6	5	4	3	2	1	0	
TM5	TC5EN	TC35OUT	TCO35C	—	TC5PS1	TC5PS0	TC5CLK1	TC5CLK0	(After reset: 000-XXXX)

TC5CLK1	TC5CLK0	Timer 5 clock source selection
0	0	TC4OF
	1	fx
1	0	fs
	1	Setting prohibited

TC5PS1	TC5PS0	Timer 5 prescaler scaling factor selection
0	0	1/1
	1	1/4
1	0	1/16
	1	1/64

TCO35C	Timer 3/timer 5 synchronous output specification (P14/TC035)
0	Synchronous output on timer 5 overflow
1	Synchronous output on timer 3 overflow

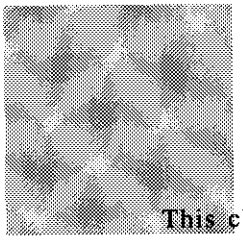
TC35OUT	Synchronous output specification
0	Port P14 output
1	Synchronous output

TC5EN	Timer 5 enable
0	Count disabled, prescaler cleared
1	Count enabled

Figure 4-10-10 Timer Mode Register 5 (TM5: X'04B' R/W)

In a reset, bits 5 (TCO35C), 6 (TC3OUT), and 7 (TC5EN) are cleared to 0, and the other bits are undefined.

Chapter 5 Serial Functions



This chapter describes the MN187XX23's serial interfaces.

5-1 Overview

The MN187XX23 has three synchronous serial interfaces, with the following features.

■ Continuous transmission/reception

Two of the serial interfaces—0 and 1—can be connected in parallel, allowing continuous transmission and reception.

■ Compatibility

Communication can be performed with the serial interfaces of the MN1500, MN1700, MN1860, MN1870, and MN1880 Series.

■ Clock rate

Internal clock: Can be selected from fs, fs/8, fs/16, and timer 0 internal output (BC0/2: output at 1/2 timer 0 overflow rate).

External clock: Serial interface 0 can use the P02 (SBT0) or P06 (PSBT0) pin as the clock input/output pin, serial interface 1 can use the P05 (SBT1) pin, and serial interface 2 can use the P93 (SBT2) pin (P06 can be used for clock input only).

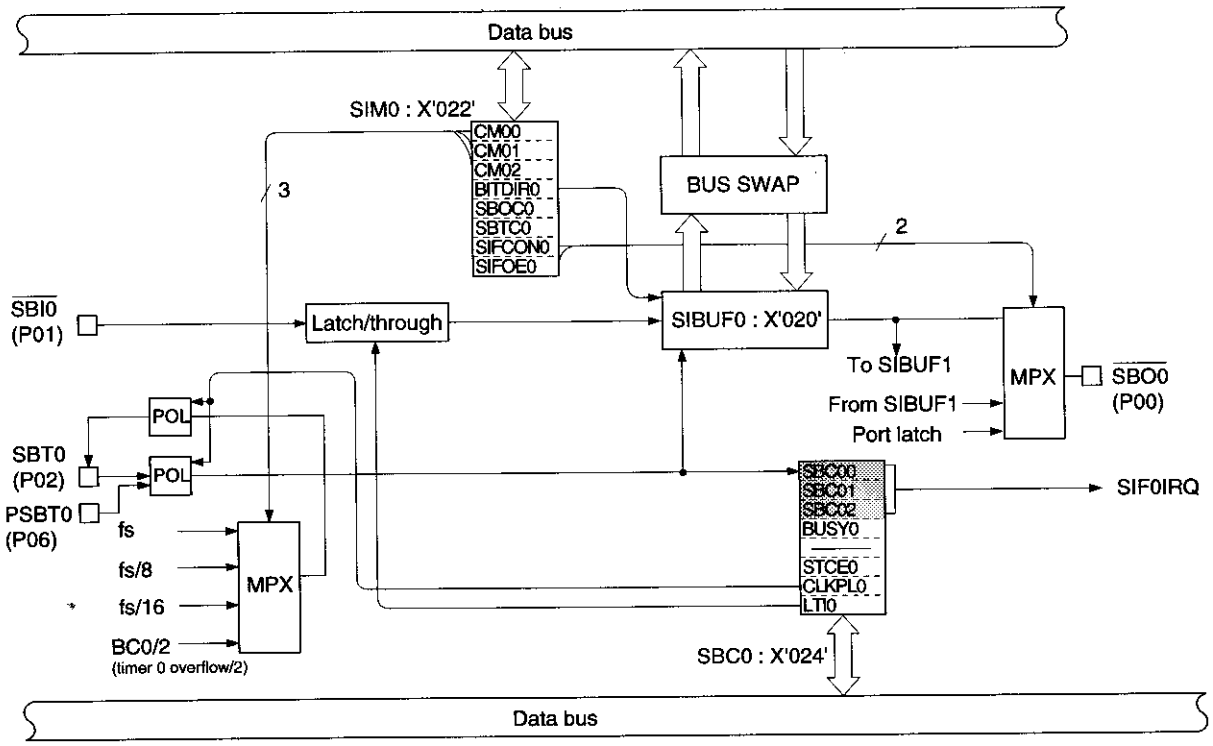


Figure 5-1-1 Serial Interface 0 Block Diagram

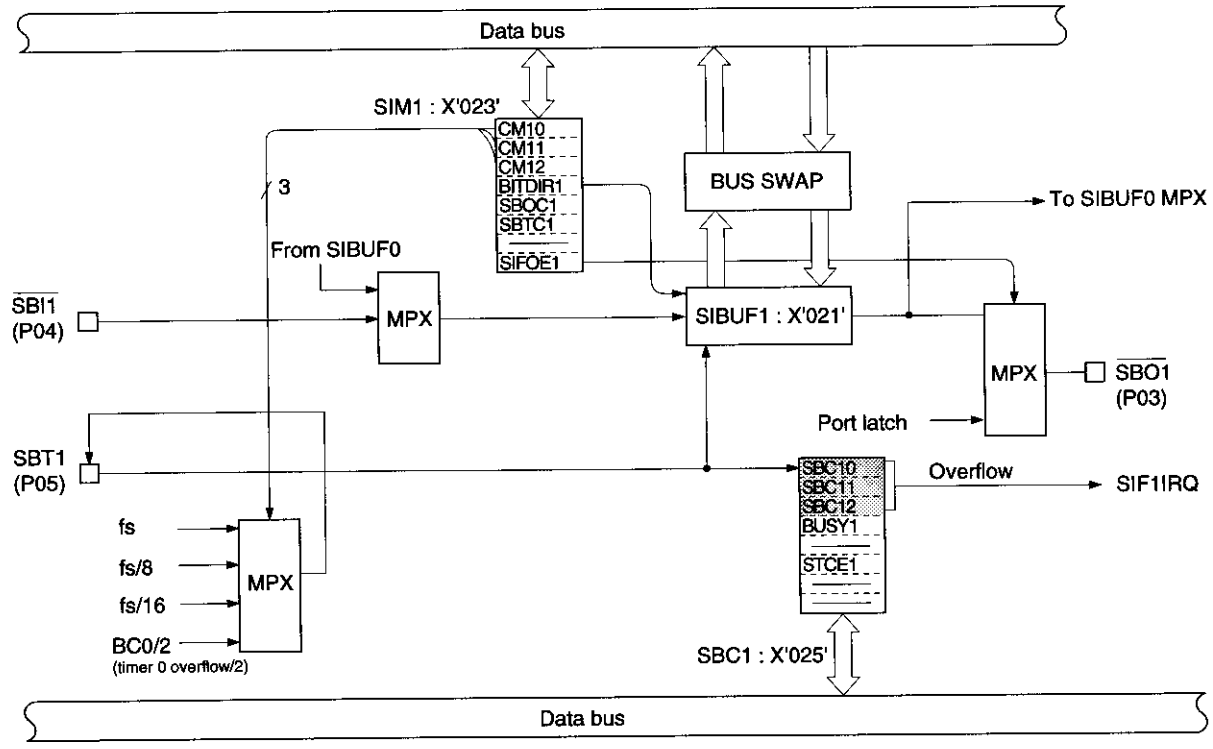


Figure 5-1-2 Serial Interface 1 Block Diagram

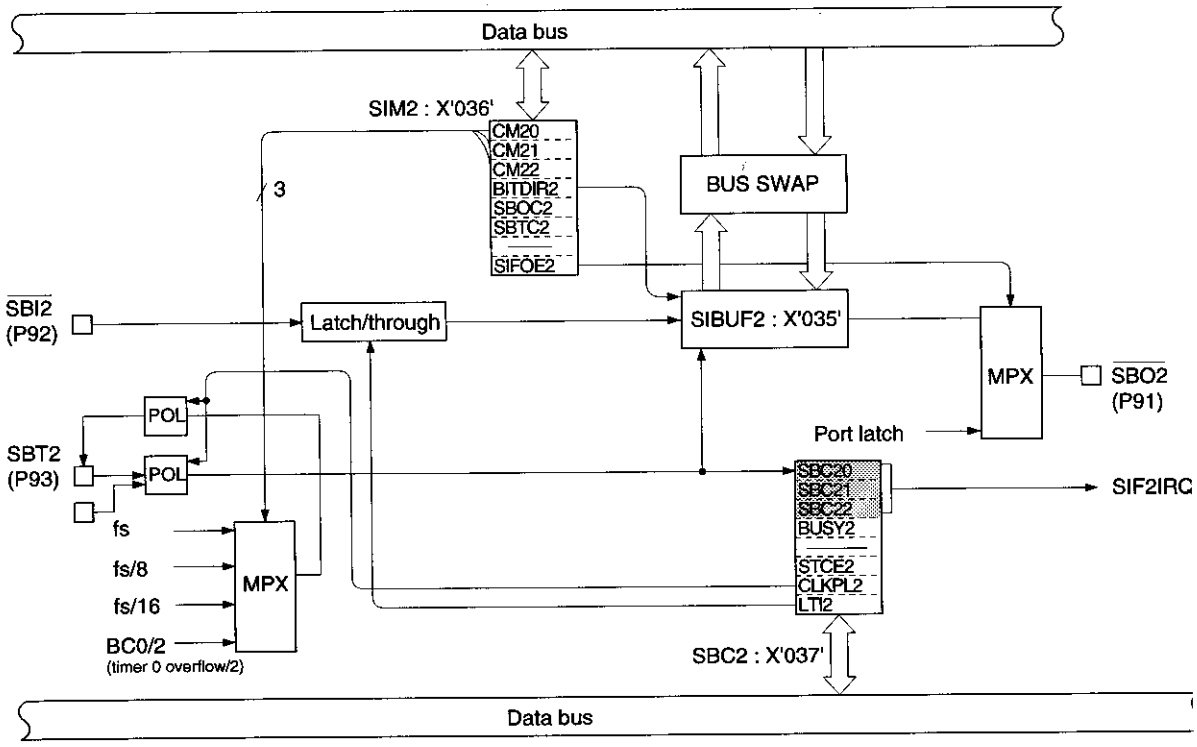


Figure 5-1-3 Serial Interface 2 Block Diagram

Transmit/receive shift buffers 0, 1, 2
(SIBUF0: X'020' R/W)
(SIBUF1: X'021' R/W)
(SIBUF2: X'035' R/W)

Serial interface bit counters 0, 1, 2
(SBC0: X'024' R/W)
Bit 5: STCE0
Bit 3: BUSY0
Bit 2: SBC02
Bit 1: SBC01
Bit 0: SBC00
(SBC1: X'025' R/W)
Bit 5: STCE1
Bit 3: BUSY1
Bit 2: SBC12
Bit 1: SBC11
Bit 0: SBC10
(SBC2: X'037' R/W)
Bit 5: STCE2
Bit 3: BUSY2
Bit 2: SBC22
Bit 1: SBC21
Bit 0: SBC20

When data is written into a transmit/receive shift buffer (SIBUF), if the SBT pin is idle, the $\overline{SBO}$ pin is goes low before the transfer. If the receiver is in the start condition enabled state (STCE bit = 0 in SBC), the serial bit counter is cleared by this signal.

Serial interface mode register 0
(SIM0: X'022' R/W)
Bit 7: SIFOE0
Bit 6: SIFCON0

Serial interface mode register 1
(SIM1: X'023' R/W)
Bit 7: SIFOE1

5-2 Serial Interface Operation

5-2-1 Overview

A serial interface transfer is started by writing data to the transmit/receive shift buffer (SIBUF). The 3-bit serial bit counter (SBC) is incremented each time a bit is transferred, and the transfer is terminated when the counter overflows (from 111 to 000).

With these serial interfaces, the serial bit counter (SBC) can be read and written to, enabling transfer of any number of bits. The number of bits must always be set before making a transfer. When n bits are to be transferred, the value 8–n is set in bits SBC2~SBC0 in the serial interface bit counter (SBC).

In reception using a start condition, however, bits SBC2~SBC0 are cleared when the start condition is input, and therefore 8-bit reception is performed automatically in this case.

5-2-2 Independent Operation Mode

In this mode, serial interfaces 0, 1, and 2 are used independently. Serial interfaces 0 and 1 are set to independent operation mode when the SIFCON0 bit in serial interface mode register 0 (SIM0) is cleared to 0. Serial interface 2 can only be used in independent operation mode.

Transmission/reception operations are described below.

(1) Transmission

① The SIFOE bit in the serial interface mode register (SIM) is set to 1 to select serial input/output.

② A value (the number of transmit bits) is set in bits SBC2~SBC0 in the serial interface bit counter (SBC). For n transmit bits, the set value is 8–n.

③ When data is written into the transmit/receive shift buffer (SIBUF), in the start condition enabled state the start condition is output and transfer is started. When data is transferred, the BUSY flag is set to 1.

④ The serial bit counter (bits SBC2~SBC0 of the serial interface bit counter (SBC)) is incremented each time a bit is transferred, and when the set number of bits have been transferred the BUSY flag is cleared to 0. At the same time, a serial interrupt (SIFIRQ) is generated and the serial interrupt request flag (IFSIF) in the interrupt request flag register (IF1) is set to 1.

Serial interface mode register 2
(SIM2: X'036' R/W)
Bit 7: SIFOE2

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 4: IFSIF2
Bit 3: IFSIF1
Bit 2: IFSIF0

(2) Reception

① The SIFOE bit in the serial interface mode register (SIM) is set to 1 to select serial input/output.

② Dummy data is written to the transmit/receive shift buffer (SIBUF).

③ In the start condition enabled state, when the start condition is input the BUSY flag is set and bits SBC2~SBC0 in the serial interface bit counter (SBC) are cleared.
In the start condition disabled state, a value (8–n in the case of n receive bits) is set in bits SBC2~SBC0 in the serial interface bit counter (SBC).

④ When the transfer clock is input, reception is started by performing transmit/receive shift buffer (SIBUF) shifting in synchronization with the clock. The BUSY flag is set to 1 when the start condition is input In the start condition enabled state, or when the external clock is input in the start condition disabled state.

⑤ The serial bit counter (bits SBC2~SBC0 of the serial interface bit counter (SBC)) is incremented each time a bit is transferred, and when the set number of bits have been transferred the BUSY flag is cleared to 0. At the same time, a serial interrupt (SIFIRQ) is generated and the serial interrupt request flag (IFSIF) is set to 1.

⑥ The contents of SIBUF are read in the interrupt handling routine.

Transmission/reception settings made in the serial interface mode register (SIM) and serial interface bit counter (SBC) are shown in table 5-2-1.

Table 5-2-1 Independent Operation Mode Transmission/Reception Settings

Register	Bits	Mnemonic	Function of Setting
SIM	0~2	CM0~CM2	Selects the transfer clock.
	3	BITDIR	Selects MSB-first or LSB-first transfer.
	4	SBOC	Selects the transmit data output pin ($\overline{SB0}$) output type.
	5	SBTC	Selects the clock input/output pin (SBT) output type.
	6	SIFCON0	Cleared to 0 to specify independent mode. (SIM0 only)
	7	SIFOE	Set to 1 to specify serial input/output.
SBC	0~2	SBC0~SBC2	For n transmit/receive bits, set with 8–n. (In the start condition enabled state, the number of receive bits need not be set.)
	5	STCE	Selects detection/ignoring of start condition.
	6	CLKPL	Selects serial interface 0 clock input/output polarity. (SBC0 and SBC2 only)
	7	LTI	Selects receive data input pin (SBI) through/latch input. (SBC0 and SBC2 only)

When a serial interface is used for transmission only, the start condition should be disabled (by setting the STCE bit to 1 in the serial interface bit counter (SBC)). If the start condition is enabled, incorrect operation may result from clearing of the serial bit counter in response to the start condition.

If a serial transfer is interrupted before completion, an interval of at least one serial clock cycle is necessary before the next transfer can be started.

5-2-3 Continuous Transmission/Reception Mode

In this mode, serial interfaces 0 and 1 are connected in parallel, and data transfer is performed continuously without stopping the clock.

Serial interfaces 0 and 1 are set to continuously transmission/reception mode when the SIFCON0 bit in serial interface mode register 0 (SIM0) is set to 1. The serial input/output pins used in this case are P00 ($\overline{SBO0}$), P01 (SBI0), P02 ($\overline{SBT0}$), and PSBT0 (P06).

Transmission/reception operations are described below.

(1) Transmission

① The SIFOE0 and SIFOE1 bits in serial interface mode registers 0 and 1 (SIM0, SIM1) are set to 1 to select serial input/output.

② A value (the number of transmit bits) is set in bits SBC2~SBC0 and SBC12~SBC10 in serial interface bit counters 0 and 1 (SBC0, SBC1).
For example, in the case of n transmit bits on serial interface 0, a value of 8–n is set in serial interface bit counter 0 (SBC0).

③ When data is written into the SIBUF0 and SIBUF1 transmit/receive shift buffers in that order, the start condition is output and transfer is started from SIBUF0. When data is transferred, the BUSY flag is set to 1.

④ The serial bit counter (bits SBC2~SBC0 of SBC0) is incremented each time a bit is transferred, and when the set number of bits have been transferred the BUSY flag is cleared to 0.
At the same time, a serial interrupt (SIFIRQ0) is generated and the serial interrupt 0 request flag (IFSIF0) is set to 1.

⑤ Next, the data in SIBUF1 is transmitted. When the transfer ends, a serial interface 1 interrupt (SIFIRQ1) is generated and the serial interrupt 1 request flag (IFSIF1) is set to 1.

⑥ When further continuous transfer is carried out, data is written to SBC0 and SIBUF0 during transmission of the SIBUF1 data.

Serial interface mode register 0
(SIM0: X'022' R/W)
Bit 7: SIFOE0
Bit 6: SIFCON0

Serial interface mode register 1
(SIM1: X'023' R/W)
Bit 7: SIFOE1

Serial interface bit counter 0
(SBC0: X'024' R/W)
Bit 3: BUSY0
Bit 2: SBC02
Bit 1: SBC01
Bit 0: SBC00

Serial interface bit counter 1
(SBC1: X'025' R/W)
Bit 3: BUSY1
Bit 2: SBC12
Bit 1: SBC11
Bit 0: SBC10

** 16 bits of data can be written simultaneously with the MOVL instruction.*

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 3: IFSIF1
Bit 2: IFSIF0

(2) Reception

① The SIFOE0 and SIFOE1 bits in serial interface mode registers 0 and 1 (SIM0, SIM1) are set to 1 to select serial input/output.

② Dummy data is written to transmit/receive shift buffer 0 (SIBUF0) in order to activate it.

Transmit/receive shift buffer 0 (SIBUF: X'020' R/W)

③ In the start condition enabled state, when the start condition is input the BUSY0 flag is set in serial interface bit counter 0 (SBC0) and bits SBC02~SBC00 are cleared. (Serial interface bit counter 1 (SBC1) is not affected.)
In the start condition disabled state, a value (8–n in the case of n receive bits) is set in bits SBC02~SBC00.

④ When the external clock is input, reception is started by shifting SIBUF0 in synchronization with the clock. The BUSY0 flag is set to 1 when the start condition is input In the start condition enabled state, or when the external clock is input in the start condition disabled state.

⑤ Serial bit counter 0 (bits SBC02~SBC00 of serial interface bit counter 0 (SBC0)) is incremented each time a bit is transferred, and when the set number of bits have been transferred the BUSY0 flag is cleared to 0.
At the same time, a serial interface 0 interrupt (SIFIRQ0) is generated and the serial interrupt 0 request flag (IFSIF0) is set to 1.

⑥ The contents of SIBUF0 are read in the interrupt handling routine.

⑦ By writing data to SIBUF1 while data is being received in SIBUF0, it is possible to receive data continuously without stopping the clock.

Transmission/reception settings made in the serial interface mode register (SIM) and serial interface bit counter (SBC), including the settings above, are shown in table 5-2-2.

Table 5-2-2 Continuous Transmission/Reception Mode Transmit/Receive Settings

Register	Bits	Mnemonic	Function of Setting
SIM0	0~2	CM00~CM02	Selects the transfer clock.
	3	BITDIR0	Selects MSB-first or LSB-first transfer.
	4	SBOC0	Selects the transmit data output pin ($\overline{SBO0}$) output type.
	5	SBTC0	Selects the external clock input/output pin (SBT0) output type.
	6	SIFCON0	Set to 1 to specify continuous transmission/ reception mode.
	7	SIFOE0	Set to 1 to specify serial input/output.
SIM1	3	BITDIR1	Selects MSB-first or LSB-first transfer.
	7	SIFOE1	Set to 1 to specify serial input/output.
SBC0	0~2	SBC00~SBC02	For n transmit/receive bits, set with 8–n. (In the start condition enabled state, the number of receive bits need not be set.)
	5	STCE0	Selects detection/ignoring of start condition.
	6	CLKPL0	Selects serial interface 0 clock input/output polarity.
	7	LTI0	Selects receive data input pin through/latch input.
SBC1	0~2	SBC10~SBC12	For n transmit/receive bits, set with 8–n. (In the start condition enabled state, the number of receive bits need not be set.)

When a serial interface is used for transmission only, the start condition should be disabled (by setting the STCE bit to 1 in the serial interface bit counter (SBC)). If the start condition is enabled, incorrect operation may result from clearing of the serial bit counter in response to the start condition.

If a serial transfer is interrupted before completion, an interval of at least one serial clock cycle is necessary before the next transfer can be started.

In the MN187XX23, when any of bits CLKPL0, LTI0, CM00~CM02, CM10~CM12, CLKPL2, LTI2, and CM20~CM22 in serial interface mode registers 0, 1, and 2 (SIM0, SIM1, SIM2) is manipulated, the level of the SBT0, SBT1, and SBT2 pins is inverted, and this inversion may cause the serial interface bit counter to be incremented. To prevent this incorrect operation, the following measures should be taken when manipulating any of these bits.

Example 1:
Clear the SIFOE0, SIFOE1, and SIFOE2 bits in serial interface mode registers 0, 1, and 2 (SIM0, SIM1, SIM2) to 0 before setting of any of the above control bits. When the SIFOE0, SIFOE1, and SIFOE2 bits are cleared to 0 (designating that serial pins are to function as ports) the serial interface bit counters are stopped, and so cannot be incremented.

Example 2:
Clear the serial interface bit counter after manipulating any of the above control bits (by writing 0 in bits 0~2 of serial interface bit counter 0/1/2 (SBC0/SBC1/SBC2).

5-2-4 Serial Interface Transfer Timing

Serial interface bit counter 0
(SBC0: X'024' R/W)
Bit 7: LTI0
Bit 6: CLKPL0

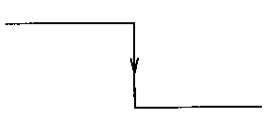
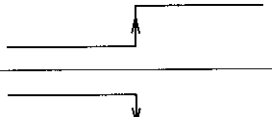
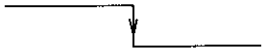
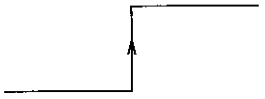
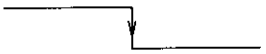
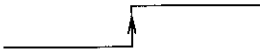
Serial interface bit counter 2
(SBC2: X'037' R/W)
Bit 7: LTI2
Bit 6: CLKPL2

The clock polarity and receive data latching edge can be controlled for serial interfaces 0 and 2 by means of bits CLKPL and LTI in the serial interface bit counter (SBC).

In transmission, transmit data is output in synchronization with the falling edge of the clock when the CLKPL bit is cleared to 0, and in synchronization with the rising edge when the CLKPL bit is set to 1.

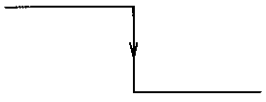
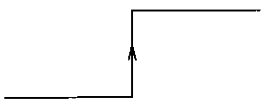
In reception, receive data is latched in synchronization with an edge of opposite polarity to the output edge in transmission when the LTI bit is cleared to 0, and in synchronization with an edge of the same polarity as the output edge in transmission when the LTI bit is set to 1.

Table 5-2-3 Serial Data Output Edge and Input Edge (Serial Interfaces 0 and 2)

CLKPL	LTI	Transmit Data Output Edge	Receive Data Input Edge
0	0		
	1		
1	0		
	1		

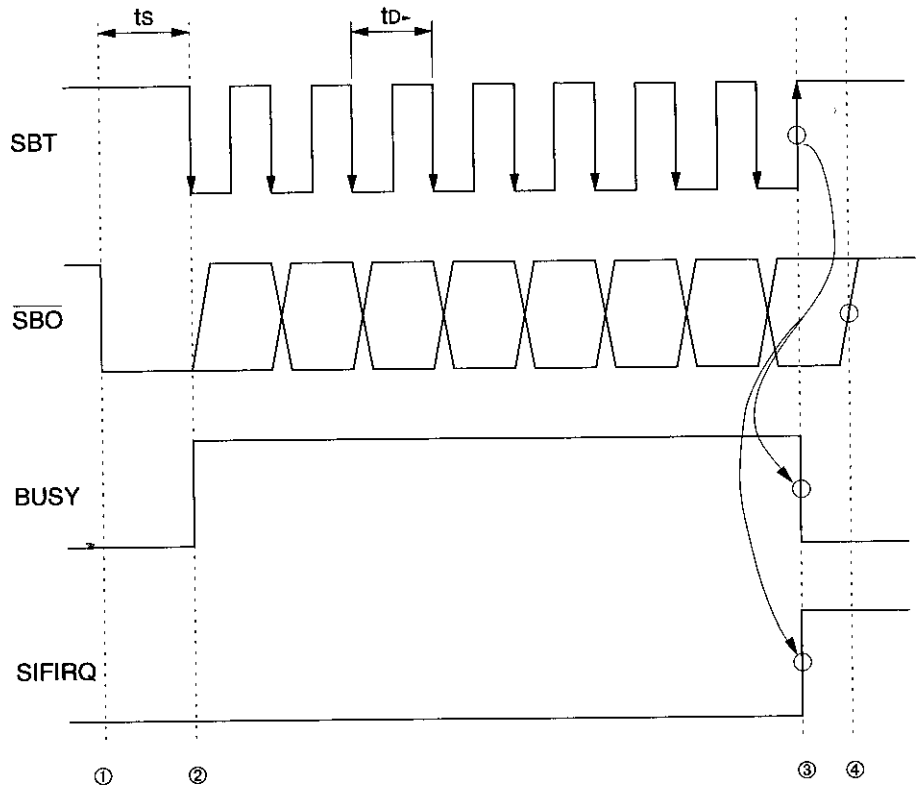
On serial interface 1, in transmission, transmit data is output on the fall of the clock, and in reception, receive data is latched on the rise of the clock.

Table 5-2-4 Serial Data Output Edge and Input Edge (Serial Interface 1)

Transmit Data Output Edge	Receive Data Input Edge
	

Transmission and reception timing is shown in figures 5-2-1 to 5-2-4. The number of clocks differs only in continuous transmission/reception mode and when transferring an arbitrary number of bits, and is the same in other operations.

(1) Transmit timing



1/2tD < ts < tD
tD ≥ 1 µs:
with internal clock
tD ≥ 500 ns:
with external clock

↑
Numbers correspond to those below

① When transmit data is written to the shift buffer, a start condition is output.

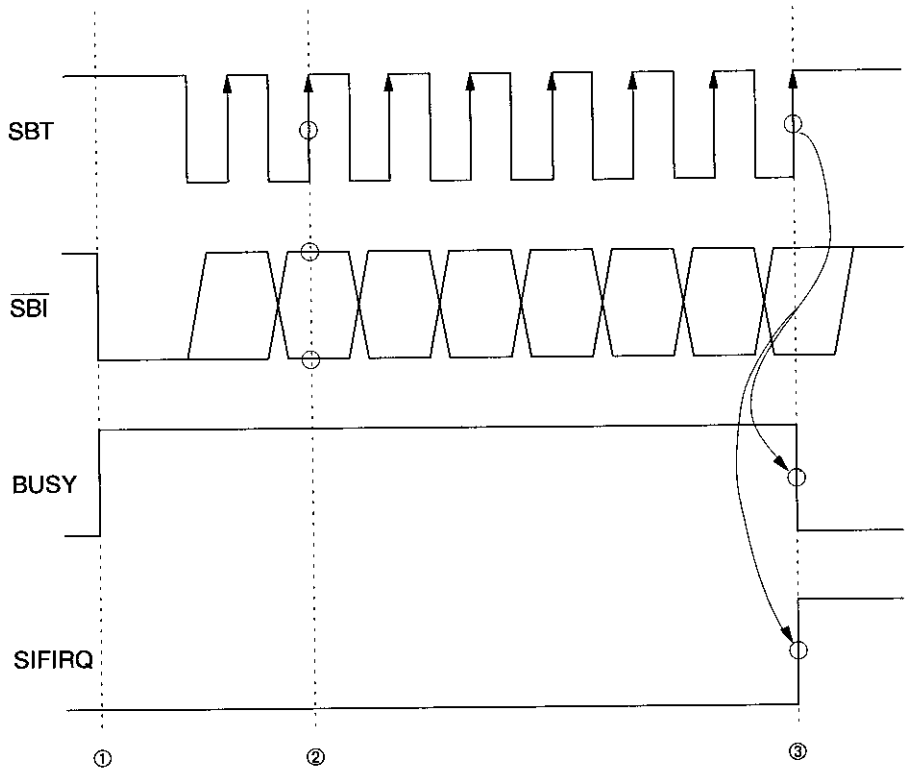
② The transmit data is output sequentially on the falling edge of the clock.

③ On the 8th rising edge of the clock an interrupt is generated and the BUSY flag is reset.

④ If the start condition is enabled, the $\overline{SBO}$ pin goes high 3.8 µs later (at 4.19MHz operation).

Figure 5-2-1 Transmit Timing (CLKPL=0)

(2) Receive timing (start condition, input through)



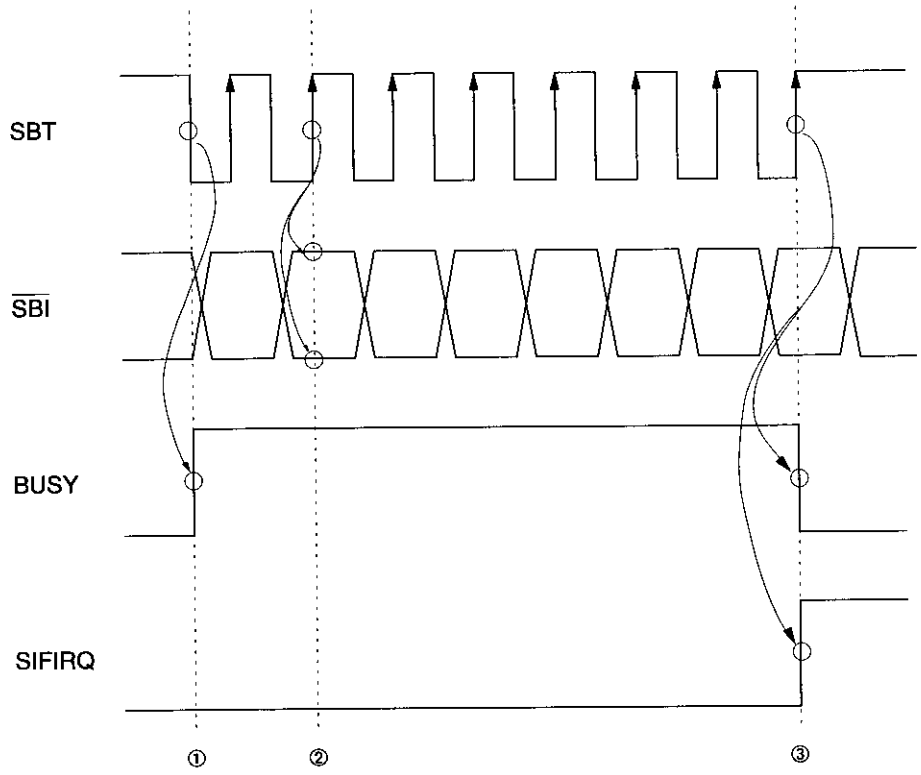
① If the start condition is enabled, the BUSY flag is set when the start condition is received.

② The receive data is latched sequentially into the shift buffer on the rising edge of the clock.

③ On the 8th rising edge of the clock an interrupt is generated and the BUSY flag is reset.

Figure 5-2-2 Receive Timing (STCE=0, LTI=0, CLKPL=0)

(3) Receive timing (no start condition, input through)



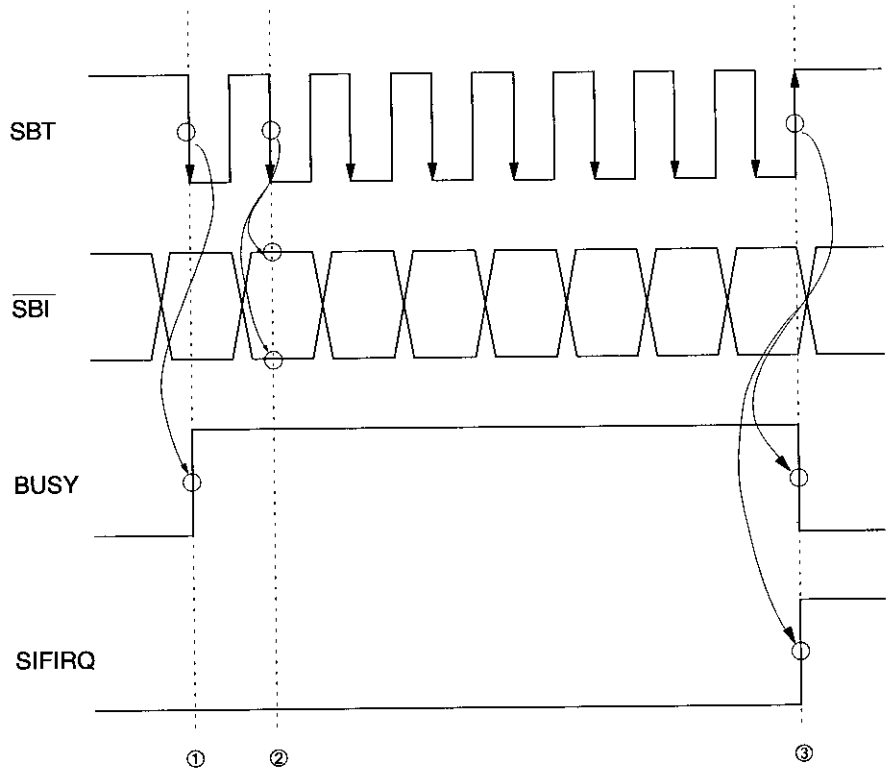
① The BUSY flag is set on the fall of the clock.

② The receive data is latched sequentially into the shift buffer on the rising edge of the clock.

③ On the 8th rising edge of the clock an interrupt is generated and the BUSY flag is reset.

Figure 5-2-3 Receive Timing (STCE=1, LTI=0, CLKPL=0)

(4) Receive timing (no start condition, input latch)



① The BUSY flag is set on the fall of the clock.

② The receive data is latched sequentially into the shift buffer on the falling edge of the clock.

③ On the 8th rising edge of the clock an interrupt is generated and the BUSY flag is reset.

Figure 5-2-4 Receive Timing (STCE=1, LTI=1, CLKPL=0)

5-3 Interrupts

When bits SBC02~00/SBC12~10/SBC22~20 of a serial interface bit counter (SBC0/SBC1/SBC2) overflow (from 111 to 000), the serial interrupt request flag (bit IFSIF0/IFSIF1/IFSIF2) in interrupt request flag register 1 (IF1) set to 1. If the serial interrupt enable flag (bit IESIF0/IESIF1/IESIF2) in interrupt enable flag register 1 (IE1) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

Serial interface bit counter 0
(SBC0: X'024' R/W)
Bit 2: SBC02
Bit 1: SBC01
Bit 0: SBC00

Serial interface bit counter 1
(SBC1: X'025' R/W)
Bit 2: SBC12
Bit 1: SBC11
Bit 0: SBC10

Serial interface bit counter 2
(SBC2: X'037' R/W)
Bit 2: SBC22
Bit 1: SBC21
Bit 0: SBC20

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 4: IFSIF2
Bit 3: IFSIF1
Bit 2: IFSIF0

Interrupt enable flag register 1
(IE1: X'015' R/W)
Bit 4: IESIF2
Bit 3: IESIF1
Bit 2: IESIF0

5-4 Serial Interface Control Registers

5-4-1 Overview

Nine register bytes are used to control the MN187XX23's serial interfaces: serial interface mode registers (SIM0, SIM1, SIM2) and serial interface bit counters (SBC0, SBC1, SBC2) which control the serial interface operating mode, and transmit/receive shift buffers (SIBUF0, SIBUF1, SIBUF2) used to send and receive transfer data. These registers are all located in RAM space.

Table 5-4-1 Serial Interface Registers

Register Abbreviation	RAM Address	R/W	Register Name
SIBUF0	X'020'	R/W	Transmit/receive shift buffer 0
SIBUF1	X'021'	R/W	Transmit/receive shift buffer 1
SIBUF2	X'035'	R/W	Transmit/receive shift buffer 2
SIM0	X'022'	R/W	Serial interface mode register 0
SIM1	X'023'	R/W	Serial interface mode register 1
SIM2	X'036'	R/W	Serial interface mode register 2
SBC0	X'024'	R/W	Serial interface bit counter 0
SBC1	X'025'	R/W	Serial interface bit counter 1
SBC2	X'037'	R/W	Serial interface bit counter 2

5-4-2 Transmit/Receive Shift Buffers

The transmit/receive shift buffers are 8-bit read/write registers that shift transmit/receive data. LSB-first or MSB-first can be selected as the transfer direction. [☞ BITDIR0 bit, BITDIR1 bit, BITDIR2 bit in "5-4-4 Serial Interface Mode Registers"]

(1) Transmit/receive shift buffer 0 (SIBUF0)

	7	6	5	4	3	2	1	0	
SIBUF0	SIBUF07	SIBUF06	SIBUF05	SIBUF04	SIBUF03	SIBUF02	SIBUF01	SIBUF00	(After reset: XXXXXXXX)

Figure 5-4-1 Transmit/Receive Shift Buffer 0 (SIBUF0: X'020' R/W)

After a reset, all bits are undefined.

(2) Transmit/receive shift buffer 1 (SIBUF1)

	7	6	5	4	3	2	1	0	
SIBUF1	SIBUF17	SIBUF16	SIBUF15	SIBUF14	SIBUF13	SIBUF12	SIBUF11	SIBUF10	(After reset: XXXXXXXX)

Figure 5-4-2 Transmit/Receive Shift Buffer 1 (SIBUF1: X'021' R/W)

After a reset, all bits are undefined.

(3) Transmit/receive shift buffer 2 (SIBUF2)

	7	6	5	4	3	2	1	0	
SIBUF2	SIBUF27	SIBUF26	SIBUF25	SIBUF24	SIBUF23	SIBUF22	SIBUF21	SIBUF20	(After reset: XXXXXXXX)

Figure 5-4-3 Transmit/Receive Shift Buffer 2 (SIBUF2: X'035' R/W)

In a reset, all bits are undefined.

5-4-3 Serial Interface Bit Counters

The serial interface bit counters control the number of transfer bits, start condition enabling, transmit/receive clock polarity, and receive data input latch/through operation. Serial interface bit counters 0 and 2 (SBC0, SBC2) are 7-bit read/write registers, and serial interface bit counter 1 (SBC1) is a 5-bit read/write register (however, bit 3 (the BUSY flag) is a read-only bit in all these registers). Bits 2~0 are used to count transmit/receive clock cycles.

(1) Serial interface bit counter 0 (SBC0)

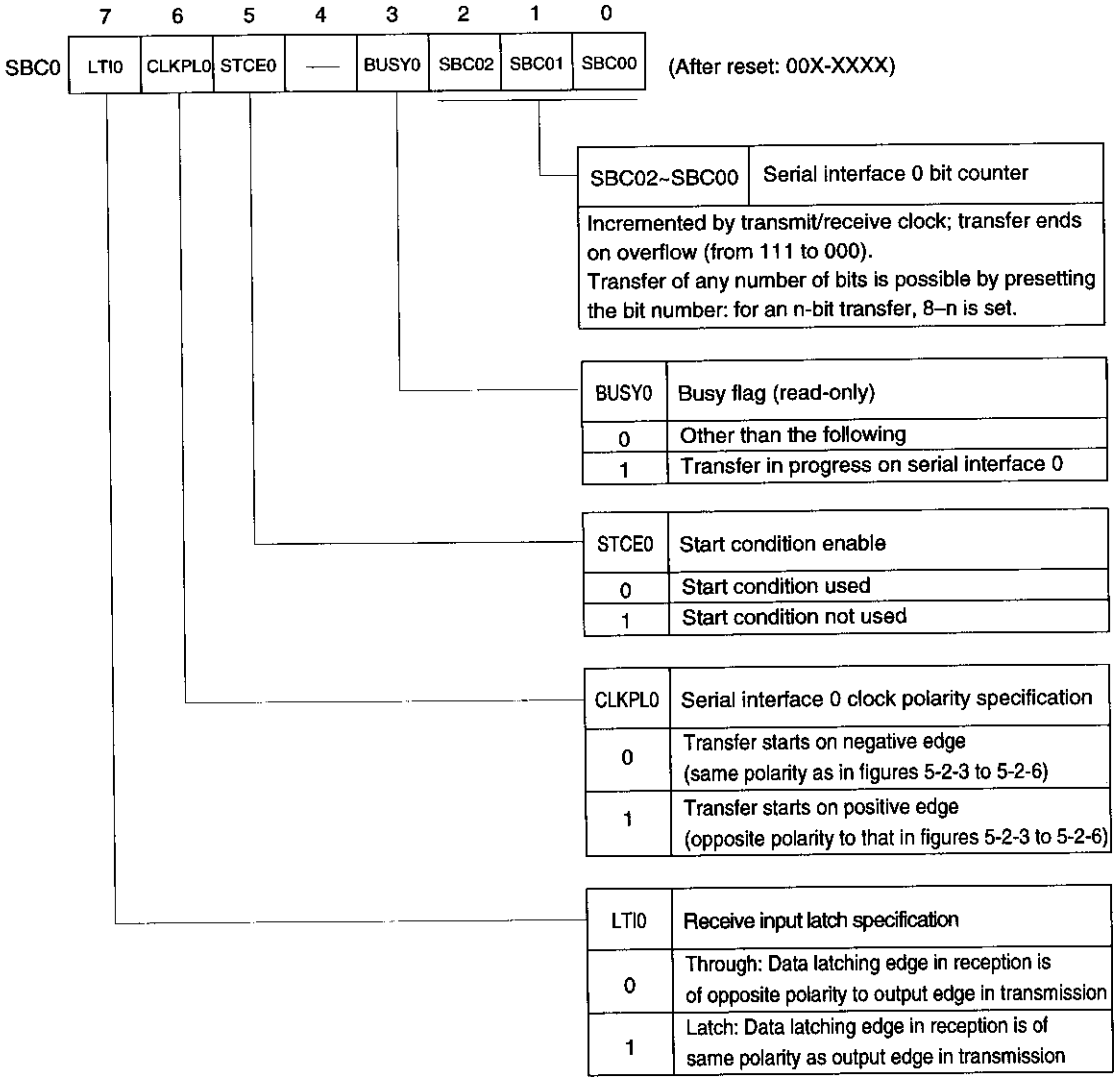


Figure 5-4-4 Serial Interface Bit Counter 0 (SBC0: X'024' R/W*)

* Bit 3 (BUSY0) only is read-only.

In a reset, bits LTI0 and CLKPL0 are cleared to 0, and the other bits are undefined.

(2) Serial interface bit counter 1 (SBC1)

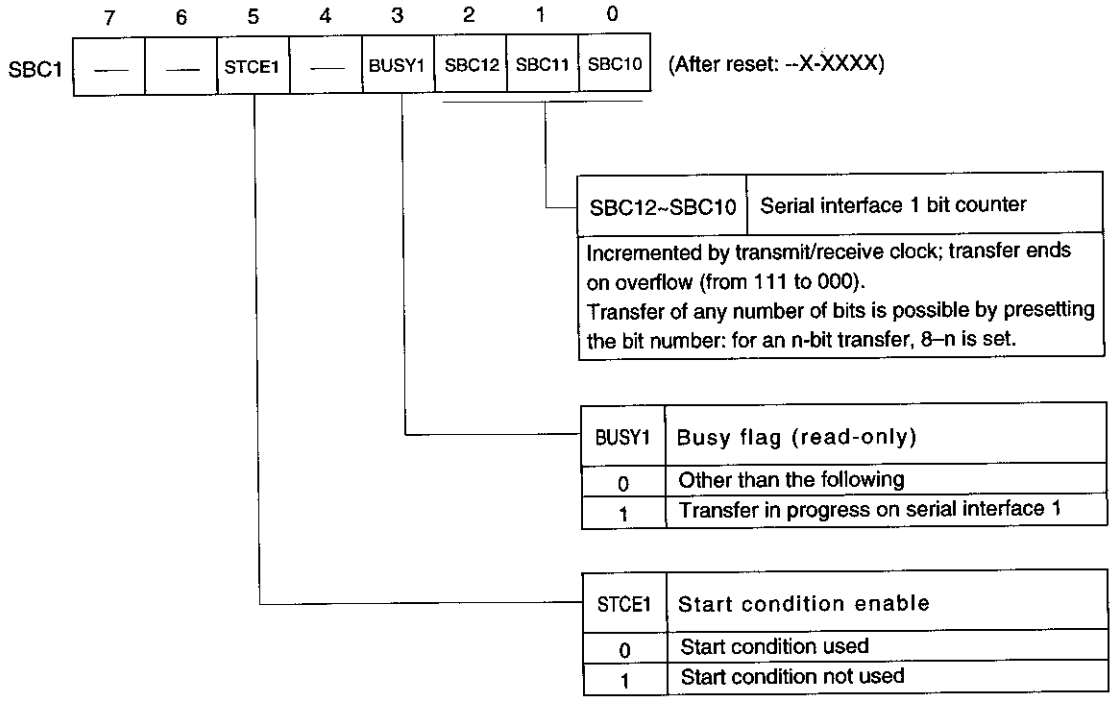
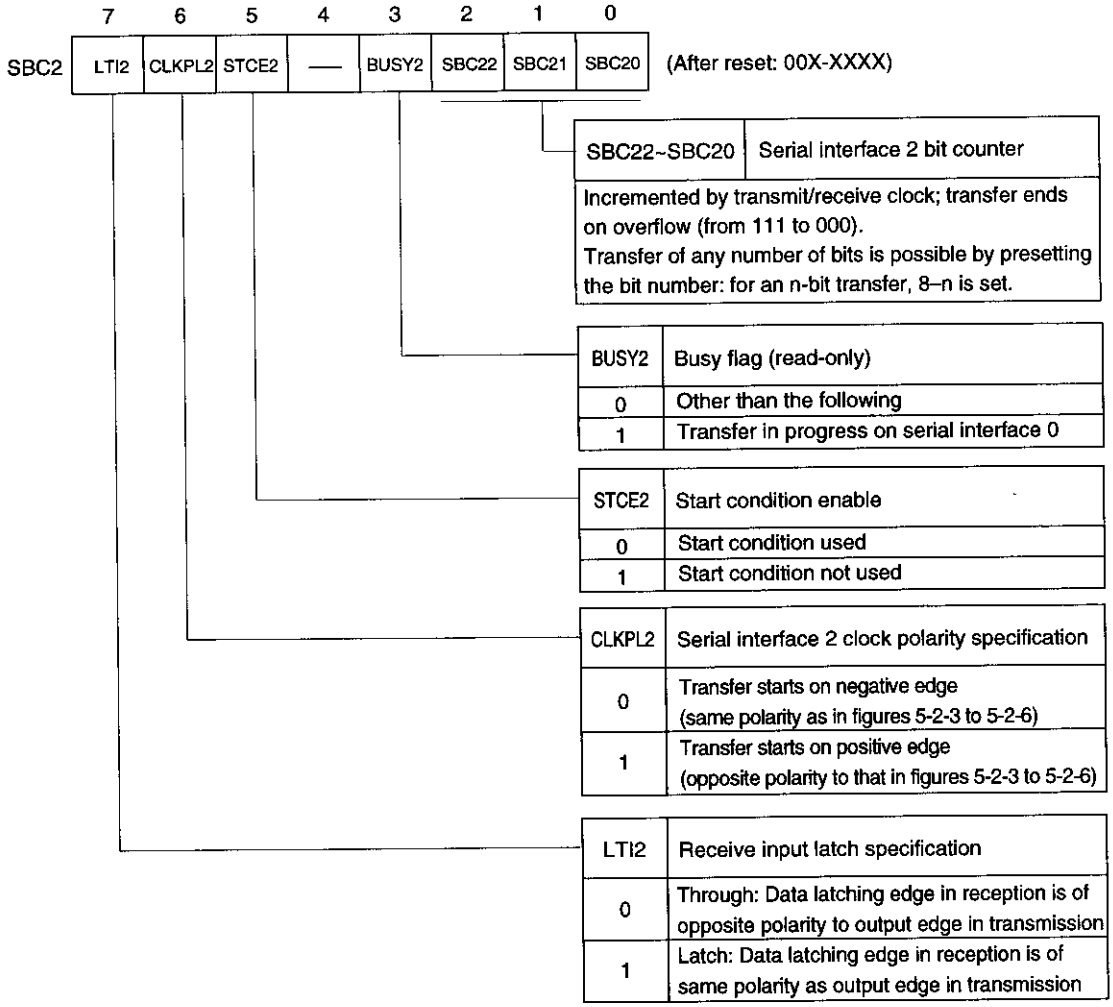


Figure 5-4-5 Serial Interface Bit Counter 1 (SBC1: X'025' R/W*)

* Bit 3 (BUSY1) only is read-only.

In a reset, all bits are undefined.

(3) Serial interface bit counter 2 (SBC2)



* Bit 3 (BUSY2) only is read-only.

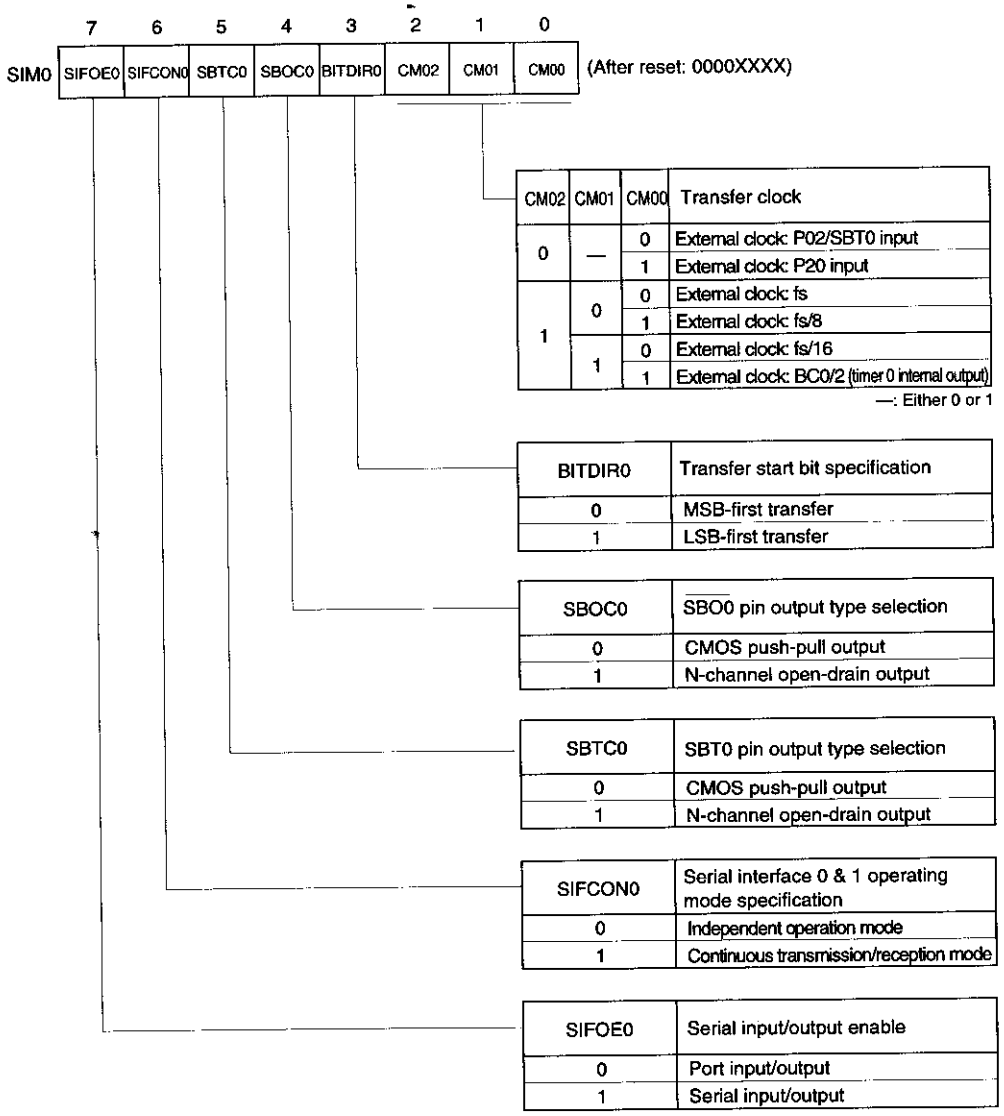
Figure 5-4-6 Serial Interface Bit Counter 2 (SBC2: X'037' R/W*)

In a reset, bits LTI2 and CLKPL2 are cleared to 0, and the other bits are undefined.

5-4-4 Serial Interface Mode Registers

The serial interface mode registers are used to specify the transfer clock, select MSB-first or LSB-first transfer, select the transmit data and clock output form, specify the operating mode for serial interfaces 0 and 1, and enable or disable serial interface 0 and 1 operation. Serial interface mode register 0 (SIM0) is an 8-bit read/write register, and serial interface mode registers 1 and 2 (SIM1, SIM2) are 7-bit read/write registers.

(1) Serial interface mode register 0 (SIM0)



When SIFOE0 is set to 1, specifying serial input/output, the P01 (SBI0) pin can be used as a port when serial interface 0 is used only for transmission, but the P00 (SBO0) pin cannot be used as a port when serial interface 0 is used only for reception.

When the P06 (PSBT0) pin is used for external clock input, clock output cannot be performed from the P02 (SBT0) pin.

Figure 5-4-7 Serial Interface Mode Register 0 (SIM0: X'022' R/W)

In a reset, bits SIFOE0, SIFCON0, SBTC0, and SBOC0 are cleared to 0, and the other bits are undefined.

☞ For details of timer 0 internal outputs SIF0/SIF1/SIF2, see "4-2-2 (4) Serial clock generation."

Serial interface 0 does not operate when the SIFOE0 bit is cleared to 0. In this case, pins P00 (SBO0), P01 (SBI0), P02 (SBT0), and P06 (PSBT0) function as ordinary I/O ports.

(2) Serial interface mode register 1 (SIM1)

☞ For details of timer 0 internal outputs SIF0/ SIF1/SIF2, see "4-2-2 (4) Serial clock generation."

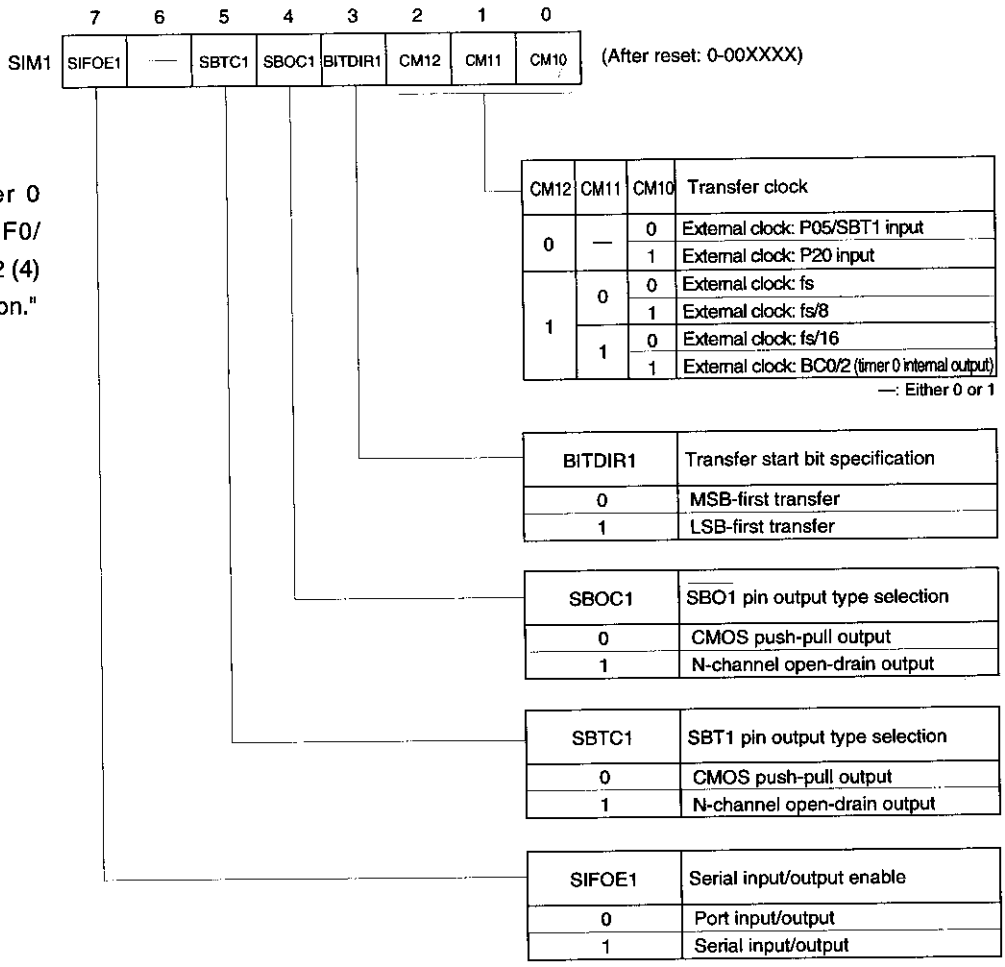


Figure 5-4-8 Serial Interface Mode Register 1 (SIM1: X'023' R/W)

In a reset, bits SIFOE1, SBTC1, and SBOC1 are cleared to 0, and the other bits are undefined.

When SIFOE1 is set to 1, specifying serial input/output, the P04 ($\overline{SBI1}$) pin can be used as a port when serial interface 1 is used only for transmission, but the P03 ($\overline{SBO1}$) pin cannot be used as a port when serial interface 1 is used only for reception.

Serial interface 1 does not operate when the SIFOE1 bit is cleared to 0. In this case, pins P03 ($\overline{SBO1}$), P04 ($\overline{SBI1}$), and P05 (SBT1) function as ordinary I/O ports.

(3) Serial interface mode register 2 (SIM2)

☞ For details of timer 0 internal outputs SIF0/ SIF1/SIF2, see "4-2-2 (4) Serial clock generation."

	7	6	5	4	3	2	1	0	
SIM2	SIFOE2	—	SBTC2	SBOC2	BITDIR2	CM22	CM21	CM20	(After reset: 0-00XXXX)

CM22	CM21	CM20	Transfer clock
0	—	0	External clock: P05/SBT1 input
		1	External clock: P20 input
1	0	0	External clock: fs
		1	External clock: fs/8
	1	0	External clock: fs/16
		1	External clock: BC0/2 (timer 0 internal output)

—: Either 0 or 1

BITDIR2	Transfer start bit specification
0	MSB-first transfer
1	LSB-first transfer

SBOC2	$\overline{\text{SBO2}}$ pin output type selection
0	CMOS push-pull output
1	N-channel open-drain output

SBTC2	SBT2 pin output type selection
0	CMOS push-pull output
1	N-channel open-drain output

SIFOE2	Serial input/output enable
0	Port input/output
1	Serial input/output

Figure 5-4-9 Serial Interface Mode Register 2 (SIM2: X'036' R/W)

In a reset, bits SIFOE2, SBTC2, and SBOC2 are cleared to 0, and the other bits are undefined.

Serial interface 2 does not operate when the SIFOE2 bit is cleared to 0. In this case, pins P91 ($\overline{\text{SBO2}}$), P92 ($\overline{\text{SBI2}}$), and P93 (SBT2) function as ordinary I/O ports.

When SIFOE2 is set to 1, specifying serial input/output, the P92 ($\overline{\text{SBI2}}$) pin can be used as a port when serial interface 2 is used only for transmission, but the P91 ($\overline{\text{SBO2}}$) pin cannot be used as a port when serial interface 2 is used only for reception.

5-5 Examples of Use of Serial Interface

5-5-1 Communication with MN1870 Series/MN1880 Series

Pin connections for communication with the MN1870 and MN1880 Series are shown below.

Table 5-5-1 Connection to MN1870 Series/MN1880 Series

MN187XX23	MN1870/MN1880 Series
SBT	SBT
$\overline{SBO}$	$\overline{SBI}$
$\overline{SBI}$	$\overline{SBO}$

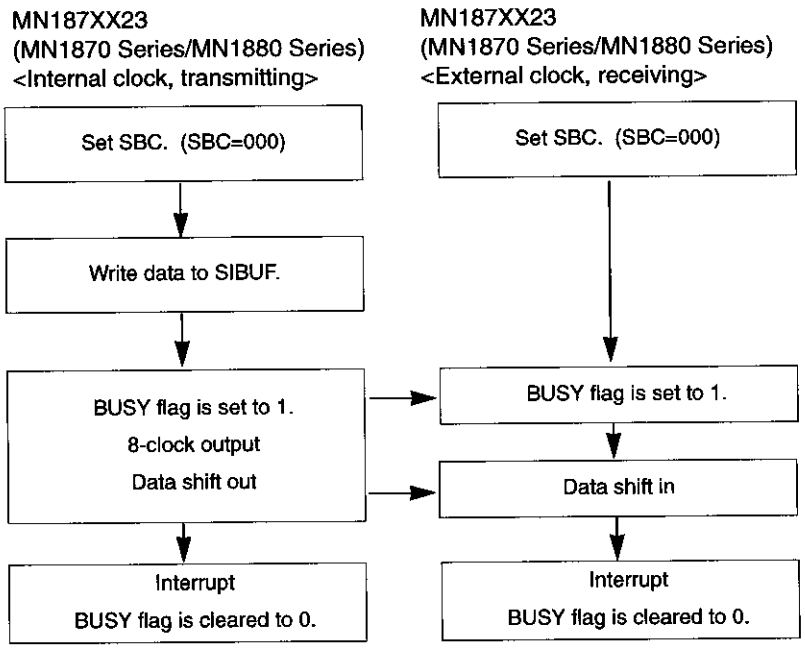


Figure 5-5-1 Sequence for Communication with MN1870 Series/MN1880 Series

5-5-2 Communication with MN1500 Series

Pin connections for communication with the MN1500 Series are shown below.

Table 5-5-2 Connection to MN1500 Series

MN187XX23	MN1500 Series
SBT	SBT
$\overline{\text{SBO}}$	$\overline{\text{SBI}}$
$\overline{\text{SBI}}$	$\overline{\text{SBO}}$

The MN1500 Series serial interface has the following features.

① Transmit/receive data is handled with negative logic (as in the MN187XX23).

② Transmit data is shifted out on the falling edge of the clock, and receive data is shifted in on the rising edge of the clock. (MN187XX23 serial interface bit counter 0 (SBC0) settings: CLKPL bit=0, LTI bit=0)

③ MSB-first transfer (MN187XX23 serial interface mode register (SIM) setting: BITDIR bit=0)

④ Continuous transmission/reception mode is not supported.

⑤ Start condition output/detection is not performed.

⑥ Arbitrary bit-length transfer is not supported (8-bit transfer is used).

⑦ There is no BUSY flag. Therefore, communication is performed in independent operation mode with the MN187XX23 using the above settings.

Serial interface bit counters
(SBC0: X'024' R/W)
Bit 7: LTI0
Bit 6: CLKPL0
(SBC2: X'037' R/W)
Bit 7: LTI2
Bit 6: CLKPL2

Serial interface mode registers
(SIM0: X'022' R/W)
Bit 3: BITDIR0
(SIM1: X'023' R/W)
Bit 3: BITDIR1
(SIM2: X'036' R/W)
Bit 3: BITDIR2

Serial interface bit counters
(SBC0: X'024' R/W)
Bit 3: BUSY0
Bit 2: SBC02
Bit 1: SBC01
Bit 0: SBC00
(SBC1: X'025' R/W)
Bit 3: BUSY1
Bit 2: SBC12
Bit 1: SBC11
Bit 0: SBC10
(SBC2: X'037' R/W)
Bit 3: BUSY2
Bit 2: SBC22
Bit 1: SBC21
Bit 0: SBC20

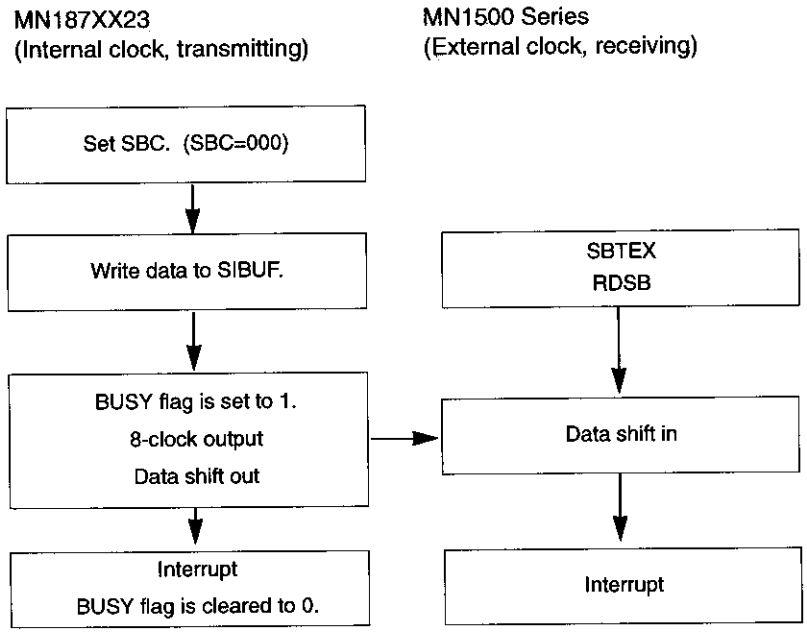


Figure 5-5-2 Sequence for Communication with MN1500 Series (1)

Transmit/receive shift buffers
(SIBUF0: X'020' R/W)
(SIBUF1: X'021' R/W)
(SIBUF2: X'035' R/W)

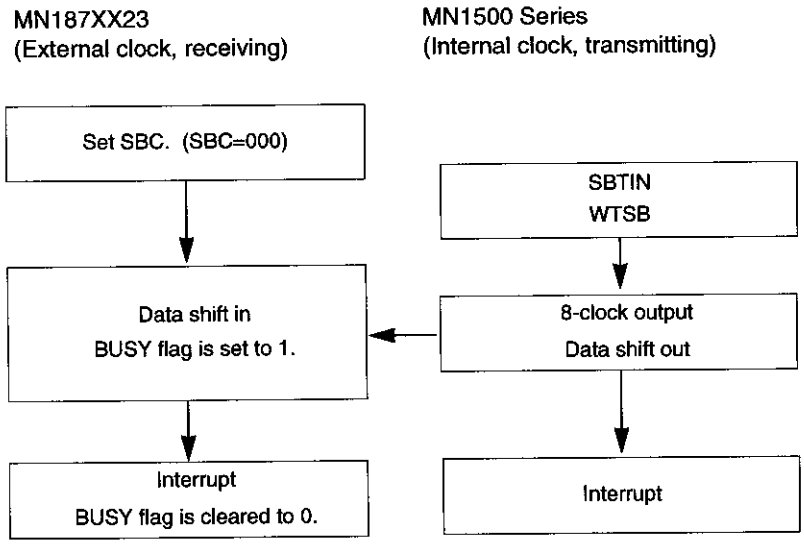
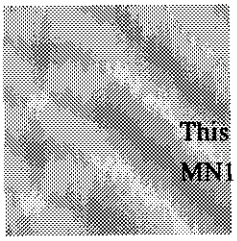


Figure 5-5-3 Sequence for Communication with MN1500 Series (2)

Chapter 6 Display Functions 6



This chapter describes the operation of the MN187XX23's FLP control circuit.

6-1 Overview

The MN187XX23 can automatically control a fluorescent light panel (FLP) display of up to 16 segments × up to 16 digits. Hardware includes 32 port pins (ports P5 to P8) with direct FLP driving, and key-scan interrupts that indicate key-scan timing to permit key-scanning using the segment pins.

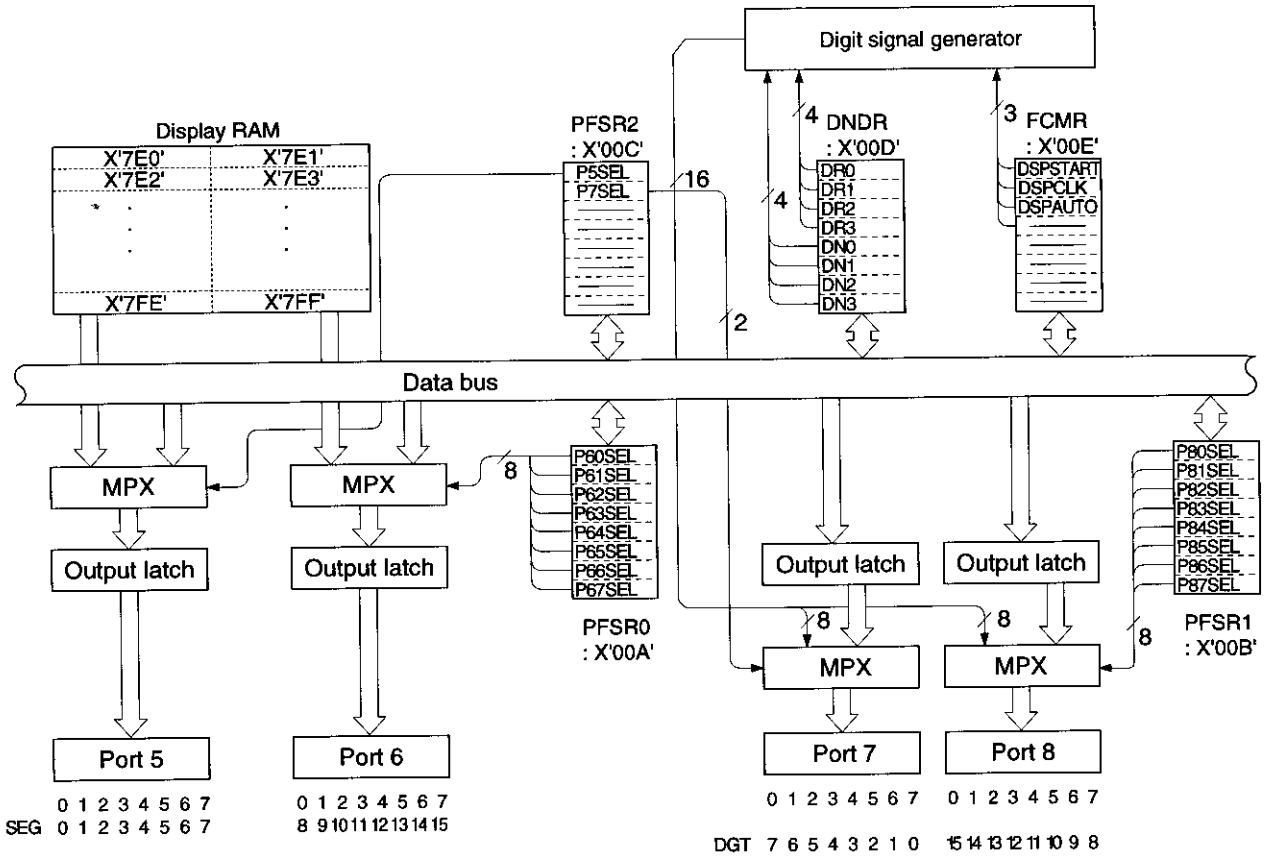


Figure 6-1-1 FLP Block Diagram

6-2 FLP Control Circuit Operation

6-2-1 Overview

FLP display is controlled by means of digit output and segment output.

The register setting procedure is as follows.

Port/FLP select register 0 (PFSR0: X'00A' R/W)

Port/FLP select register 1 (PFSR1: X'00B' R/W)

Port/FLP select register 2 (PFSR2: X'00C' R/W)

Digit number/dimmer register (DNDR: X'00D' R/W)
Bit 7: DN3
Bit 6: DN2
Bit 5: DN1
Bit 4: DN0
Bit 3: DR3
Bit 2: DR2
Bit 1: DR1
Bit 0: DR0

FLP clock mode register (FCMR: X'00E' R/W)
Bit 2: DSPAUTO
Bit 1: DSPCLK
Bit 0: DSPSTART

① Set the digit pins and segment pins to be used by means of port/FLP select register 0 (PFSR0), port/FLP select register 1 (PFSR1), and port/FLP select register 2 (PFSR2). Set the number of pin digits with bits DN0~DN3, and the digit waveform with bits DR0~DR3, in the digit number/dimmer register (DNDR).
[☞ "6-2-2 FLP Pin (Digit/Segment) Setting"]

② Set the segment data in the FLP display RAM (X'7E0'~X'7FF').
[☞ "6-2-3 FLP Display Data Setting"]

③ Select the display time per digit with the DSPCLK bit, and serial display/automatic display repetition with the DSPAUTO bit, in the FLP clock mode register (FCMR), and start display by setting the DSPSTART bit to 1.
Within the set digit pins DGT0~DGTn, display is performed in order from the highest pin, DGTn, through DGTn–1, etc., to DGT0. When one display cycle (the DGTn~DGT0 display cycle) is completed, a key-scan interrupt (KYSIRQ) is generated. [☞ "6-2-4 FLP Display Timing Control"]

④ After key-scan interrupt handling has been executed, if automatic display repetition has been specified, display continues again from DGTn. If serial display has been specified, display starts again when the DPSTART bit is set to 1. [☞ "6-2-5 Key-Scan Interrupt"]

6-2-2 FLP Pin (Digit/Segment) Setting

(1) Digit and segment assignment to ports

FLP pins (digit/segment) are assigned to ports as shown in figure 6-2-1.

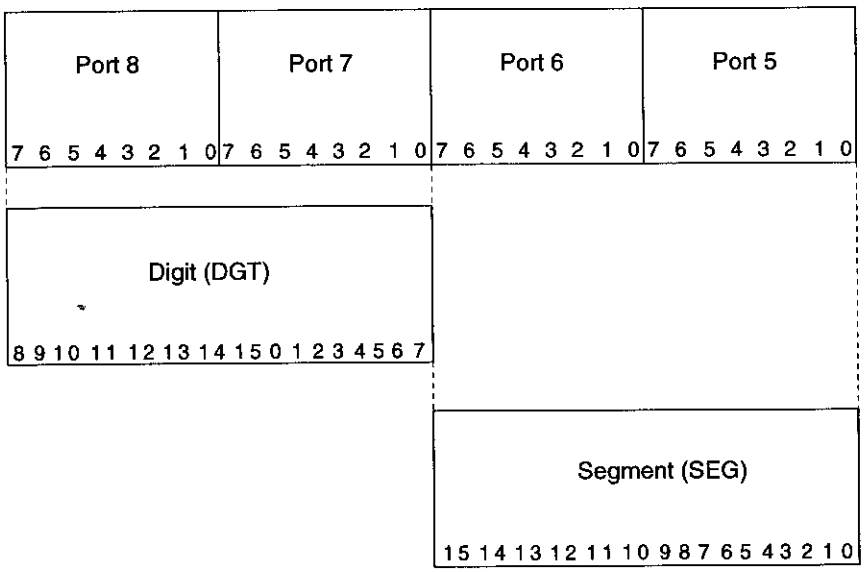


Figure 6-2-1 FLP (Digit/Segment) Pin Port Assignments

1. Port 5 and 7 pins can be switched between port and FLP functions in byte units.

2. Port 6 and 8 pins can be switched between port and FLP functions in bit units.

Port 5 and 6 pins are assigned as segment output pins, and port 7 and 8 pins as digit output pins. These FLP pins are all high-voltage output pins, and the port 7 and 8 digit output pins are large-current output pins.

Port/FLP select register 0 (PFSR0: X'00A' R/W)

Port/FLP select register 1 (PFSR1: X'00B' R/W)

Port/FLP select register 2 (PFSR2: X'00C' R/W)

Digit number/dimmer register (DNDR: X'00D' R/W)

(2) Setting the number of segments and digits

Four registers are used in setting the digit output pins and segment output pins: port/FLP select register 0 (PFSR0), port/FLP select register 1 (PFSR1), port/FLP select register 2 (PFSR2), and the digit number/dimmer register (DNDR).

[☞ "6-3-2 Port/FLP Select Registers"]
[☞ "6-3-3 Digit Number/Dimmer Register (DNDR)"]

The register setting procedure is as follows.

① Make FLP settings in port/FLP select register 0 (PFSR0), port/FLP select register 1 (PFSR1), and port/FLP select register 2 (PFSR2), according to the number of FLP pins (digit/segment) to be used. Settings should be made so that the numbers (DGT0~DGTn) of the digit pins used are consecutive.

② Set the number of digit outputs (1~16) in bits DN3~DN0 of the digit number/dimmer register (DNDR). The set value is the number of digits – 1.

It is useful to remember that the value of the largest number, n, among the digit pins used (DGT0~DGTn) is set in DN3~DN0 of DNDR.

By means of port/FLP select register 0 (PFSR0), port/FLP select register 1 (PFSR1), and port/FLP select register 2 (PFSR2), pins not used as FLP pins can be used as general-purpose ports.

Port 5	High-voltage output port (P-channel open-drain output, incorporating pull-down resistor between pin and VPP pin)
Port 6	High-voltage output port (P-channel open-drain output, with pull-down resistor between pin and VPP pin selectable as a mask option)
Port 7	High-voltage output port (P-channel open-drain output, with pull-down resistor between pin and VPP pin selectable as a mask option)
Port 8	High-voltage output port (P-channel open-drain output, incorporating pull-down resistor between pin and VPP pin)

6-2-3 FLP Display Data Setting

As segment data, the contents of RAM addresses X'7E0'~X'7FF' are transferred to the segment output ports for.

Figure 6-2-2 shows the correspondence between FLP pins and segment data.

		Segment Pin															
		SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
	Bits	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Digit Pin	DGT0	X'7E1'								X'7E0'							
	DGT1	X'7E3'								X'7E2'							
	DGT2	X'7E5'								X'7E4'							
	DGT3	X'7E7'								X'7E6'							
	DGT4	X'7E9'								X'7E8'							
	DGT5	X'7EB'								X'7EA'							
	DGT6	X'7ED'								X'7EC'							
	DGT7	X'7EF'								X'7EE'							
	DGT8	X'7F1'								X'7F0'							
	DGT9	X'7F3'								X'7F2'							
	DGT10	X'7F5'								X'7F4'							
	DGT11	X'7F7'								X'7F6'							
	DGT12	X'7F9'								X'7F8'							
	DGT13	X'7FB'								X'7FA'							
	DGT14	X'7FD'								X'7FC'							
	DGT15	X'7FF'								X'7FE'							

Figure 6-2-2 FLP Display RAM

As 8-bit installation is used for the entire FLP display area (X'7E0'~X'7FF'), areas not used for FLP display can be used as ordinary RAM.

Of these 32 bytes of memory, all bits in unused addresses when there are fewer than 16 digits, and bits not used for segments, can be used in the same way as ordinary RAM (in the latter case as flags, etc.).

6-2-4 FLP Display Timing Control

FLP display is controlled by the FLP clock mode register (FCMR) and digit number/dimmer register (DNDR).

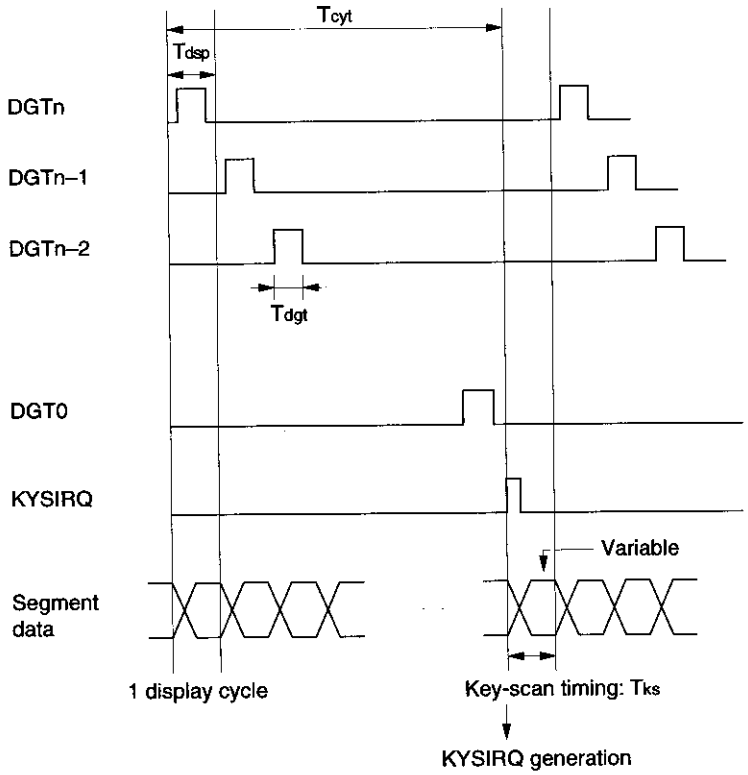
FLP clock mode register (FCMR: X'00E' R/W)

Digit number/dimmer register (DNDR: X'00D' R/W)
Bit 3: DR3
Bit 2: DR2
Bit 1: DR1
Bit 0: DR0

The FLP clock mode register (FCMR) specified FLP start/stop operation, the single-display cycle time (display time per digit), and serial display/automatic display repetition.
Bits DR3~DR0 of the digit number/dimmer register (DNDR) are used to select one of 15 dimmer waveform steps from 15/16 to 1/16.

[☞ "6-3-4 FLP Clock Mode Register (FCMR)"]
[☞ "6-3-3 Digit Number/Dimmer Register (DNDR)"]

The timing for serial display is shown in figure 6-2-3.



<Legend>
n: Set value (0~F) of bits DN3~DN0 in digit number/dimmer register (DNDR)
T_{dsp}: Single-display cycle (display time per digit) (244µs/488µs when fosc=4.19MHz)
T_{cyt}: Display cycle [$T_{cyt}=T_{dsp}\times(n+1)$]
T_{dgt}: Digit signal pulse width. Variable in 15 steps (15/16~1/16) by means of bits DR3~DR0 in the digit number/dimmer register (DNDR).

Figure 6-2-3 FLP Display Timing

The operating procedure for FLP display is as follows

① The digit pins go high in order from pin DGTn (where n is the value set in bits DN3~DN0 of the digit number/dimmer register (DNDR)) through DGTn–1, etc., to DGT0. The high-level period for one digit can be selected in the range 15/16 to 1/16 by means of bits DR3~DR0 in the digit number/dimmer register (DNDR), and the single-display cycle (display time per digit) can be selected from 512/fs or 256/fs with the DSPCLK bit in the FLP clock mode register (FCMR).

FLP clock mode register
(FCMR: X'00E' R/W)
Bit 2: DSPAUTO
Bit 1: DSPCLK
Bit 0: DSPSTART

② When a digit pin goes high, the corresponding FLP display RAM segment data is output from the segment pin.

③ The above operation is repeated, and with serial display, when DGT0 display ends, a key-scan interrupt (KYSIRQ) is generated and display is halted. When FLP display stops, all the digit pins go low. During this time, the segment pins can be used to perform key-scanning.

④ When the DSPAUTO bit is set to 1 in the FLP clock mode register (FCMR) (continuous display), there is no key-scan period, and display is restarted automatically from the DGTn pin.
When the DSPAUTO bit is cleared to 0 (serial display), display is restarted by setting the DSPSTART bit to 1 in the interrupt handling routine.
[☞ "6-2-5 Key-Scan Interrupt"]

6-2-5 Key-Scan Interrupt

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 5: IFKYS

Interrupt enable flag register 1
(IE1: X'015' R/W)
Bit 5: IEKYS

Port/FLP select register 0
(PFSR0: X'00A' R/W)

Port/FLP select register 1
(PFSR1: X'00B' R/W)

Port/FLP select register 2
(PFSR2: X'00C' R/W)

When the FLP control circuit completes a single display cycle (DGTn~DGT0 display cycle), a key-scan interrupt (KYSIRQ) is generated and the key-scan interrupt request flag (bit IFKYS) is set to 1 in interrupt request flag register 1 (IF1). If the key-scan interrupt enable flag (bit IEKYS) in interrupt enable flag register 1 (IE1) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

The interrupt handler carries out the following processing:

① Modifies the port/FLP select register 0 (PFSR0), port/FLP select register 1 (PFSR1), and port/FLP select register 2 (PFSR2), designating the pins to be used for key-scanning as general-purpose output ports.

② Performs key-scanning using the pins designated as general-purpose output ports in ①.

③ On completion of key-scanning, changes the pins back to their FLP function

④ Restarts FLP display by setting the DSPSTAT bit to 1 in the FLP clock mode register (FCMR), and returns to the main routine.

Note that the port output latch and segment output data latch are shared by the port 5 and 6 pins used as segment output pins.

6-3 FLP Control Registers

6-3-1 Overview

Five register bytes are used to control the MN187XX23's FLP: port/FLP select registers (PFSR0, PFSR1, PFSR2) which switch pins between port and FLP functions, and the digit number/dimmer register (DNDR) and FLP clock mode register (FCMR) which control FLP operation. These registers are all located in RAM space.

Table 6-3-1 FLP Control Registers

Register Abbreviation	RAM Address	R/W	Register Name
PFSR0	X'00A'	R/W	Port/FLP select register 0
PFSR1	X'00B'	R/W	Port/FLP select register 1
PFSR2	X'00C'	R/W	Port/FLP select register 2
DNDR	X'00D'	R/W	Digit number/dimmer register
FCMR	X'00E'	R/W	FLP clock mode register

6-3-2 Port/FLP Select Registers

The port/FLP select registers are read/write registers that switch pins between port and FLP (digit/segment) pin functions.

Port/FLP select register 0 (PFSR0) switches port 6 pin functions in bit units, port/FLP select register 1 (PFSR1) switches port 8 pin functions in bit units, and port/FLP select register 2 (PFSR2) switches port 5 and 7 pin functions in byte units. Pins for which the port function is selected can be used as general-purpose P-channel open-drain output pins.

When the FLP/port switch is set to the FLP, data written to a port address is not written to the output latch. To write data to the output latch, the FLP/port switch must first be set to the port side.

(1) Port/FLP select register 0 (PFSR0)

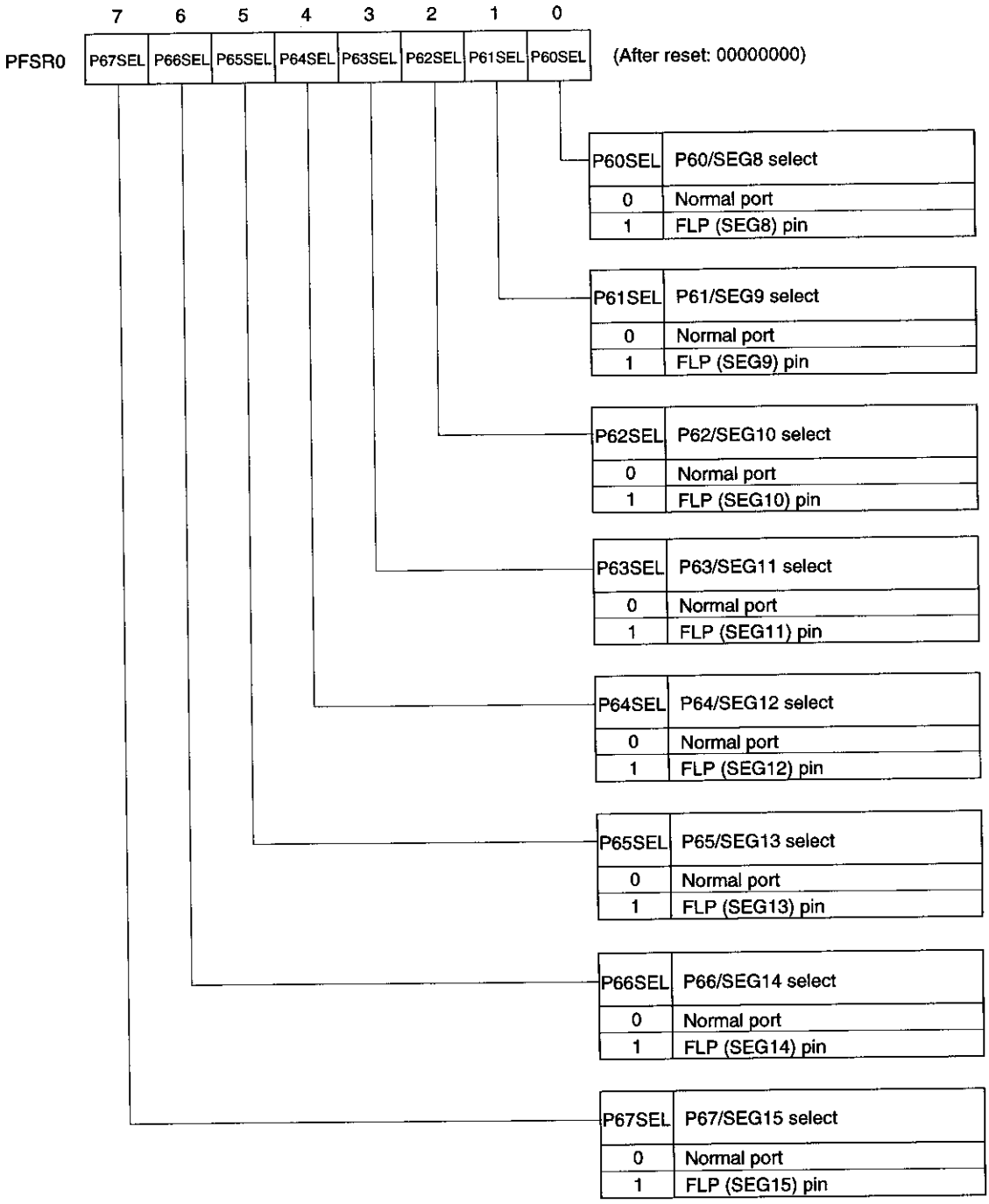


Figure 6-3-1 Port/FLP Select Register 0 (PFSR0: X'00'A R/W)

In a reset, all bits in PFSR0 are cleared to 0.

(2) Port/FLP select register 1 (PFSR1)

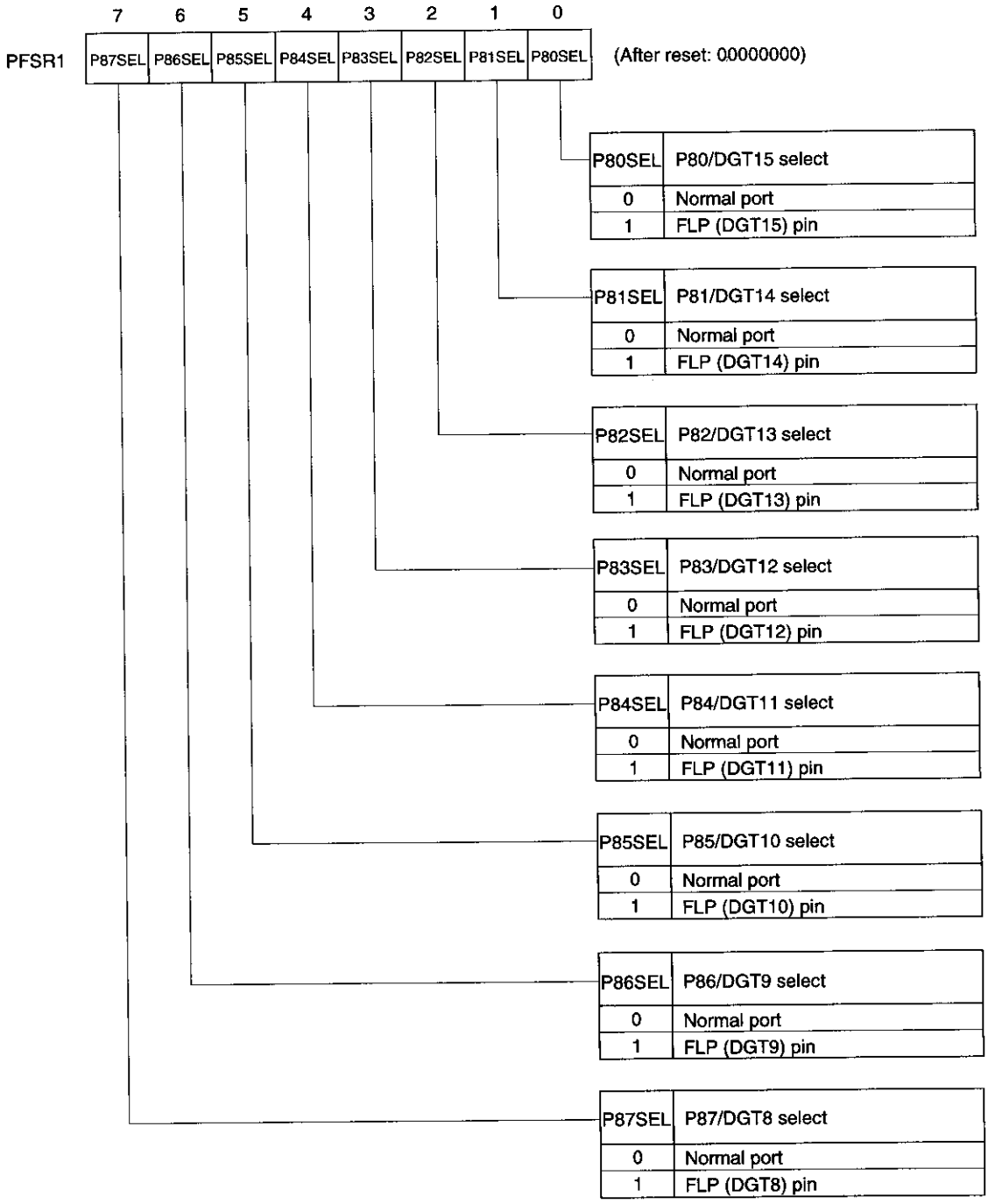


Figure 6-3-2 Port/FLP Select Register 1 (PFSR1: X'00B' R/W)

In a reset, all bits in PFSR1 are cleared to 0.

(3) Port/FLP select register 2 (PFSR2)

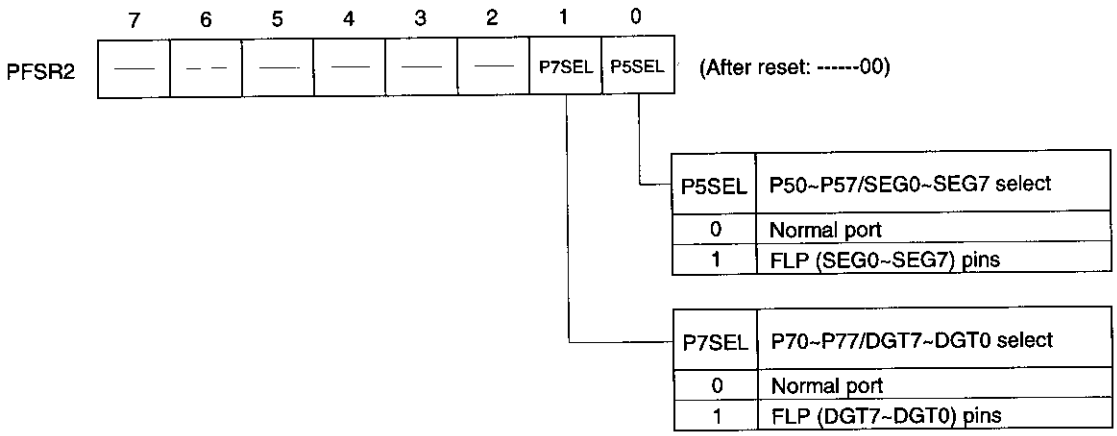


Figure 6-3-3 Port/FLP Select Register 2 (PFSR2: X'00C' R/W)

In a reset, bits P5SEL and P7SEL in PFSR1 are both cleared to 0.

6-3-3 Digit Number/Dimmer Register (DNDR)

DNDR is an 8-bit read/write register in which the number of digits is set in bits 7~4, and the dimmer data in bits 3~0.

	7	6	5	4	3	2	1	0	
DNDR	DN3	DN2	DN1	DN0	DR3	DR2	DR1	DR0	(After reset: XXXXXXXX)

DR3	DR2	DR1	DR0		Dimmer waveform (T_{dsp})
0	0	0	0	(0)	15/16
			1	(1)	14/16
		1	0	(2)	13/16
			1	(3)	12/16
	1	0	0	(4)	11/16
			1	(5)	10/16
		1	0	(6)	9/16
			1	(7)	8/16
1	0	0	0	(8)	7/16
			1	(9)	6/16
		1	0	(A)	5/16
			1	(B)	4/16
	1	0	0	(C)	3/16
			1	(D)	2/16
		1	0	(E)	1/16
			1	(F)	Setting prohibited

DN3	DN2	DN1	DN0	Number of digit
0	0	0	0	1
			1	2
		1	0	3
			1	4
	1	0	0	5
			1	6
		1	0	7
			1	8
1	0	0	0	9
			1	10
		1	0	11
			1	12
	1	0	0	13
			1	14
		1	0	15
			1	16

Figure 6-3-4 Digit Number/Dimmer Register (DNDR: X'00D' R/W)

T_{dsp}: Single-display cycle

In a reset, all bits in DNDR are undefined.

The number of display digits is the value set in bits DN0~DN3+1.

The digit waveform cut width Tcut is given by the following equation:

$T_{cut}=\{(n+1)/16\}T_{dsp}$ (n=0~E)

(n: value set in bits DR0~DR3)

6-3-4 FLP Clock Mode Register (FCMR)

FCMR is a 3-bit read/write register used to specify FLP display start/stop operation and the single-display cycle time.

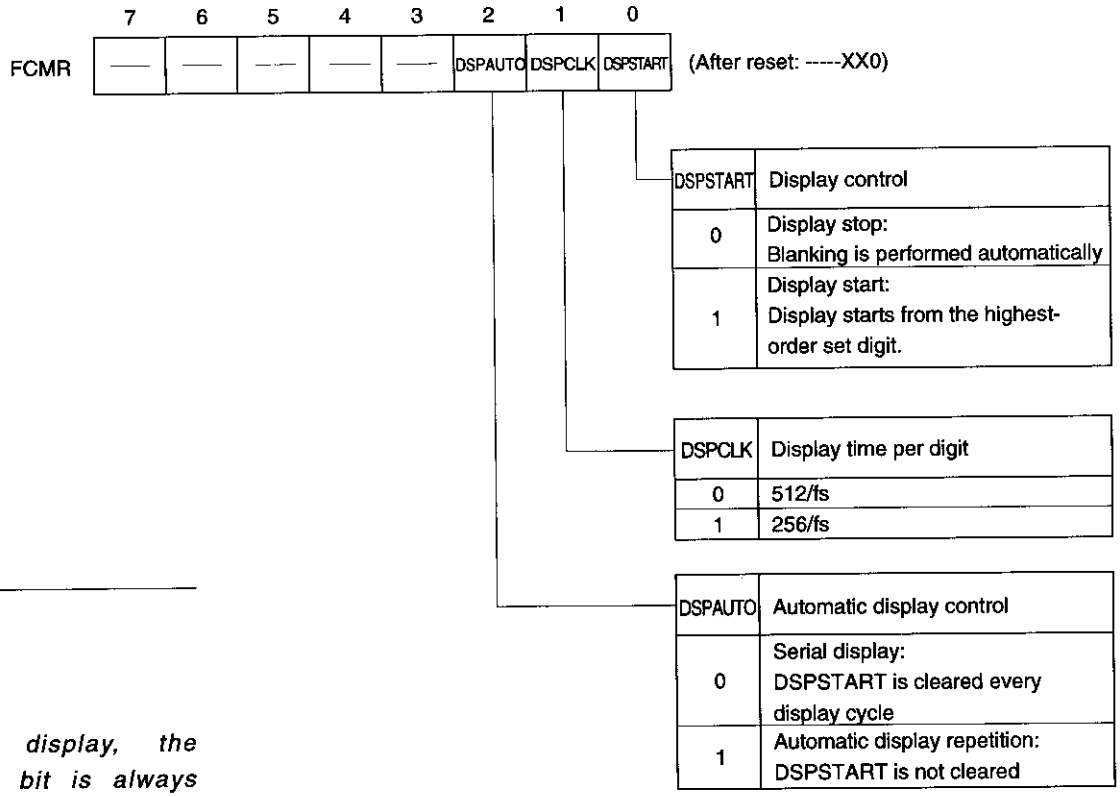


Figure 6-3-5 FLP Clock Mode Register (FCMR: X'00E' R/W)

In a reset, the DSPSTART bit is cleared to 0 and the other bits are undefined.

In serial display, the DSPSTART bit is always cleared when a key-scan interrupt is generated. If display is interrupted by software, the key-scan interrupt request flag (IFKYS) may be set.

6-4 Notes

① When data for one digit is automatically transferred from the FLP display RAM to the output port, execution of the user program stops for 2 machine cycles. That is, program execution stops for 1.91μs every 244μs or 488μs (when fosc=4.19MHz). The instruction execution timing in this case is shown in figure 6-4-1.

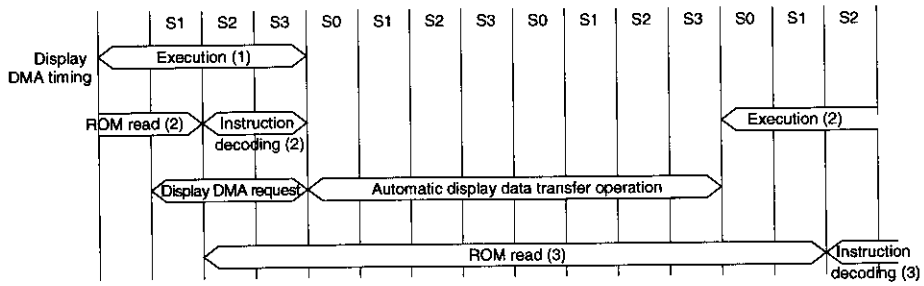
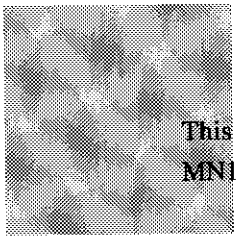


Figure 6-4-1 Timing of Interruption of Instruction Execution by FLP Display

② When the program writes 0 in the DSPSTART bit in the FLP clock mode register (FCMR), the display buffer is cleared and the first digit is not displayed. To continue display, 0 should be written in the DSPSTART bit by the interrupt routine.

FLP clock mode register (FCMR: X'00E' R/W) Bit 0: DSPSTART

Chapter 7 PWM Output



This chapter describes the operation of the MN187XX23's PWM output circuits.

7-1 Overview

The MN187XX23 provides one 14-bit PWM output circuit (PWM0) and two 8-bit PWM output circuits (PWM1, PWM2).
A PWM0 has 14-bit resolution, with a minimum pulse width of 1/fs and PWM repetition cycle of 2^{14}/fs.
PWM1 and PWM2 have 8-bit resolution, with a minimum pulse width of 1/fs and PWM repetition cycle of 2^{8}/fs. PWM1 and PWM2 use the lower 8 bits of the PWM0 14-bit counter as a counter, and therefore PWM0 must be in operation when PWM1 and PWM2 are used.

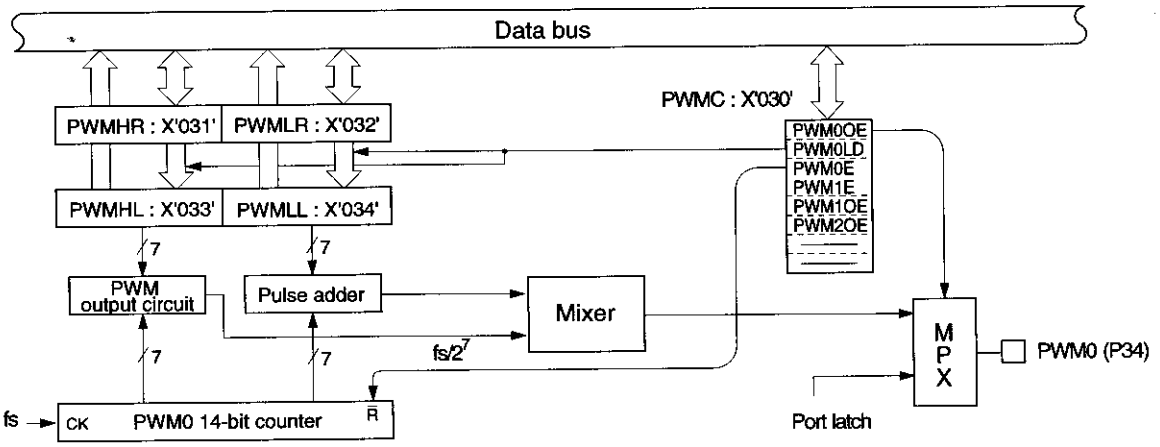


Figure 7-1-1 PWM0 Output Circuit Block Diagram

The PWM0 14-bit counter incorporated in PWM0 is used for fs divider by the MN187XX23 modules listed below. Therefore, PWM0 must be in operation when these modules are used.

- *Timer 1: When using internal clock divided from fs*
- *Timer 2: When fs/2^{12} is selected as clock*
- *Timer 3: When using internal clock divided from fs*
- *Timer 4: When using clock divided from fs*
- *Watchdog timer*
- *Buzzer output function*
- *Serial interface 0: When using internal clock divided from fs*
- *Serial interface 1: When using internal clock divided from fs*
- *Serial interface 2: When using internal clock divided from fs*
- *PWM1*
- *PWM2*
- *Noise filter circuit*
- *Oscillation stabilization time count when exiting STOP mode*

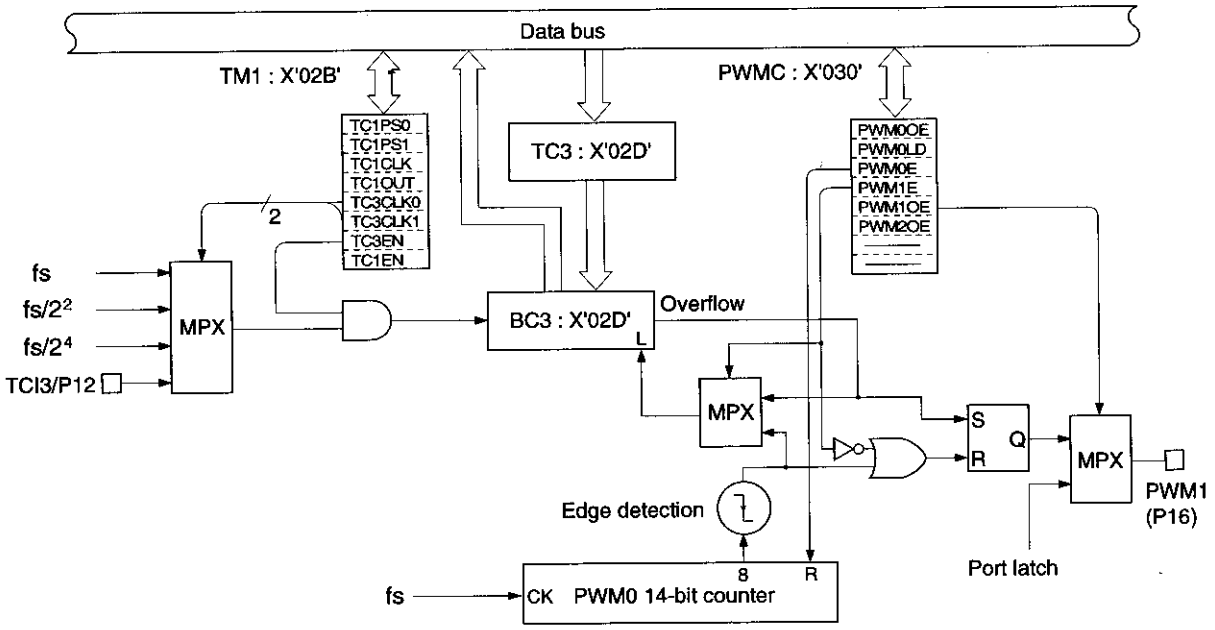


Figure 7-1-2 PWM1 Output Circuit Block Diagram

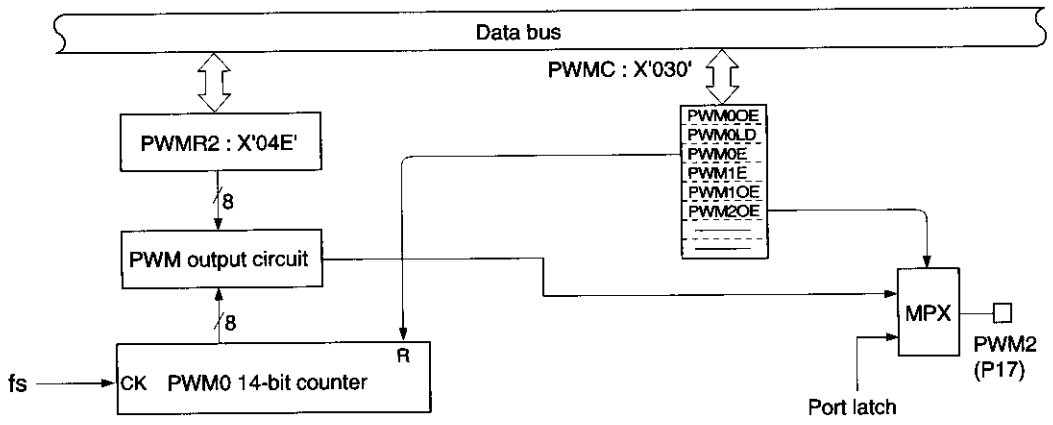


Figure 7-1-3 PWM2 Output Circuit Block Diagram

7-2 PWM0 Output Circuit Operation

7-2-1 Overview

PWM0 output waveform data is written to the PWM0 data register (PWMHR, PWMLR).

PWM0 data register (PWMHR: X'031' R/W, PWMLR: X'032' R/W)

PWM output is started when bit PWM0OE in the PWM control register (PWMC) and bit P34ON in the port 3 direction control register (P3DIR) are both set to 1, designating the port P34 (PWM0) pin for PWM0 output, and bit PWM0E is set to 1. The PWM0 data latch (PWMHL, PWMLL) is provided to prevent disruption of the waveform if the PWM0 data register (PWMHR, PWMLR) is rewritten during PWM0 output.

PWM control register (PWMC: X'030' R/W)
Bit 2: PWM0E
Bit 0: PWM0OE

PWM0 data latch (PWMHL: X'033' R, PWMLL: X'034' R)

7-2-2 Register Settings

The register settings for controlling PWM0 are shown below.

Table 7-2-1 PWM Output Register Settings

Register	Bits	Mnemonic	Setting
PWMC	0	PWM0OE	Set to 1 to designate P34 (PWM0) pin as PWM0 pin.
	1	PWM0LD	Cleared to 0 to enable transfer from PWM data register to PWM data latch.
	2	PWM0E	Set to 1 to activate PWM.
PWMHR	0~6	PWMHR0 ~ PWMHR6	Sets PWM0 basic waveform in PWM0 basic cycle (Tsub = 2^7/fs). Choice of 128 duty cycles in 1/fs units.
PWMLR	0~6	PWMLR0 ~ PWMLR6	Sets position of additional pulse superimposed on PWM0 output basic waveform determined by PWMHR in repeat cycle (Tstd = 2^{14}/fs). Additional pulse width is 1/fs.

The PWM0LD bit in the PWM control register (PWMC) is used to disable transfer to the PWM0 data latch (PWMHL, PWMLL) by being set to 1 when the PWM0 data register (PWMHR, PWMLR) is rewritten during PWM output.

[☞ "7-2-4 Changing Output Data in PWM0 Output"]

The PWM0E bit must also be set to 1 when the PWM0 14-bit counter is used by a timer, serial interface, etc., to divide the system clock (fs).

The upper 7 bits (PWMHR) of the PWM0 data register specify 127:1 duty at PWMHR=00, and 0:128 duty at PWMHR=7F.

The lower 7 bits (PWMLR) of the PWM0 data register specify superimposition of additional pulses on all PWM basic waveforms at PWMLR=00, and superimposition of an additional pulse only on the last PWM basic waveform in a repeat cycle at PWMLR=7F. [☞ "7-2-3 Output Waveform"]

7-2-3 Output Waveform

CPU mode register
(CPUM: X'016' R/W)
Bit 6: CLKSEL

The PWM0 output waveform is shown in figure 7-2-1. In the following description, conditions are set with fosc=4.19MHz, the CLKSEL bit in the CPU mode register (CPUM) cleared to 0, and NORMAL mode selected (in this case, fs=1.0475MHz). For PWM0 output, a waveform modulated by the upper 7 bits (PWMHR) of the PWM0 data register with a cycle of Tsub (122μs) is used as the basic waveform, and an additional pulse (1/fs=0.95μs) is superimposed at the position specified by the lower 7 bits (PWMLR) of the PWM0 data register.

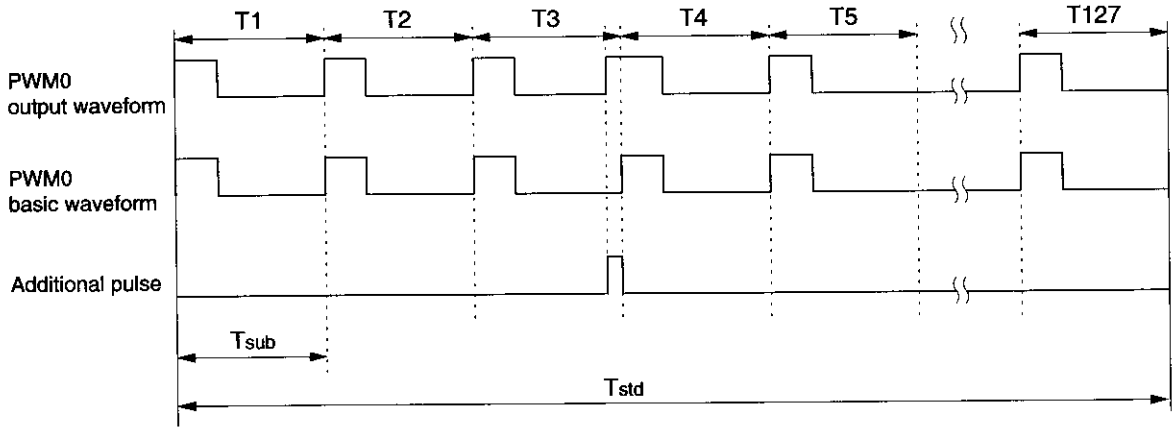


Figure 7-2-1 PWM0 Output Waveform

T_{sub}: PWM0 basic cycle (2^7/fs)

T_{std}: PWM0 repeat cycle (2^{14}/fs)

T1~T127: PWM0 basic cycle number

(1) PWM0 basic waveform setting by upper 7 bits (PWMHR) of PWM0 data register

PWM0 data register upper 7 bits (PWMHR: X'031' R/W)

The PWMHR specifies the PWM0 basic waveform in the PWM0 basic cycle (T_{sub}=122μs). The relation between the PWMHR set value and the PWM0 basic cycle is shown in figure 7-2-2.

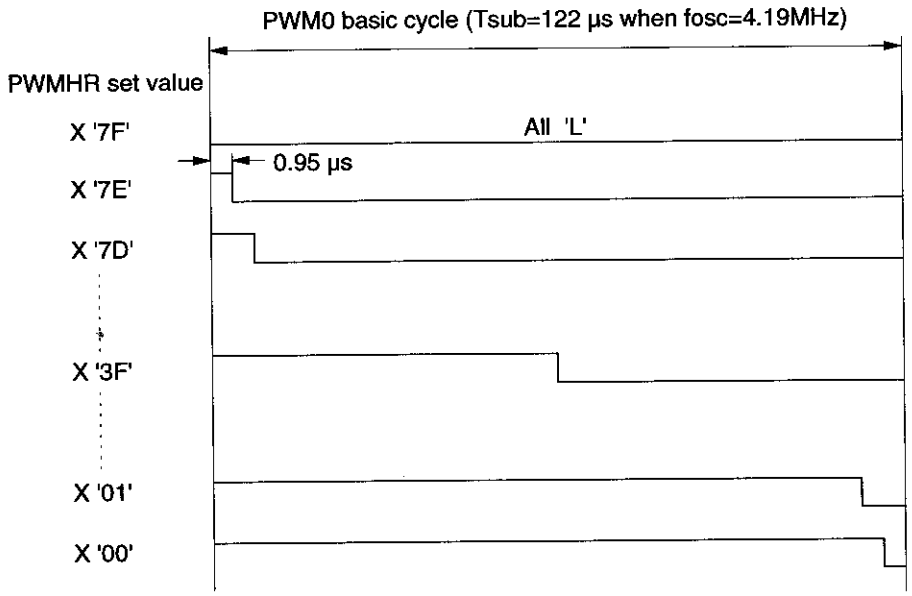


Figure 7-2-2 Relation between PWM0 Basic Waveform and PWMHR Set Value

(2) Additional pulse setting by lower 7 bits (PWMLR) of PWM0 data register

The PWMLR specifies the position of additional pulses superimposed on the PWM0 basic waveform in a PWM0 repeat cycle (T_{std}=15.64ms). The relation between the PWMLR set value and the position of the additional pulses is shown in table 7-2-2, and the output waveforms are shown in figure 7-2-3.

Table 7-2-2 Position of Additional Pulse Waveform Superimposition

PWMLR Data (MSB) (LSB)	Tn at which Additional Pulses are Superimposed (Value of n)
1 1 1 1 1 1 1	127
1 1 1 1 1 1 0	63, 127
1 1 1 1 1 0 1	31, 95, 127
1 1 1 1 0 1 1	15, 47, 79, 111, 127
1 1 1 0 1 1 1	7, 23, 39, 55, 71, 87, 103, 119, 127
1 1 0 1 1 1 1	3, 11, 19, 27, 35, 43, 51, 59, ... 123, 127
1 0 1 1 1 1 1	1, 5, 9, 13, 17, 21, 25, 29, ... 125, 127
0 1 1 1 1 1 1	0, 2, 4, 6, 10, 12, 14, ... 126, 127

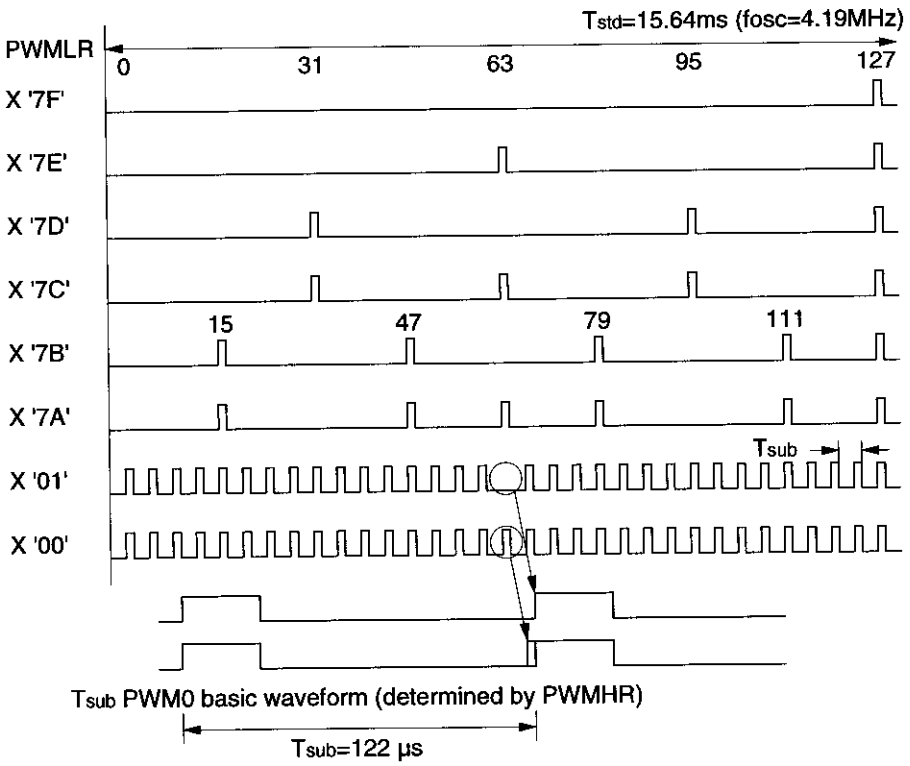


Figure 7-2-3 Relation between PWMLR Set Value and Additional Pulse Position

7-2-4 Changing Output Data in PWM Output

When the PWM0LD bit is cleared to 0 in the PWM control register (PWMC), 14-bit data is transferred from the PWM0 data register (PWMHR, PWMLR) to the PWM0 data latch (PWMHL, PWMLL) in the next PWM0 basic cycle (T_{sub}). This is done to prevent disruption of the PWM waveform when data is changed during PWM0 output, and the transfer is performed with a maximum of 2^8/fs (244μs at fosc=4.19MHz) after the PWM0LD bit is cleared.

The PWM0 waveform does not change if the value in the PWM0 data register (PWMHR, PWMLR) is modified after transfer from the PWM0 data register (PWMHR, PWMLR) to the PWM0 data latch (PWMHL, PWMLL) has been disabled by setting the PWM0LD bit to 1.

The PWM0 data latch (PWMHL, PWMLL) contents after the transfer can be read at any time.

PWM control register (PWMC: X'030' R/W)
Bit 2: PWM0E
Bit 1: PWM0LD
Bit 0: PWM0OE

PWM0 data register (PWMHR: X'031' R/W, PWMLR: X'032' R/W)

PWM0 data latch (PWMHL: X'033' R, PWMLL: X'034' R)

The procedure for rewriting the PWM0 data register (PWMHR, PWMLR) is as follows.

① First set a value in the PWM0 data register (PWMHR, PWMLR), then set bit PWM0OE to 1, bit PWM0LD to 0, and bit PWM0E to 1 in the PWM control register (PWMC) to start PWM0 output.

② Next, set the PWM0LD bit to 1 to disable transfer to the PWM0 data latch.

③ Write the new data to the PWM0 data register (PWMHR, PWMLR).

④ Clear the PWM0LD bit to 0 to enable transfer to the PWM0 data latch (PWMHL, PWMLL).

⑤ Set the PWM0LD bit to 1 in preparation for the next change.

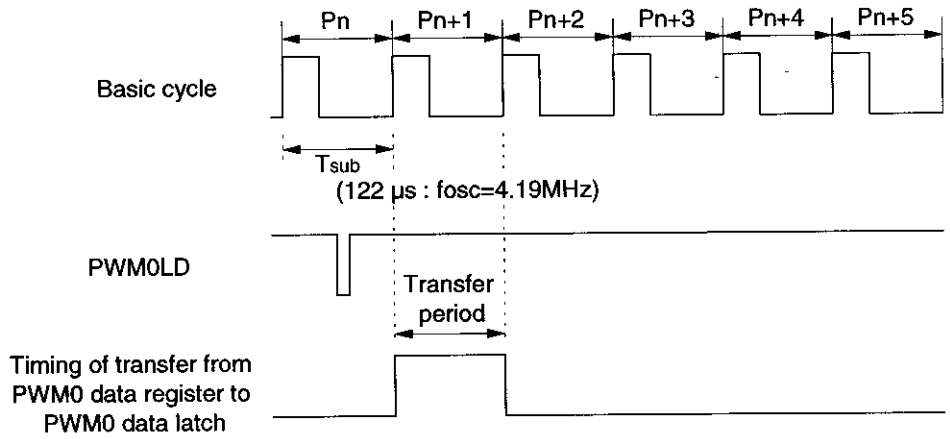


Figure 7-2-4 Timing of Data Transfer from PWM0 Data Register to PWM0 Data Latch

The output waveform may be disrupted if the PWM0 data register is rewritten during the transfer period.

Do not rewrite the PWM0 data register for an interval of 2^8/fs (244 µs when fosc = 4.19MHz) after clearing the PWM0LD bit to 0.

7-3 PWM1 Output Circuit Operation

7-3-1 Overview

PWM1 shares functions with timer 3, and its output waveform data is set in timer latch 3 (TC3). PWM output is started when bit PWM1OE in the PWM control register (PWMC) and bit P16ON in the port 1 direction control register (P1DIR) are both set to 1, designating the port P16 (PWM1) pin for PWM1 output, and bits PWM0E and PWM1E are both set to 1.

7-3-2 Register Settings

The register settings for controlling PWM1 are shown below.

Table 7-3-1 PWM1 Output Register Settings

Register	Bits	Mnemonic	Setting
PWMC	2	PWM0E	Set to 1 to make PWM0 operational.
	3	PWM1E	Set to 1 to use timer latch 3/binary counter 3 (TC3/BC3) as PWM1.
	4	PWM1OE	Set to 1 to use P16 (PWM1) pin as PWM1 pin.
TM1	4, 5	TC3CLK0, TC3CLK1	TC3CLK0, TC3CLK1 set to 00 to set count clock to fs.
	6	TC3EN	Set to 1 to make timer 3 operational.
TC3	0~7	TC30~TC37	Sets PWM1 output low-level width.

In PWM1 output, the output of the lower 8 bits of the PWM0 14-bit counter is used for PWM cycle generation. Therefore, PWM0 must be made operational when PWM1 output is used.

7-3-3 Output Waveform

The PWM1 output waveform is shown in figure 7-3-1. When the timer overflows while PWM1 is enabled (PWM1E=1), P16 (PWM1) pin output changes from 0 to 1. The transition from 1 to 0 is made on the falling edge of bit 8 of the PWM0 14-bit counter, and at the same time, the timer latch 3 (TC3) value is loaded into binary counter 3 (BC3). Thus, the PWM1 output waveform has a 2^8/fs PWM cycle, determined by the PWM0 14-bit counter bit 8 output, and a low-level width determined by the TC3 set value.

[☞ Figure 7-1-2 PWM1 Output Circuit Block Diagram]

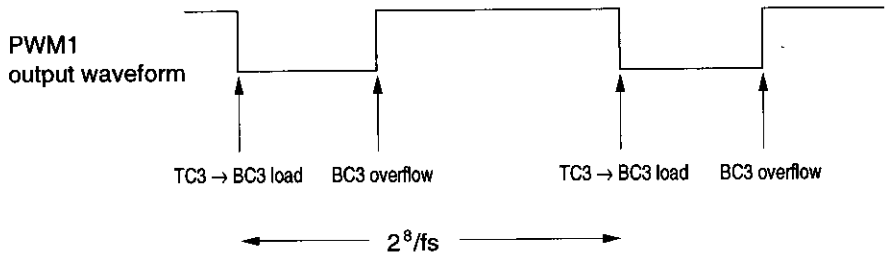


Figure 7-3-1 PWM1 Output Waveform

With PWM1 output, the following timer 3 settings should be made in timer mode register 1 (TM1).

① Clear bits TC3CLK0 and TC3CLK1 to 00, designating fs as the timer 3 count clock.

② Set the TC3EN bit to 1, making timer 3 operational.

7-4 PWM2 Output Circuit Operation

7-4-1 Overview

PWM2 output waveform data is written to the PWM2 data register (PWMR2). PWM2 output, for which P17 direction control is set for output, is started when bit PWM2OE in the PWM control register (PWMC) and bit P17ON in the port 1 direction control register (P1DIR) are both set to 1, designating the port P17 (PWM2) pin as the PWM2 pin, and bit PWM0E is set to 1. PWM2 use the lower 8 bits of the PWM0 14-bit counter as its PWM cycle generation counter.

7-4-2 Register Settings

The register settings for controlling PWM2 are shown below.

Table 7-4-1 PWM2 Output Register Settings

Register	Bits	Mnemonic	Setting
PWMC	2	PWM0E	Set to 1 to make PWM0 operational.
	5	PWM2OE	Set to 1 to use P17 (PWM2) pin as PWM2 pin.
PWMR2	0~7	PWMR20 ~PWMR27	Sets PWM2 output waveform. Choice of 256 duty cycles in 1/fs units.

In PWM2 output, the output of the lower 8 bits of the PWM0 14-bit counter is used for PWM cycle generation. Therefore, PWM0 must be made operational when PWM2 output is used.

A PWM2 data register (PWMR2) setting of 00 specifies 255:1 duty, and a setting of FF specifies 0:256 duty.

7-4-3 Output Waveform

The PWM2 output waveform is shown in figure 7-4-1. In the following description, conditions are set with fosc=4.19MHz, the CLKSEL bit in the CPU mode register (CPUM) cleared to 0, and NORMAL mode selected (in this case, fs=1.0475MHz). For PWM2 output, a waveform modulated by the PWM2 data register (PWMR2) with a Tsub (244μs) cycle. The minimum pulse width is 0.95μs and the resolution is 8 bits.

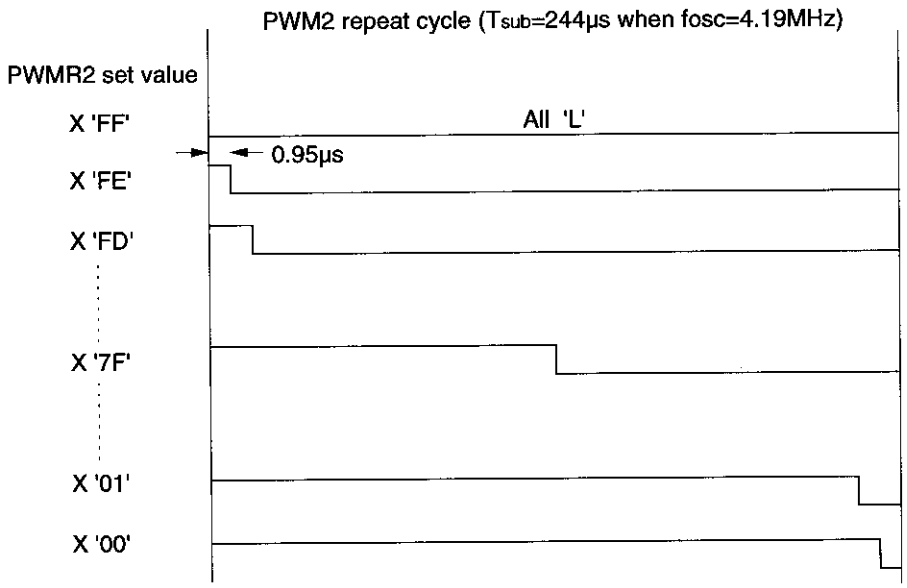


Figure 7-4-1 PWM2 Output Waveform

7-5 PWM Control Registers

7-5-1 Overview

Eight registers are used to control the MN187XX23's PWM output circuits: the PWM control register (PWMC) which controls the operation of the PWM0, PWM1, and PWM2 output circuits, the PWM0 data register (PWMHR, PWMLR) and PWM0 data latch (PWMHL, PWMLL) in which PWM0 output waveform data is set, timer mode register 1 (TM1) which control the operation of PWM1 (timer 3), timer latch 3 (TC3) in which PWM1 output waveform data is set, and the PWM2 data register (PWMR2) in which PWM2 output waveform data is set. These registers are all located in RAM space.

Table 7-5-1 PWM Output Circuit Registers

Register Abbreviation	RAM Address	R/W	Register Name
PWMC	X'030'	R/W	PWM control register
PWMHR	X'031'	R/W	PWM0 data register upper 7 bits
PWMLR	X'032'	R/W	PWM0 data register lower 7 bits
PWMHL	X'033'	R	PWM0 data latch upper 7 bits
PWMLL	X'034'	R	PWM0 data latch lower 7 bits
TM1	X'02B'	R/W	Timer mode register 1
TC3/BC3	X'02D'	R*/W	Programmable timer counter 3 (timer latch 3/binary counter 3)
PWMR2	X'04E'	R/W	PWM2 data register

R/W: Can be both read and written to.

R: Read-only

R*/W: When read, the binary counter value is returned.
When written to, the value is written into the timer latch.

7-5-2 PWM Control Register (PWMC)

PWMC is a 6-bit read/write register that controls the operation of the PWM output circuit.

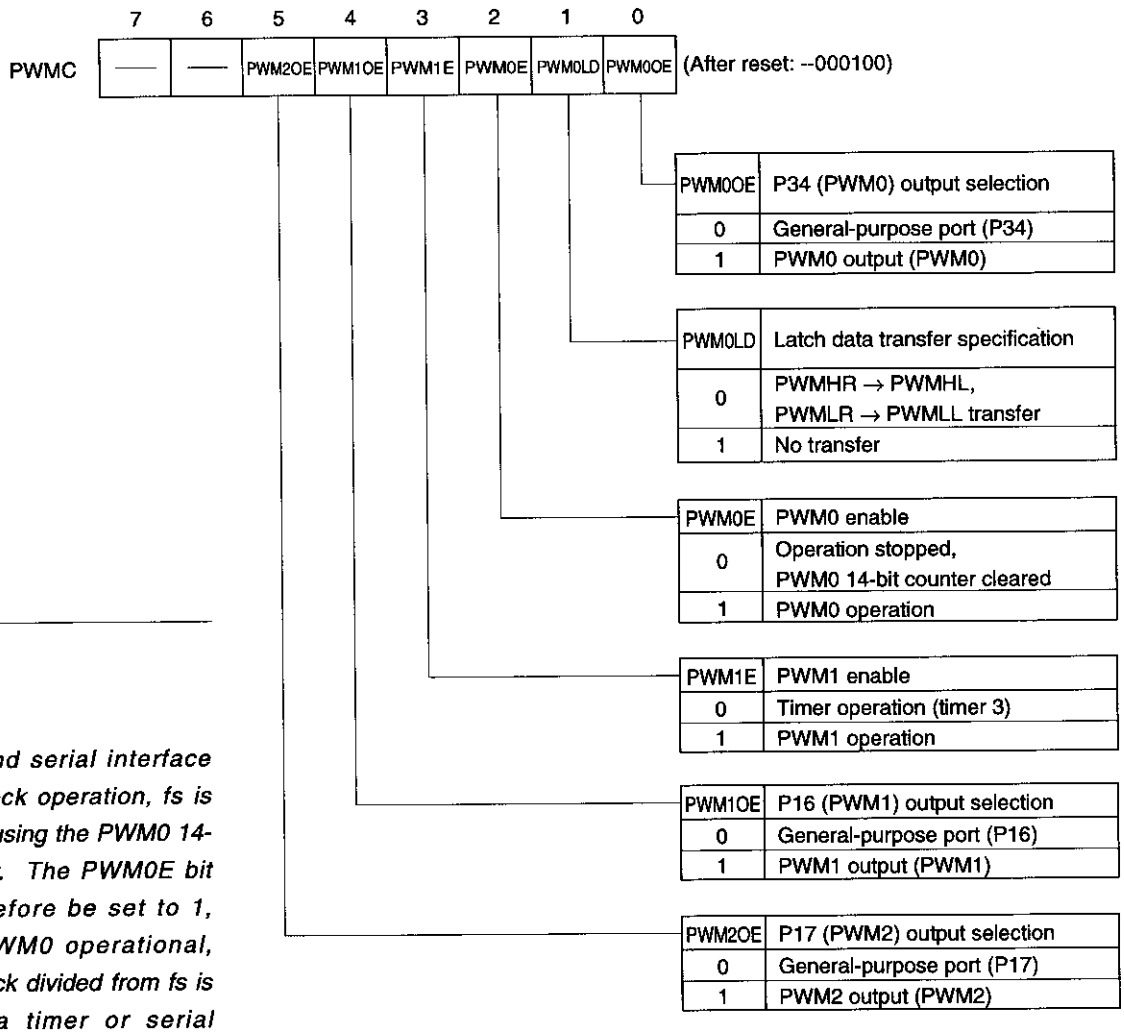


Figure 7-5-1 PWM Control Register (PWMC: X'030' R/W)

In timer and serial interface internal clock operation, fs is divided by using the PWM0 14-bit counter. The PWM0E bit must therefore be set to 1, making PWM0 operational, when a clock divided from fs is used by a timer or serial interface.

At reset, the PWM0E bit is set to 1 and the other bits are cleared to 0.

The PWM0 14-bit counter is used to count the oscillation stabilization time when recovering from STOP mode. The PWM0E bit must therefore be set to 1 before STOP mode is entered.

7-5-3 PWM0 Data Register (PWMHR, PWMLR)

PWMHR and PWMLR are 7-bit read/write registers used to set PWM0 output waveform data in one repeat cycle.

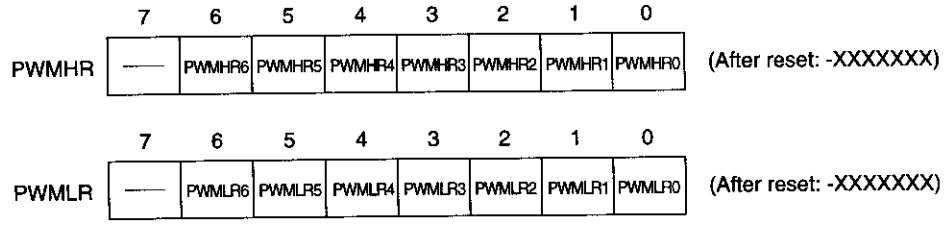


Figure 7-5-2 PWM0 Data Register (PWMHR: X'031' R/W, PWMLR: X'032' R/W)

At reset, all bits are undefined.

☞ For the relation between the value set in the PWM0 data register and the PWM0 output waveform, see "7-2-3 Output Waveform."

7-5-4 PWM0 Data Latch (PWMHL, PWMLL)

PWMHL and PWMLL are 7-bit read-only registers, used for transfer of the PWM0 data register (PWMHR, PWMLR) contents when bit 1 (PWM0LD) of PWMC is cleared to 0. These registers are used to prevent disruption of the waveform when the output data is changed during waveform output.

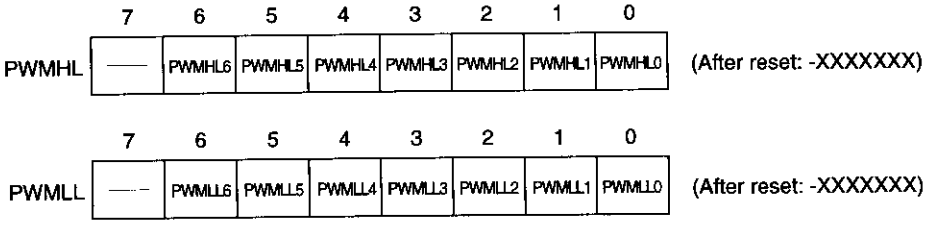


Figure 7-5-3 PWM0 Data Latch (PWMHL: X'033' R, PWMLL: X'034' R)

At reset, all bits are undefined.

PWMHL and PWMLL can be written to only by a transfer of the PWMHR and PWMLR contents. PWMHL and PWMLL can be read at all times, but the value of bit 7 in each register is undefined.

☞ For the functions of PWMHL and PWMLL, see "7-2-4 Changing Output Data in PWM Output."

7-5-5 PWM2 Data Register (PWMR2)

PWMR2 is an 8-bit read/write register, used to set the output waveform data within one repeat cycle.

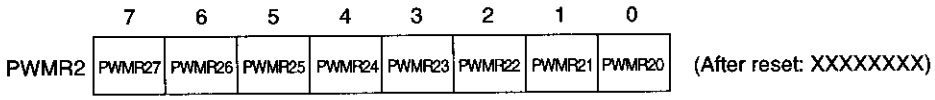
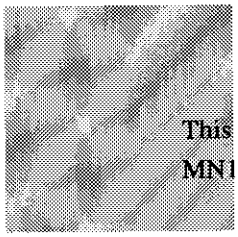


Figure 7-5-4 PWM2 Data Register (PWMR2: X'04E' R/W)

☞ For the relation between the value set in the PWM2 data register and the PWM2 output waveform, see "7-4-3 Output Waveform."

At reset, all bits are undefined.



This chapter describes the operation of the MN187XX23's A/D converter.

Chapter 8 A/D Conversion

8

8-1 Overview

The MN187XX23 has an on-chip A/D converter with an 8-bit resolution. It includes a sample-and-hold circuit, and the 8 analog input channels, 0 to 7 (ADIN0 (P20)~ADIN7 (P27)), can be switched by software. It has a dedicated analog power supply (VREFH, VREFL), and any comparison voltage can be set in the range 0V to 5V, enabling high-precision A/D conversion.

When the A/D converter is not operating, the built-in ladder resistor can be turned off to reduce power consumption.

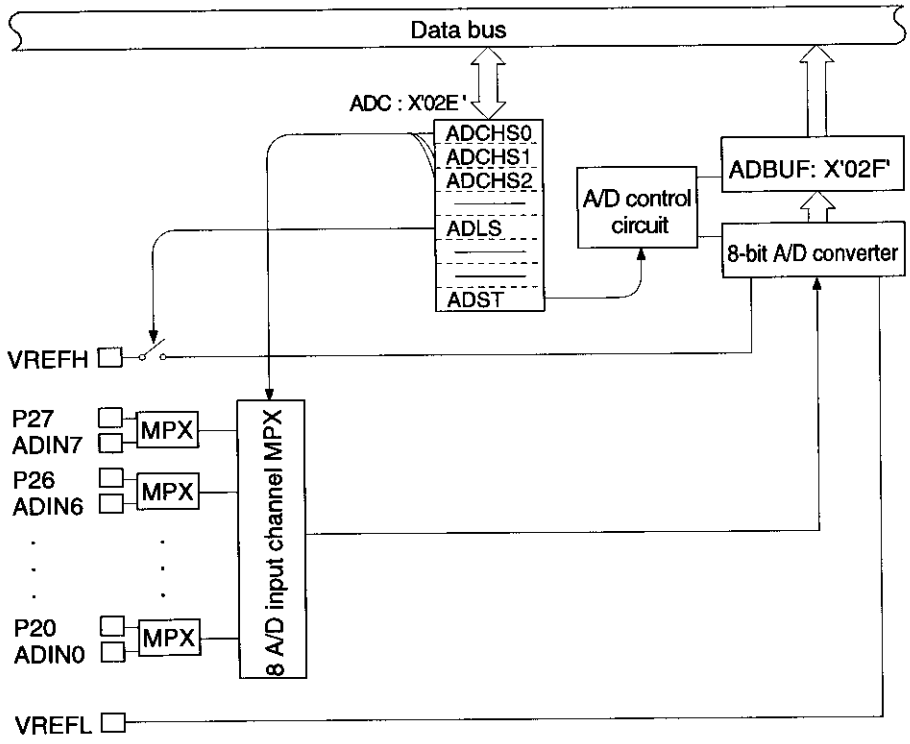


Figure 8-1-1 A/D Converter Block Diagram

8-2 A/D Conversion Operation

The A/D conversion procedure is as follows.

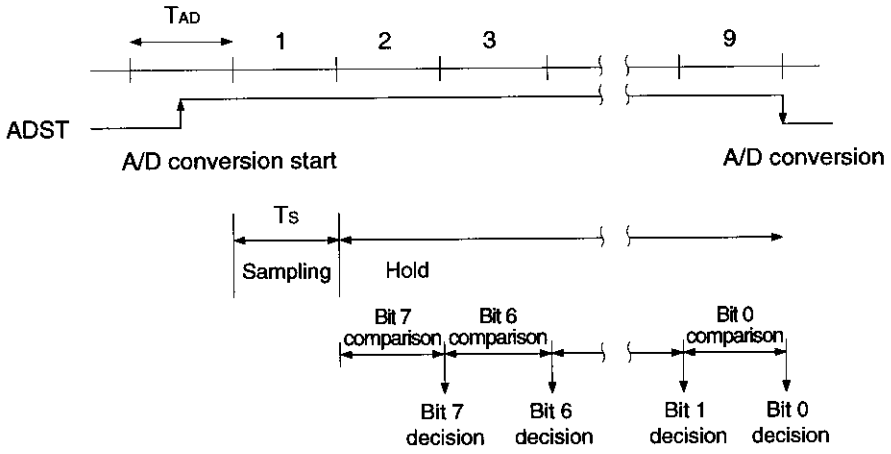
A/D control register
(ADC: X'02E' R/W)
Bit 7: ADST
Bit 4: ADLS
Bit 2: ADCHS2
Bit 1: ADCHS1
Bit 0: ADCHS0

A/D buffer
(ADBUF: X'02F' R)
Bits 7~0:
ADBUF7~ADBUF0

① Select the analog input pin from ADIN0~ADIN7 (P20~P27) by means of bits ADCHS2~ADCHS0 in the A/D control register (ADC).

② Set the ADLS bit to 1 in the A/D control register (ADC) to send current through the ladder resistor, and set the ADST bit to 1 to start A/D conversion.

③ In A/D conversion, after sampling has been performed in the sampling interval, Ts, successive comparisons and decisions are made, starting with the MSB. If the A/D conversion reference clock cycle is designated TAD (0.954μs when fosc=4.19MHz), conversion completes at 9×TAD (= 8.59μs).



Ts= TAD (=0.954 μs)

Figure 8-2-1 A/D Conversion Timing

④ When A/D conversion is completed, the ADST bit is cleared to 0 and the A/D conversion result is stored in the A/D buffer (ADBUF). Check that the ADST bit has been cleared to 0 before reading the A/D buffer (ADBUF) contents.

8-3 A/D Converter Control Registers

8-3-1 Overview

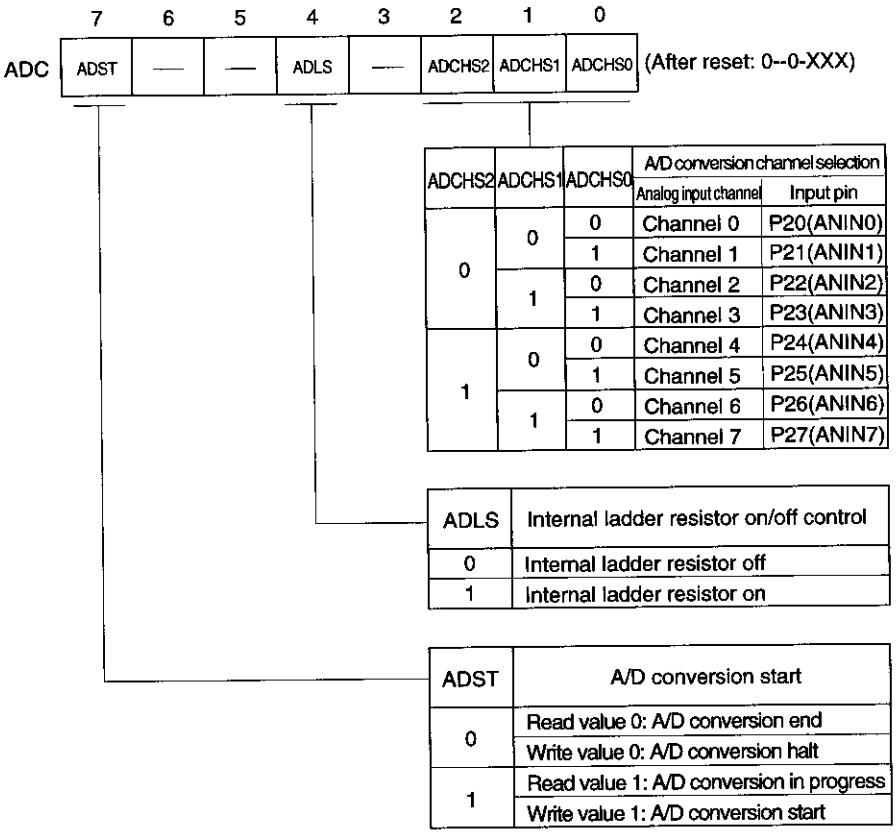
Two registers are used to control the MN187XX23's A/D converter: the A/D control register (ADC) which controls analog input channel selection, the on/off status of the ladder resistor, and the start of A/D conversion, and the A/D buffer (ADBUF) which stores the conversion result. These registers are both located in RAM space.

Table 8-3-1 A/D Converter Control Registers

Register Abbreviation	RAM Address	R/W	Register Name
ADC	X'02E'	R/W	A/D control register
ADBUF	X'02F'	R	A/D buffer

8-3-2 A/D Control Register (ADC)

ADC is a 5-bit read/write register which controls the operation of the A/D converter.



7	6	5	4	3	2	1	0	
ADC	ADST	—	—	ADLS	—	ADCHS2	ADCHS1	ADCHS0	(After reset: 0--0-XXX)

ADCHS2	ADCHS1	ADCHS0	A/D conversion channel selection	
			Analog input channel	Input pin
0	0	0	Channel 0	P20(ANIN0)
		1	Channel 1	P21(ANIN1)
	1	0	Channel 2	P22(ANIN2)
		1	Channel 3	P23(ANIN3)
1	0	0	Channel 4	P24(ANIN4)
		1	Channel 5	P25(ANIN5)
	1	0	Channel 6	P26(ANIN6)
		1	Channel 7	P27(ANIN7)

ADLS	Internal ladder resistor on/off control
0	Internal ladder resistor off
1	Internal ladder resistor on

ADST	A/D conversion start
0	Read value 0: A/D conversion end
	Write value 0: A/D conversion halt
1	Read value 1: A/D conversion in progress
	Write value 1: A/D conversion start

Figure 8-3-1 A/D Control Register (ADC: X'02E' R/W)

When the A/D converter is not used, current should be restricted by writing 0 in the ADLS bit in ADC.

At reset, bits ADST and ADLS are cleared to 0, and the other bits are undefined.

8-3-3 A/D Buffer (ADBUF)

ADBUF is an 8-bit read-only register which stores the A/D conversion result.

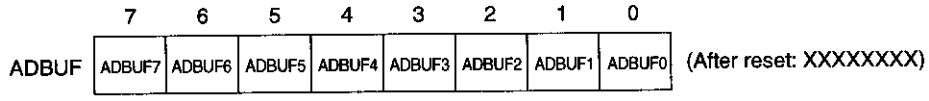
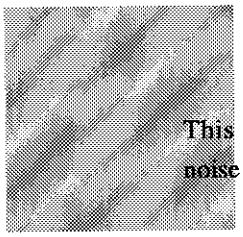


Figure 8-3-2 A/D Buffer (ADBUF: X'02F' R)

At reset, all bits are undefined.



This chapter describes the MN187XX23's noise cancellation function.

Chapter 9 Noise Cancellation

9

9-1 Overview

External interrupt pins IRQ1 and IRQ2 include a noise cancellation circuit, and can be used for remote control reception.

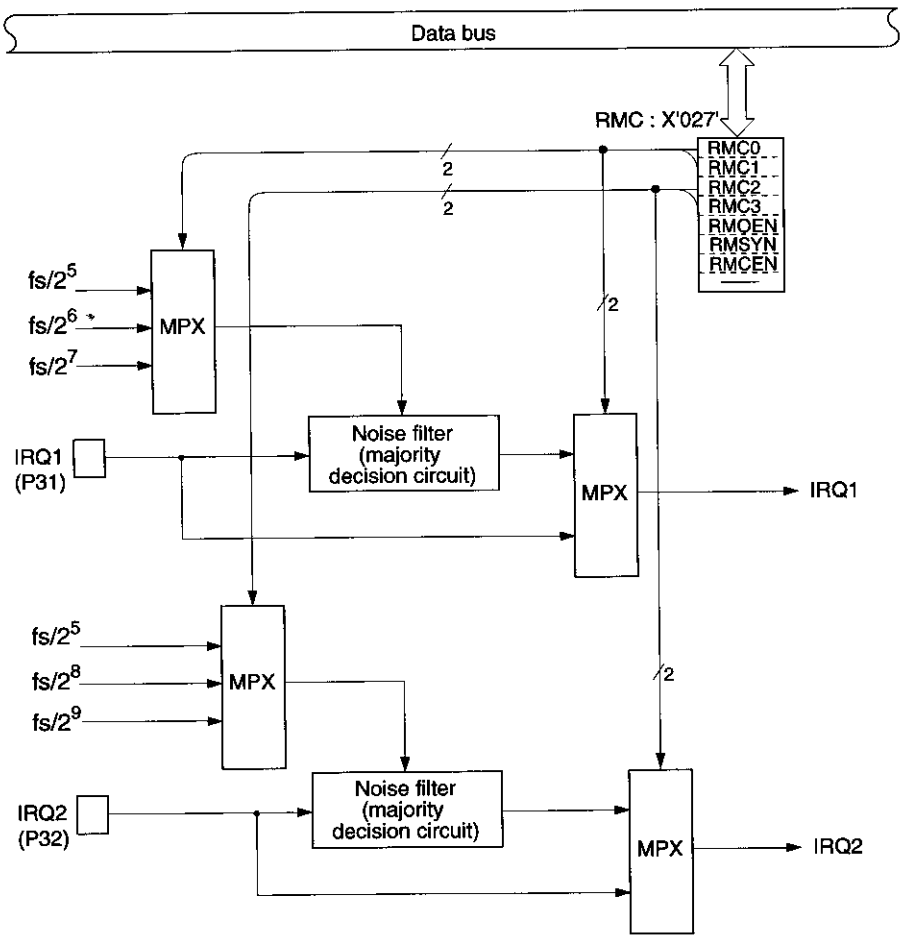


Figure 9-1-1 Noise Cancellation Circuit Block Diagram

IRQ1: External interrupt 1

IRQ2: External interrupt 2

Remote control/noise filter control register (RMC: X'027' R/W)
Bit 3: RMC3
Bit 2: RMC2
Bit 1: RMC1
Bit 0: RMC0

When the noise cancellation circuits are used, the waveform input from the IRQ1 pin is sampled based on the clock specified by bits RMC0 and RMC1 in the remote control/noise filter control register (RMC), and the waveform input from the IRQ2 pin is sampled based on the clock specified by bits RMC2 and RMC3. The signal is sampled four times, and the level in the majority is transferred to the CPU. If there are two low-level and two high-level samples, the previous level is held.

The noise cancellation circuits cannot be used in STOP mode or HALT mode.

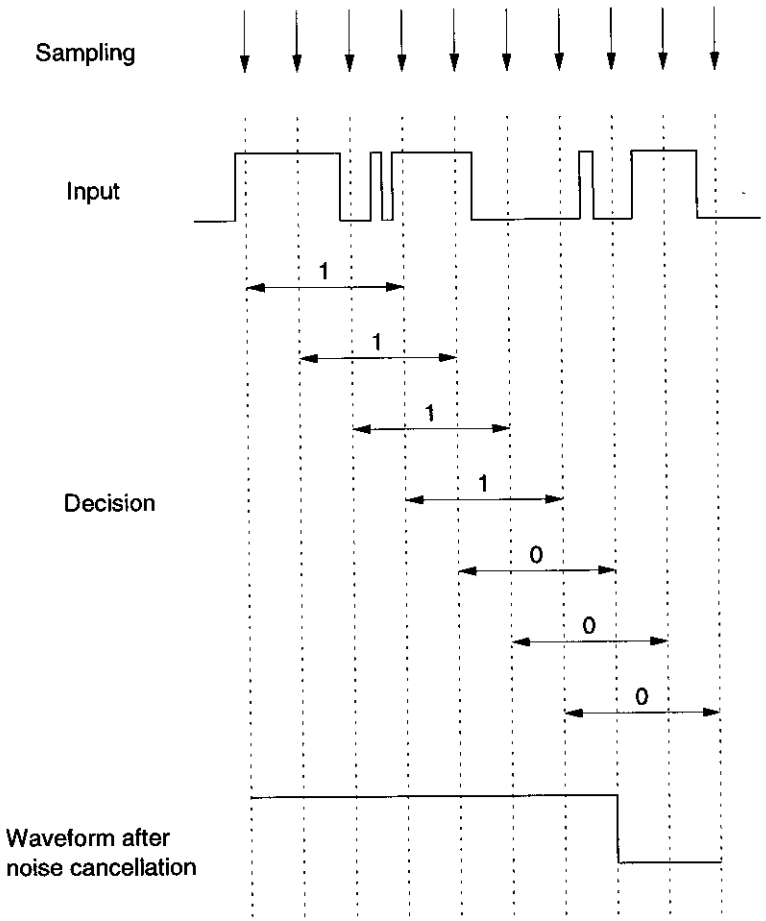


Figure 9-1-2 Example of Noise Filter Input and Output Waveforms

9-2 Remote Control/Noise Filter Control Register

9-2-1 Overview

One register is used to control the MN187XX23's noise filters: the remote control/noise filter control register (RMC) which selects the sampling clock. This register is located in RAM space.

Table 9-2-1 Noise Filter Control Register

Register Abbreviation	RAM Address	R/W	Register Name
RMC	X'027'	R/W	Remote control/noise filter control register

9-2-2 Remote Control/Noise Filter Control Register (RMC)

RMC is a 7-bit read/write register used for IRQ1 and IRQ2 pin noise filter control and remote control transmission function control. The IRQ1 and IRQ2 pins are used for remote control reception.

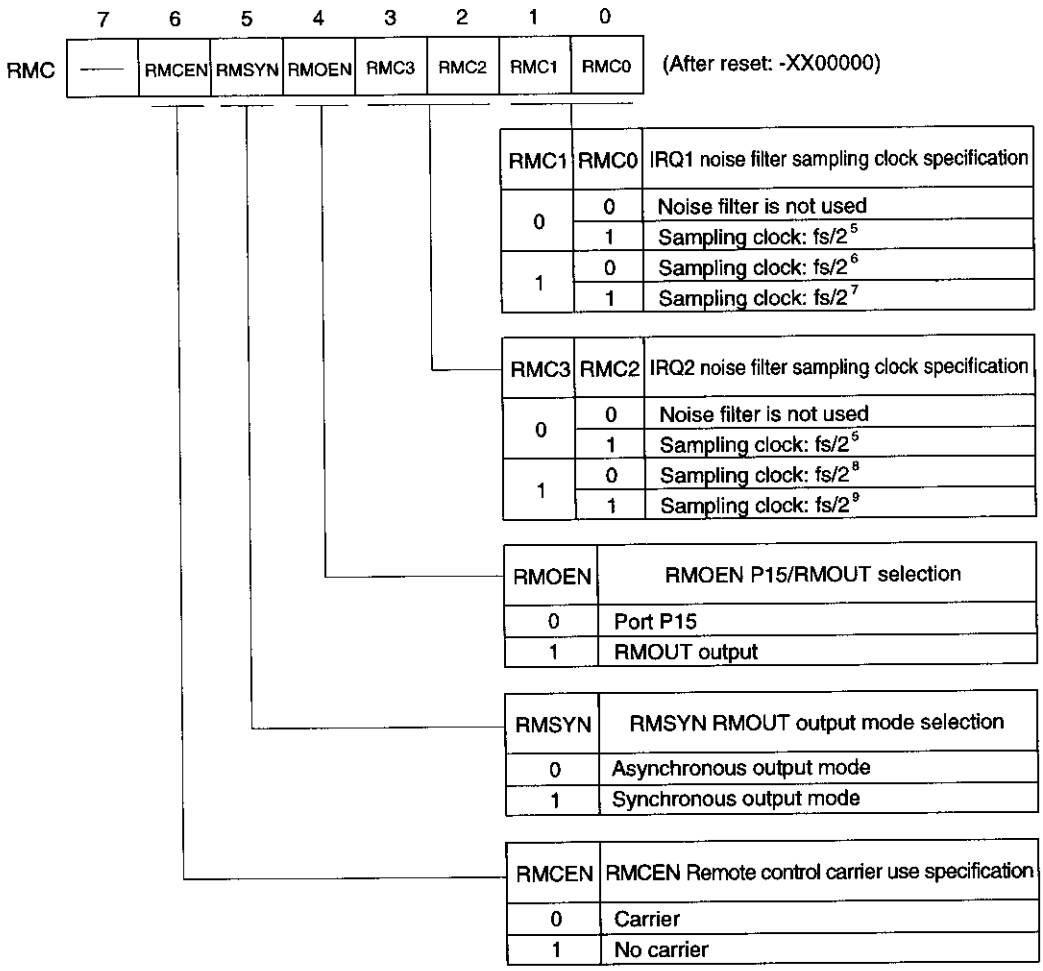
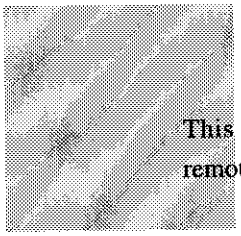


Figure 9-2-1 Remote Control/Noise Filter Control Register
(RMC: X'027' R/W)

At reset, bits of RMC0~RMC3 and RMOEN are cleared to 0, and the other bits are undefined.



This chapter describes the MN187XX23's remote control transmission functions.

Chapter 10 Remote Control Transmission

10

10-1 Overview

The MN187XX23 has an on-chip remote control transmission function that uses timers 4 and 5.

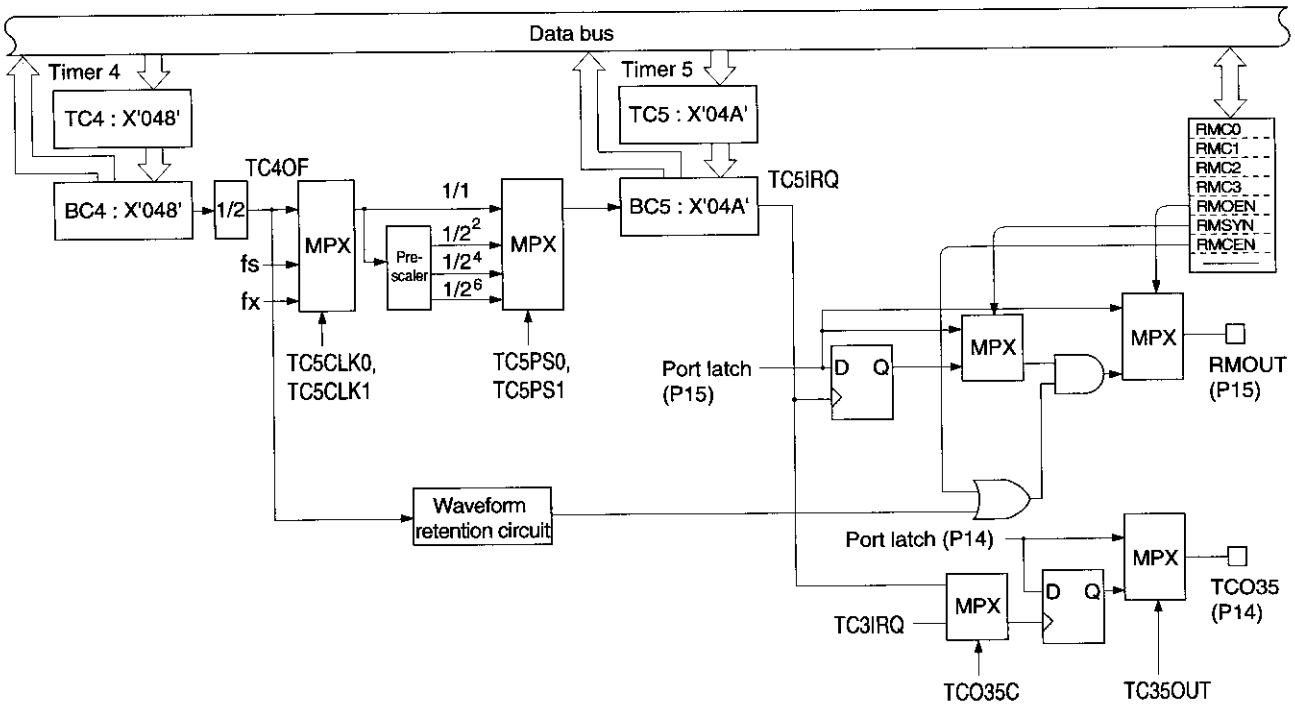


Figure 10-1-1 Remote Control Transmission Function Block Diagram

TC5LK0, 1:
Timer mode register 5 (TM5) bits 0, 1. Timer 5 clock source selection

TC5PS0, 1:
Timer mode register 5 (TM5) bits 2, 3. Timer 5 clock prescaler scaling factor selection

TC4OF:
Timer 4 internal output signal

TC5IRQ: Timer 5 interrupt signal

TC3IRQ: Timer 3 interrupt signal

TCO35C:
Timer mode register 5 (TM5) bit 5. Synchronous output clock specification

TC35OUT:
Timer mode register 5 (TM5) bit 6. Synchronous output specification

The remote control output signal is output from the RMOUT (P15) pin, and the remote control carrier signal is generated using timer 4 output. The carrier signal cycle is twice the timer 4 interrupt cycle.

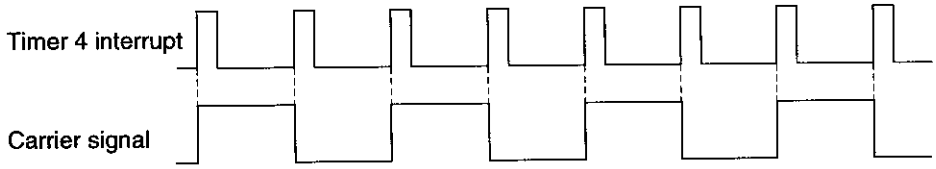


Figure 10-1-2 Remote Control Carrier Signal Generation

There are two ways of outputting the remote control output signal (RMOUT):

(1) Asynchronous output mode

Remote control output is performed asynchronously from the carrier signal, in accordance with the value set by the program in the P15 data latch. The remote control output signal is output from the RMOUT (P15) pin while the P15 data latch value is 1.

(2) Synchronous output mode

Carrier signal (timer 4 output signal TC4OF) pulses are counted using timer 5, and remote control output is performed in accordance with the value set by the program in the P15 data latch when a timer 5 interrupt is generated. As the remote control output data is changed in synchronization with the carrier signal, a high-precision remote control output waveform can be generated.

10-2 Remote Control Output Operation

(1) Asynchronous output mode with carrier signal

With this method, remote control output is controlled by having the P15 data latch rewritten by the program asynchronously from the carrier signal. After the register settings in Table 10-2-1 have been made, remote control output is performed by directly manipulating the P15 data latch value. The remote control output signal is output while the P15 data latch value is 1, and is not output while the value is 0.
A narrow carrier signal is not output when the P15 data latch value is changed over.

Table 10-2-1 Register Settings for Asynchronous Output Mode with Carrier Signal

Register	Bit	Mnemonic	Setting
RMC *	4	RMOEN	Set to 1 to designate P15 (RMOUT) pin as RMOUT pin.
	5	RMSYN	Cleared to 0 to select asynchronous output mode.
	6	RMCEN	Set to 1 to select "with carrier."
TM4	4,5	TC4CLK0, TC4CLK1	Select the timer 4 clock source. Remote control carrier cycle is timer 4 overflow cycle × 2.
	7	TC4EN	Set to 1 to make timer 4 operational.

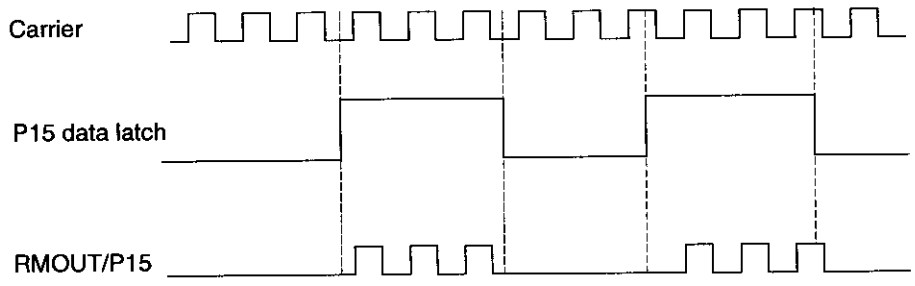


Figure 10-2-1 Remote Control Output (Asynchronous Output Mode with Carrier)

(2) Synchronous output mode with carrier signal

In this mode, remote control output is synchronized with timer 5, enabling a high-precision remote control output waveform to be obtained. In synchronous mode, remote control output is controlled by using timer 5 to count the carrier signal pulses generated by timer 4. The initial remote control output data (0 or 1) is set in the P15 data latch beforehand.

The timer 5 interrupt handler performs setting of the carrier signal count up to the next timer 5 interrupt, the next remote control output data, and the timer 5 interrupt enable flag.

Table 10-2-2 Register Settings for Synchronous Output Mode with Carrier Signal

Register	Bit	Mnemonic	Setting
RMC	4	RMOEN	Set to 1 to designate P15 (RMOUT) pin as RMOUT pin.
	5	RMSYN	Set to 1 to select synchronous output mode.
	6	RMCEN	Set to 1 to select "with carrier."
TM4	4,5	TC4CLK0, TC4CLK1	Select the timer 4 clock source. Remote control carrier cycle is timer 4 overflow cycle × 2.
	7	TC4EN	Set to 1 to make timer 4 operational.
TM5	0,1	TC5CLK0, TC5CLK1	Set to 00 to select timer 4 internal output (TC4OF) as clock source.
	2,3	TC5PS0, TC5PS1	Select timer 5 prescaler divider. Normally set to 00 to select 1/1.
	7	TC5EN	Set to 1 to make timer 5 operational.

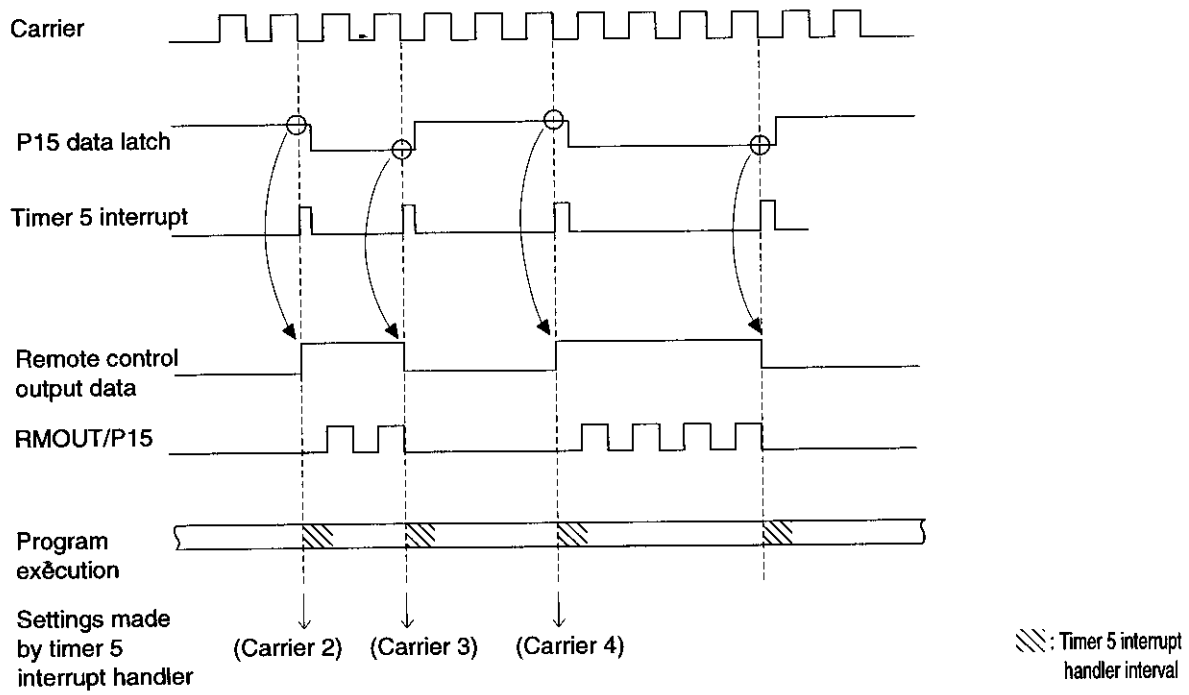


Figure 10-2-2 Remote Control Output (Synchronous Output Mode with Carrier)

(3) Remote control output without carrier signal

Remote control output without carrier can be used by clearing the RMCEN bit to 0 in the remote control/noise filter control register (RMC). In this case, the RMSYN bit asynchronous mode/synchronous mode setting selects whether or not output is to be performed in synchronization with timer 5 interrupts.

In asynchronous mode, after the register settings in Table 10-2-3 have been made, remote control output is performed by directly manipulating the P15 data latch value.

Table 10-2-3 Register Settings for Asynchronous Output Mode, No Carrier Signal

Register	Bit	Mnemonic	Setting
RMC	4	RMOEN	Set to 1 to designate P15 (RMOUT) pin as RMOUT pin.
	5	RMSYN	Cleared to 0 to select asynchronous output mode.
	6	RMCEN	Cleared to 0 to select "no carrier."

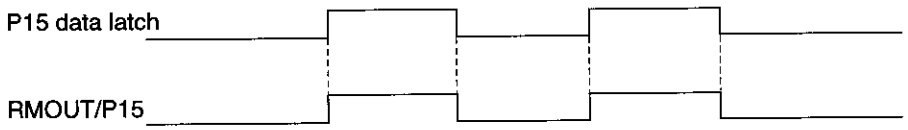


Figure 10-2-3 Remote Control Output (Asynchronous Output Mode, No Carrier)

In synchronous mode, the timer 5 interrupt handler performs setting of the next timer 5 interrupt interval and writes the next remote control output data to the P15 data latch.

Table 10-2-4 Register Settings for Synchronous Output Mode, No Carrier Signal

Register	Bit	Mnemonic	Setting
RMC	4	RMOEN	Set to 1 to designate P15 (RMOUT) pin as RMOUT pin.
	5	RMSYN	Set to 1 to select synchronous output mode.
	6	RMCEN	Cleared to 0 to select "no carrier."
TM5	0,1	TC5CLK0, TC5CLK1	Select the timer 5 clock source and prescaler divider.
	2,3	TC5PS0, TC5PS1	
	7	TC5EN	Set to 1 to make timer 5 operational.

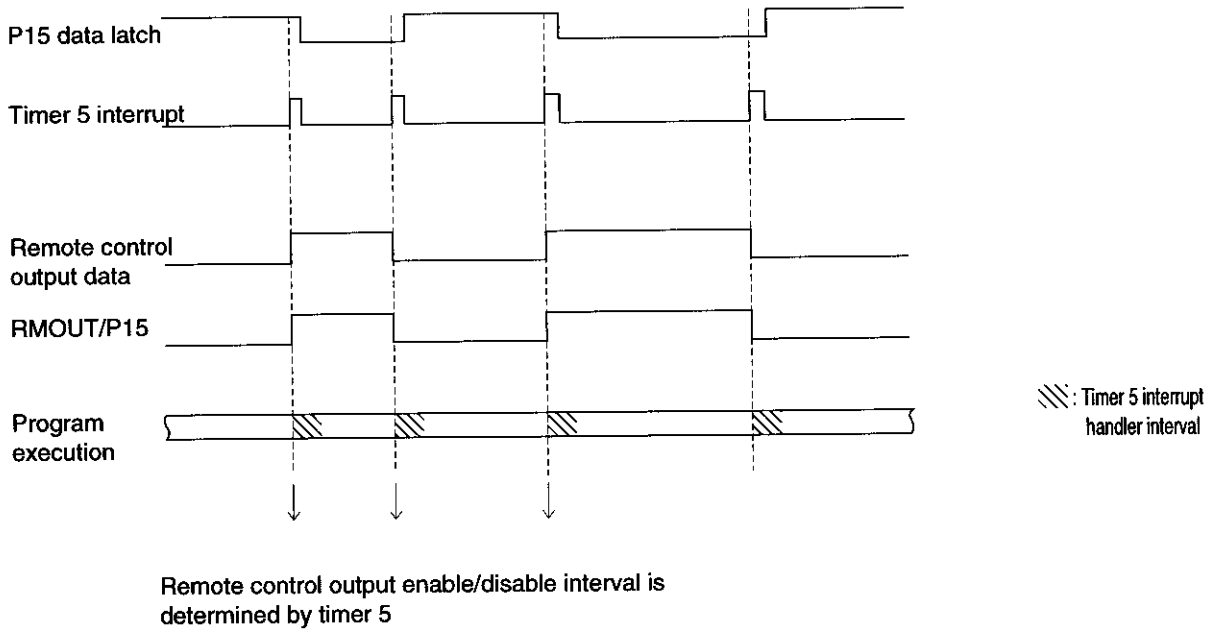


Figure 10-2-4 Remote Control Output (Synchronous Output Mode, No Carrier)

10-3 Remote Control/Noise Filter Control Register (RMC)

10-3-1 Overview

One register is used to control the MN187XX23's remote control transmission function: the remote control/noise filter control register (RMC) which controls port/remote control output switching, synchronous mode/asynchronous mode selection, and whether the remote control carrier is used. This register is located in RAM space.

Table 10-3-1 Remote Control Control Register

Register Abbreviation	RAM Address	R/W	Register Name
RMC	X'027'	R/W	Remote control/noise filter control register

10-3-2 Remote Control/Noise Filter Control Register (RMC)

RMC is a 7-bit read/write register used for IRQ1 and IRQ2 pin noise filter control and remote control transmission function control. The IRQ1 and IRQ2 pins are used for remote control reception.

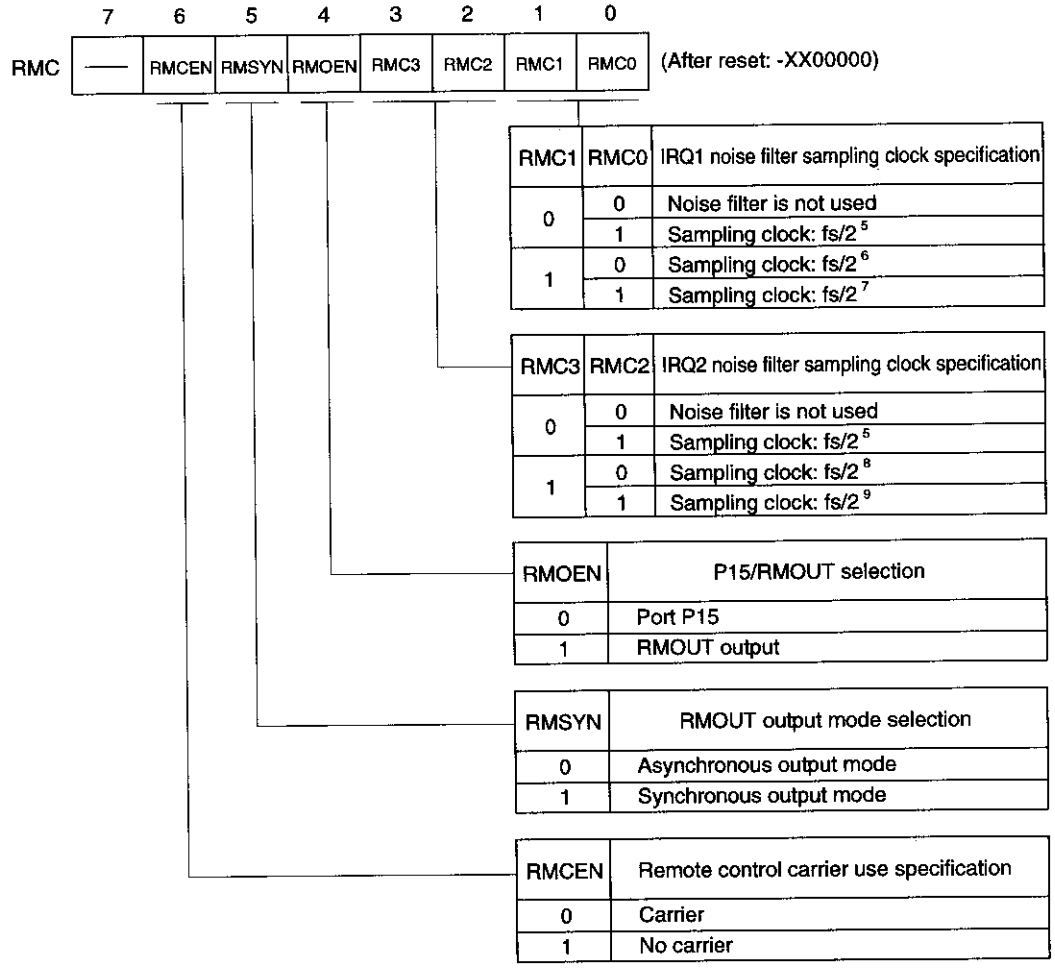
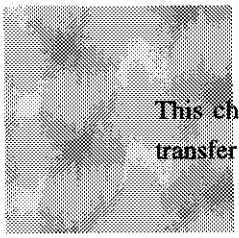


Figure 10-3-1 Remote Control/Noise Filter Control Register (RMC: X'027' R/W)

At reset, bits of RMC0~RMC3 and RMOEN are cleared to 0, and the other bits are undefined.



This chapter describes the MN187XX23's automatic data transfer (ADT) function.

Chapter 11 Automatic Data Transfer

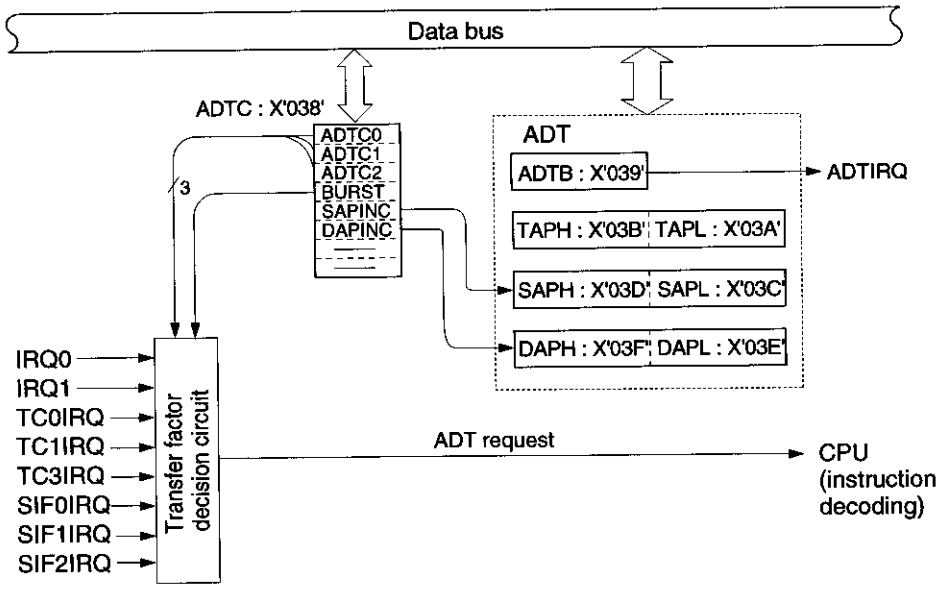
11

11-1 Overview

The MN187XX23 includes an automatic data transfer function that enables data transfer inside the RAM space to be executed by hardware. Transfer is started by a transfer factor interrupt, and one byte is transferred each time a transfer factor is generated. Transfer factor interrupts can be selected from two external interrupts, timer 0, 1, and 3 interrupts, and serial interface 0, 1, and 2 interrupts.

A burst transfer mode is also provided, in which data within the specified range is transferred consecutively without regard to a transfer factor.

Figure 11-1-1 Automatic Data Transfer Block Diagram



ADT: Automatic data transfer register
ADTB: Transfer byte counter
TAPH, TAPL: Data transfer target address register
SAPH, SAPL: Data transfer source address register
DAPH, DAPL: Data transfer destination address register
ADTC: Automatic data transfer control register
ADTIRQ: Automatic data transfer interrupt

IRQ0: External interrupt 0
IRQ1: External interrupt 1
TC0IRQ: Timer 0 interrupt
TC1IRQ: Timer 1 interrupt
TC3IRQ: Timer 3 interrupt
SIF0IRQ: Serial interface 0 interrupt
SIF1IRQ: Serial interface 1 interrupt
SIF2IRQ: Serial interface 2 interrupt

11-2 Automatic Data Transfer

11-2-1 Overview

ADTC: Automatic data transfer control register (ADTC: X'038' R/W) Bit 3: BURST

ADTB: Transfer byte counter (ADTB: X'039' R/W)

Automatic data transfer is specified by means of the automatic data transfer control register (ADTC), transfer byte counter (ADTB), and three kinds of transfer address pointer.

The transfer byte counter (ADTB) specify any byte number of automatic data transfer between 1 and 255 bytes. At the end of the specified transfer, an automatic data transfer interrupt (ADTIRQ) is generated.

To prevent inadvertently overwriting data, a three-address transfer method is used which allows transfer to be performed without restrictions as specified by the settings. Use of address incrementing or a fixed address can be specified for two of the three addresses (the transfer source address and transfer destination address).

11-2-2 Automatic Data Transfer

The transfer procedure is as follows.

Data transfer target address register (TAPL: X'03A' R/W, TAPH: X'03B' R/W)

Data source target address register (SAPL: X'03C' R/W, SAPH: X'03D' R/W)

Data transfer destination address register (DAPL: X'03E' R/W, DAPH: X'03F' R/W)

① Set the three transfer address pointers: the data transfer target address register (TAPL, TAPH), data transfer source address register (SAPL, SAPH), and data transfer destination address register (DAPL, DAPH).

② Use the automatic data transfer control register (ADTC) to specify address incrementing and the transfer factor or continuous transfer.

③ Set the number of transfer bytes in the transfer byte counter (ADTB).

④ When factor transfer mode is used, data transfer is not performed until the specified transfer factor is generated.
When burst transfer mode is used, setting the BURST bit in the automatic data transfer control register (ADTC) to 1 selects burst transfer mode and simultaneously starts the transfer.

11-2 Automatic Data Transfer

11-2-1 Overview

ADTC: Automatic data transfer control register (ADTC: X'038' R/W) Bit 3: BURST

ADTB: Transfer byte counter (ADTB: X'039' R/W)

Automatic data transfer is specified by means of the automatic data transfer control register (ADTC), transfer byte counter (ADTB), and three kinds of transfer address pointer.

The transfer byte counter (ADTB) specify any byte number of automatic data transfer between 1 and 255 bytes. At the end of the specified transfer, an automatic data transfer interrupt (ADTIRQ) is generated.

To prevent inadvertently overwriting data, a three-address transfer method is used which allows transfer to be performed without restrictions as specified by the settings. Use of address incrementing or a fixed address can be specified for two of the three addresses (the transfer source address and transfer destination address).

11-2-2 Automatic Data Transfer

The transfer procedure is as follows.

Data transfer target address register (TAPL: X'03A' R/W, TAPH: X'03B' R/W)

Data source target address register (SAPL: X'03C' R/W, SAPH: X'03D' R/W)

Data transfer destination address register (DAPL: X'03E' R/W, DAPH: X'03F' R/W)

① Set the three transfer address pointers: the data transfer target address register (TAPL, TAPH), data transfer source address register (SAPL, SAPH), and data transfer destination address register (DAPL, DAPH).

② Use the automatic data transfer control register (ADTC) to specify address incrementing and the transfer factor or continuous transfer.

③ Set the number of transfer bytes in the transfer byte counter (ADTB).

④ When factor transfer mode is used, data transfer is not performed until the specified transfer factor is generated.
When burst transfer mode is used, setting the BURST bit in the automatic data transfer control register (ADTC) to 1 selects burst transfer mode and simultaneously starts the transfer.

⑤ In automatic data transfer, the following processing is performed by hardware:

1. The contents of the address indicated by the data transfer target address register (TAP) are stored at the address indicated by the data transfer destination address register (DAP).
2. The contents of the address indicated by the data transfer source address register (SAP) are stored at the address indicated by the data transfer target address register (TAP).
3. If incrementation is specified for the data transfer source address register (SAP) and data transfer destination address register (DAP), SAP and DAP are incremented by 1 after the above transfer is completed.

[☞ Figure 11-2-1 Three-Address Transfer Method]

⑥ When one byte transfer is completed, the transfer byte counter (ADTB) is decremented. In factor transfer mode, the processing in step ⑤ is repeated each time a transfer factor is generated, and an automatic data transfer interrupt (ADTIRQ) is generated when ADTB reaches 0.

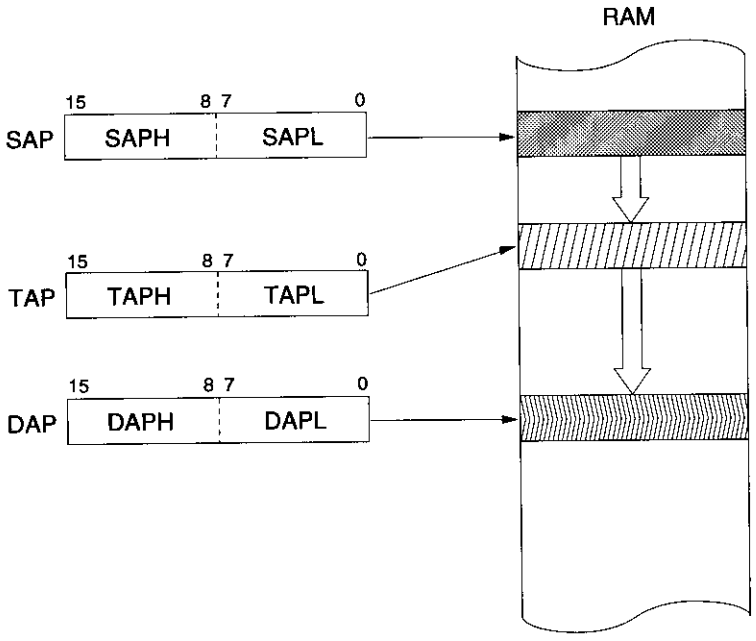


Figure 11-2-1 Three-Address Transfer Method

11-2-3 Examples of ADT Operation

Use of the automatic data transfer function simplifies execution of various kinds of data processing within the RAM space.

Some examples are given below. The descriptions have been kept brief for the sake of clarity; for fuller details, refer to the descriptions of the registers noted in the margin.

ADTC: Automatic data transfer control register (ADTC: X'038' R/W)

ADTB: Transfer byte counter (ADTB: X'039' R/W)

TAP: Data transfer target address register (TAPL: X'03A' R/W, TAPH: X'03B' R/W)

SAP: Data source target address register (SAPL: X'03C' R/W, SAPH: X'03D' R/W)

DAP: Data transfer destination address register (DAPL: X'03E' R/W, DAPH: X'03F' R/W)

(1) Transfer from area A (i to i+n) to area B (j to j+n)

① Set the address of RAM used for temporary storage in TAP.

② Set address (i) in SAP and address (j–1) in DAP.

③ Specify the transfer factor and SAP and DAP incrementing in ADTC.

④ Set the number of transfer bytes (n+2) in ADTB, starting automatic transfer.

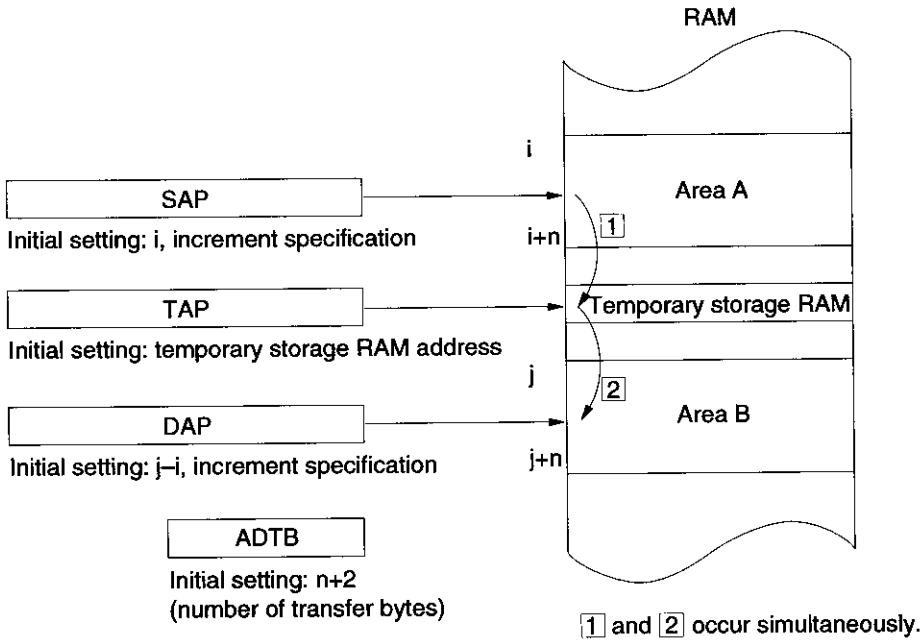


Figure 11-2-2 Transfer from Area A (i to i+n) to Area B (j to j+n)

In this example, area (j–1) data is overwritten.

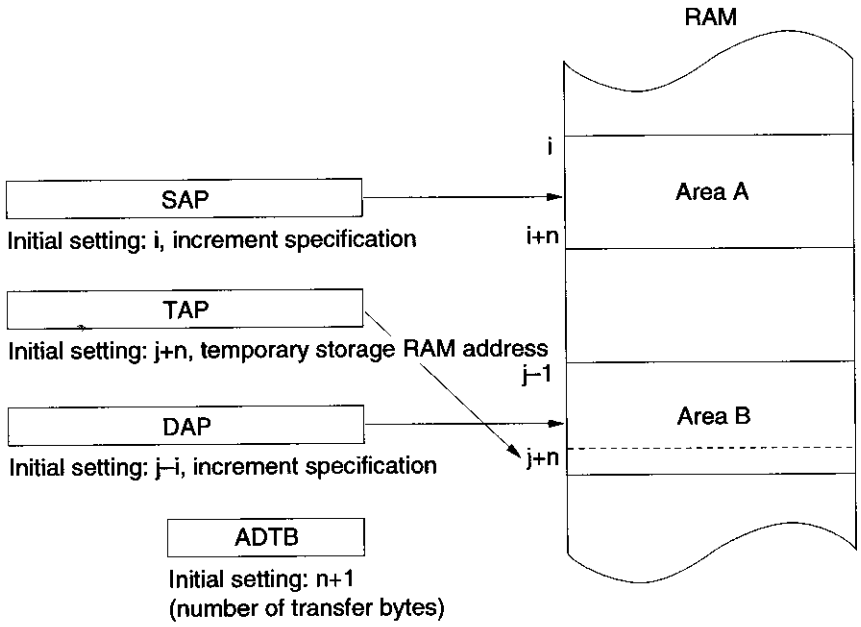
(2) Transfer from area A (i to i+n) to area B (j–1 to j+n)

① Set address (i+n (transfer destination end address)) in TAP.

② Set address (i) in SAP, and address (j–1) in DAP.

③ Specify the transfer factor and SAP and DAP incrementing in ADTC.

④ Set the number of transfer bytes (n+1) in ADTB, starting automatic transfer.



In this example, area (j–1) data is overwritten.

Figure 11-2-3 Transfer from Area A (i to i+n) to Area B (j–1 to j+n)

(3) Transfer from area A (i to i+n) to area B (j)

① Set B address (j) in TAP.

② Set address (i) in SAP.

③ As DAP is not used, specify any RAM address.

④ Specify the transfer factor, SAP incrementing, and DAP non-incrementing in ADTC.

⑤ Set the number of transfer bytes (n+1) in ADTB, starting automatic transfer.

For "any RAM address," make sure that an installed RAM address is set.

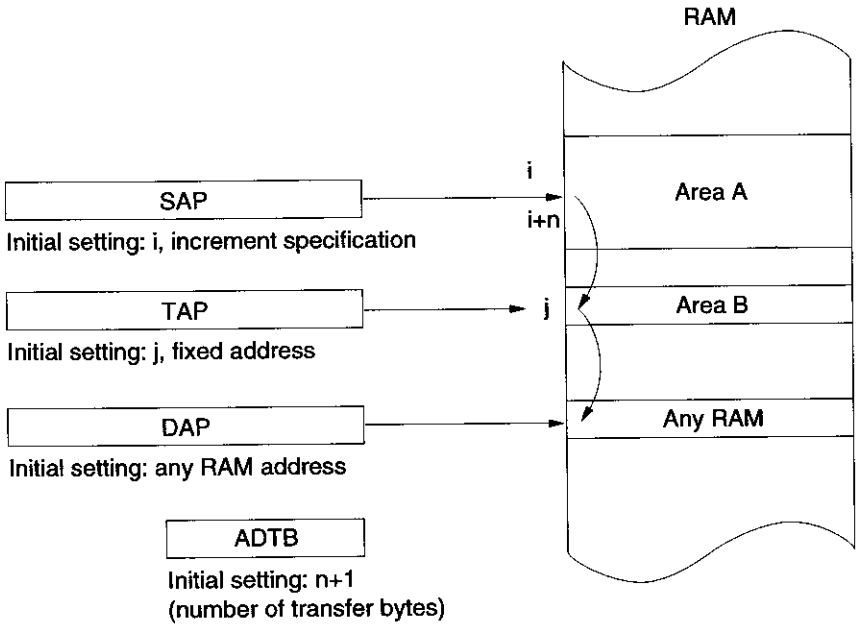


Figure 11-2-4 Transfer from Area A (i to i+n) to Area B (j)

(4) Transfer from area A (i) to area B (j to j+n)

① Set A address (i) in TAP.

② Set address (j) in DAP.

③ Set A address (i), or the address of RAM with the same contents as A, in SAP. If A address (i) is set, A will be read twice and written to once.

④ Specify the transfer factor, SAP non-incrementing, and DAP incrementing in ADTC.

⑤ Set the number of transfer bytes (n+1) in ADTB, starting automatic transfer.

ADTC: Automatic data transfer control register (X'038' R/W)

ADTB: Transfer byte counter (X'039' R/W)

TAP: Data transfer target address register (TAPL: X'03A' R/W, TAPH: X'03B' R/W)

SAP: Data source target address register (SAPL: X'03C' R/W, SAPH: X'03D' R/W)

DAP: Data transfer destination address register (DAPL: X'03E' R/W, DAPH: X'03F' R/W)

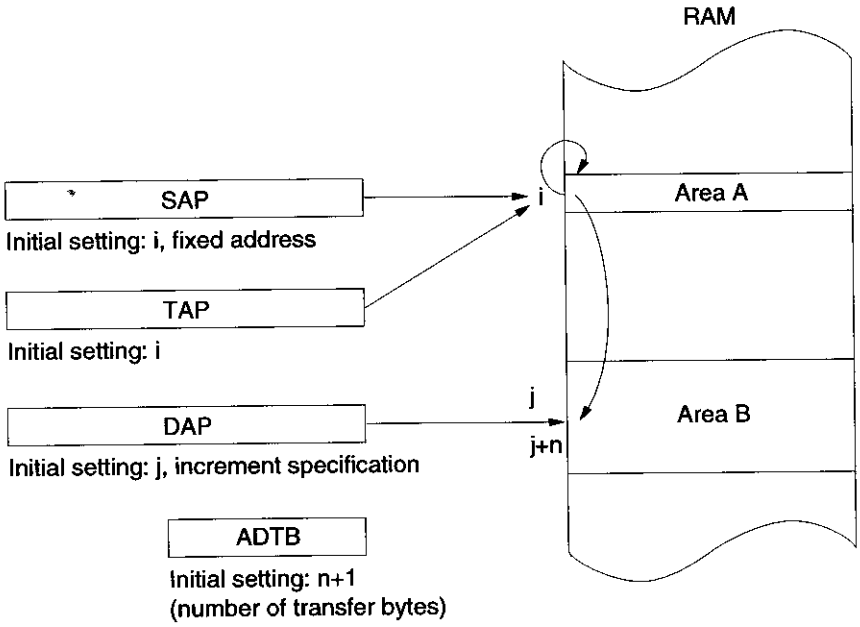


Figure 11-2-5 Transfer from Area A (i) to Area B (j to j+n)

(5) Data exchange between area A (i) and area B (j to j+n)

① Set A address (i) in TAP.

② Set address (j) in SAP and DAP.

③ Specify the transfer factor, and SAP and DAP incrementing, in ADTC.

④ Set the number of transfer bytes (n+1) in ADTB, starting automatic transfer.

1 and 2 occur simultaneously.

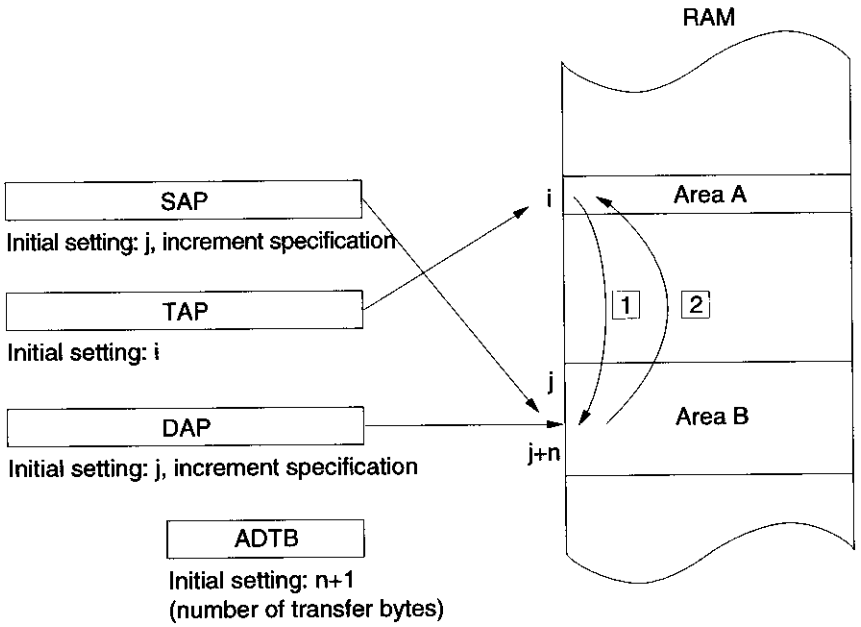


Figure 11-2-6 Data Exchange between Area A (i) and Area B (j to j+n)

11-3 Automatic Data Transfer Interrupt

With the automatic data transfer function, when the transfer byte counter (ADTB) reaches 0 and transfer is terminated, the automatic data transfer interrupt request flag (IFADT) is set to 1 in interrupt request flag register 1 (IF1). If the automatic data transfer interrupt enable flag (IEADT) in interrupt enable flag register 1 (IE1) is 1 at this time, an interrupt request is issued to the CPU. [☞ "2-8 Interrupt Controller"]

Interrupt request flag register 1
(IF1: X'013' R/W)
Bit 6: IFADT

Interrupt enable flag register 1
(IE1: X'015' R/W)
Bit 6: IEADT

In the MN187XX23, if the transfer byte counter (ADTB) value is changed to X'00' when its value is manipulated by the program, the IFADT bit will be set to 1 in interrupt request flag register 1 (IF1). If interrupts are enabled at this time, an automatic data transfer interrupt (ADTIRQ) will be accepted. To prevent inadvertent interrupt generation, use the following setting procedure when the transfer byte counter (ADTB) is cleared to X'00' by the program.

<Setting Procedure>

1. Clear the IEADT bit in interrupt enable flag register 1 (IE1) to disable the ADT interrupt.
2. Clear the transfer byte counter (ADTB).
3. Clear the IFADT bit in interrupt request flag register 1 (IF1).

Example:

```
CLR   DF
CLR   (IE1)IEADT
MOV   (ADTB),0
CLR   (IF1)IFADT
```

11-4 Automatic Data Transfer Control Registers

11-4-1 Overview

Eight registers are used to control the MN187XX23's automatic data transfer function: the automatic data transfer control register (ADTC) which controls the operation of the ADT function, the transfer byte counter (ADTB) which specifies the number of transfer bytes, and the transfer address pointers (TAPL, TAPH, SAPL, SAPH, DAPL, DAPH) which specify the transfer target, transfer source, and transfer destination addresses. These registers are all located in RAM space.

Table 11-4-1 Automatic Data Transfer Control Registers

Register Abbreviation	RAM Address	R/W	Register Name
ADTC	X'038'	R/W	Automatic data transfer control register
ADTB	X'039'	R/W	Transfer byte counter
TAPL	X'03A'	R/W	Data transfer target address register lower 8 bits
TAPH	X'03B'	R/W	Data transfer target address register upper 8 bits
SAPL	X'03C'	R/W	Data transfer source address register lower 8 bits
SAPH	X'03D'	R/W	Data transfer source address register upper 8 bits
DAPL	X'03E'	R/W	Data transfer destination address register lower 8 bits
DAPH	X'03F'	R/W	Data transfer destination address register upper 8 bits

11-4-2 Automatic Data Transfer Control Register (ADTC)

ADTC is a 6-bit read/write register which controls the automatic data transfer (ADT) function.

	7	6	5	4	3	2	1	0	
ADTC	—	—	DAPINC	SAPINC	BURST	ADTC2	ADTC1	ADTC0	(After reset: --000000)

ADTC2	ADTC1	ADTC0	Transfer factor specification
0	0	0	External interrupt 0 (IRQ0)
		1	External interrupt 1 (IRQ1)
	1	0	Timer 0 interrupt (TC0IRQ)
		1	Timer 1 interrupt (TC1IRQ)
1	0	0	Serial 2 interrupt (SIF2IRQ)
		1	Timer 3 interrupt (TC3IRQ)
	1	0	Serial 0 interrupt (SIF0IRQ)
		1	Serial 1 interrupt (SIF1IRQ)

BURST	Factor transfer/continuous transfer specification
0	Transfer using transfer factor: 1 byte transferred on generation of factor
1	Continuous transfer: Number of bytes set in ADTB transferred consecutively without regard to transfer factor

SAPINC	Transfer source address pointer increment specification
0	SAP incremented by 1 after 1-byte transfer
1	SAP not incremented

DAPINC	Transfer destination address pointer increment specification
0	DAP incremented by 1 after 1-byte transfer
1	DAP not incremented

Figure 11-4-1 Automatic Data Transfer Control Register (ADTC: X'038' R/W)

In a reset, all bits are cleared to 0.

When BURST is set to 1, automatic transfer is started regardless of the setting of ADTC2~ADTC0, and the number of bytes set in the transfer byte counter (ADTB) are transferred consecutively. When BURST is cleared to 0, a one-byte transfer is performed for one occurrence of the transfer factor specified by ADTC2~ADTC0.

11-4-3 Transfer Byte Counter (ADTB)

ADTB is an 8-bit read/write register used to set the total number of bytes in a data transfer. When one byte is transferred, ADTB is decremented by 1, and when its value reaches X'00' an automatic data transfer interrupt (ADTIRQ) is generated and transfer is terminated.

If ADTB = X'00', data transfer will not be performed when a transfer factor is generated (or the BURST bit is set to 1 in ADTC).

	7	6	5	4	3	2	1	0	
ADTB	ADTB7	ADTB6	ADTB5	ADTB4	ADTB3	ADTB2	ADTB1	ADTB0	(After reset: 00000000)

Figure 11-4-2 Transfer Byte Counter (ADTB: X'039' R/W)

In a reset, all bits are cleared to 0.

In the MN187XX23, if the transfer byte counter (ADTB) is set before the automatic data transfer control register (ADTC) when using automatic data transfer, automatic transfer may be performed in accordance with the previous settings (transfer factor and transfer method). This may result in generation of an unwanted automatic data transfer interrupt (ADTIRQ). To prevent inadvertent interrupt generation, use the following register setting procedure.

<Setting Procedure>

1. Set the data transfer source address register (SAPH, SAPL), data transfer target address register (TAPH, TAPL), and data transfer destination address register (DAPH, DAPL).
2. Set the transfer conditions in the automatic data transfer control register (ADTC).
3. Lastly, write the number of transfer bytes into the transfer byte counter (ADTB), starting automatic data transfer.

11-4-4 Transfer Address Pointers

The transfer address pointers are three 16-bit read/write registers that indicate the transfer addresses.

A three-address transfer method is used to prevent data from being inadvertently overwritten. In a data transfer, the contents of the address indicated by the data transfer target address register (TAP) are transferred to the address indicated by the data transfer destination address register (DAP), and at the same time, the contents of the address indicated by the data transfer source address register (SAP) are transferred to the address indicated by TAP.

(1) Data transfer target address register (TAPH, TAPL)

Specifies the transfer target RAM address.

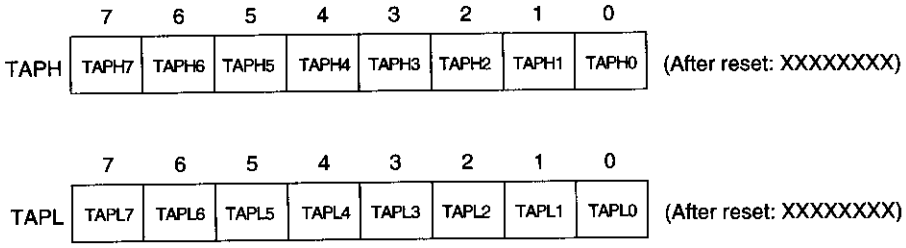


Figure 11-4-3 Data Transfer Target Address Register
(TAPH: X'03B' R/W, TAPL: X'03A' R/W)

(2) Data transfer source address register (SAPH, SAPL)

Automatic data transfer control register (ADTC: X'038' R/W)
Bit 5: DAPINC
Bit 4: SAPINC

Specifies the transfer source address. Incrementing after each transfer can be specified with the SAPINC bit in the automatic data transfer control register (ADTC). At reset, all bits are undefined.

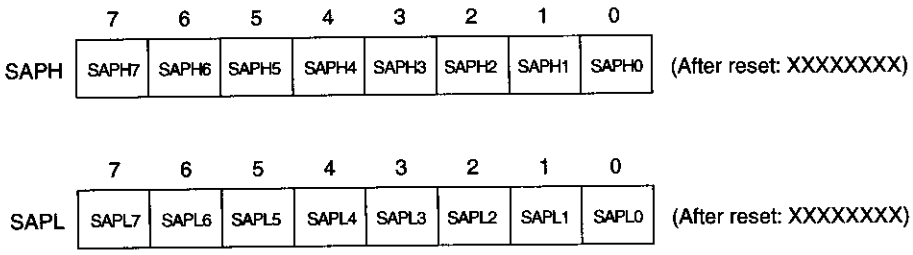


Figure 11-4-4 Data Transfer Source Address Register
(SAPH: X'03D' R/W, SAPL: X'03C' R/W)

(3) Data transfer destination address register (DAPH, DAPL)

Specifies the transfer destination address. Incrementing after each transfer can be specified with the DAPINC bit in the automatic data transfer control register (ADTC). At reset, all bits are undefined.

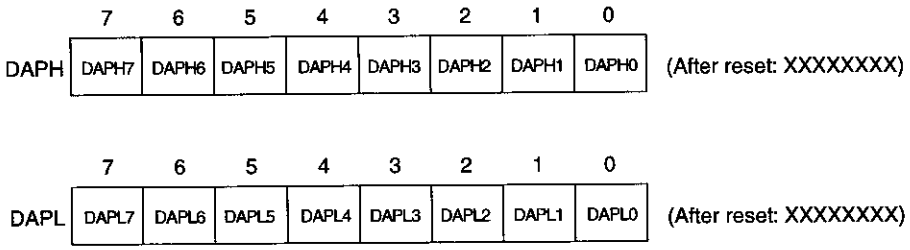


Figure 11-4-5 Data Transfer Destination Address Register
(DAPH: X'03F' R/W, DAPL: X'03E' R/W)

11-5 Notes

(1) When one-byte data transfer processing is performed, the user program halts during 5 machine cycles. That means, in continuous transfer mode, processing halts during continuously 5 × (number of transfer bytes) machine cycles. No interrupts except a reset are accepted during data transfer.

(2) The automatic data transfer interrupt (ADTIRQ) is generated on completion of the write to RAM. Caution is therefore necessary if a serial buffer is specified by TAP, since a serial transfer may still be in progress when the automatic data transfer interrupt (ADTIRQ) is generated.

If simultaneous transmit and receive operations are performed on the serial interface, and a serial buffer is specified by the data transfer target address register (TAP), transfer is performed as shown below when a serial interrupt is specified as the transfer factor.

Data transfer target address register (TAPL: X'03A' R/W, TAPH: X'03B' R/W)

Data source target address register (SAPL: X'03C' R/W, SAPH: X'03D' R/W)

Data transfer destination address register (DAPL: X'03E' R/W, DAPH: X'03F' R/W)

① When the serial interface completes a transfer, a serial interrupt is generated and the automatic data transfer function is activated.

② The contents of the serial buffer are written to the RAM indicated by the data transfer destination address register (DAP), and the contents of the RAM indicated by the data transfer source address register (SAP) are written to the serial buffer.

③ An automatic data transfer interrupt (ADTIRQ) is generated, and preparations for the next serial transfer are made in the interrupt handling routine.

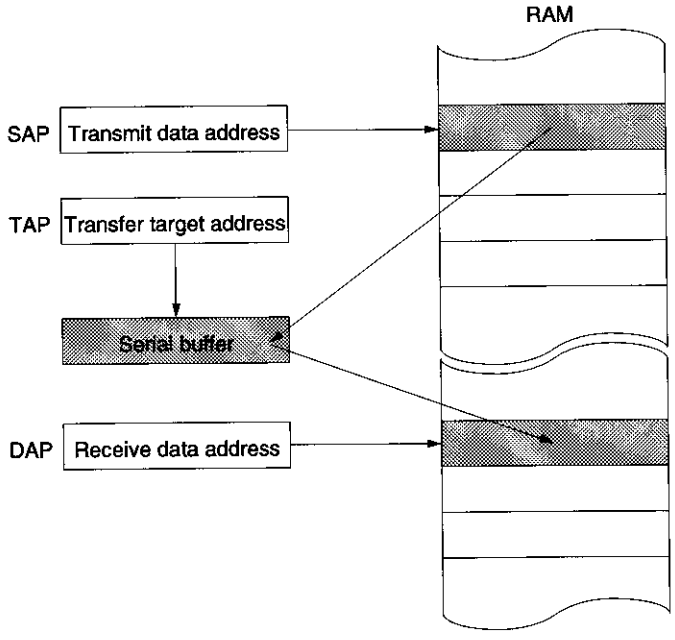
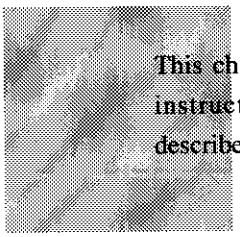


Figure 11-5-1 Example of Serial Transfer Using Automatic Data Transfer

(3) In factor transfer mode, automatic data transfer is started only by generation of an interrupt specified as a transfer factor. Transfer cannot be performed by having software set the interrupt request flag for the relevant interrupt.



This chapter contains tables showing the MN187XX23's instruction set, operand map, and registers, and also describes the MN18P76423 version with on-chip EPROM.

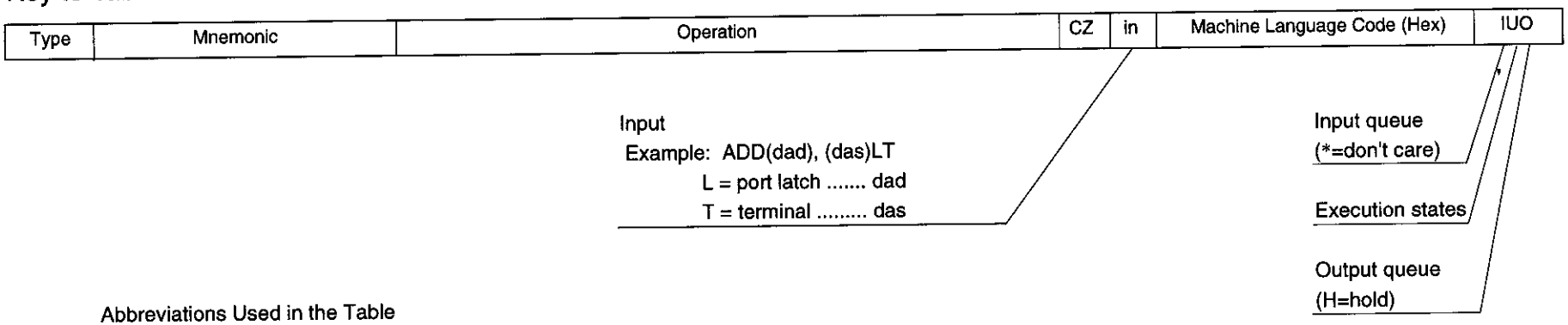
Chapter 12 Appendix

12

12-1 MN1870 Series Instruction Set

The MN1870 Series instruction set is summarized in table 12-1-1, and fuller details are given in table 12-1-2.

Key to Table 12-1-2



Abbreviations Used in the Table

XP, YP, SP, XPh, XPl, YPh, YPl	: Pointer register
(XP), (YP), (SP), (da), (das), (dad), (da1), (da2)	: RAM data ( )
imm, imm4, mask, imm16	: Immediate data
bp, bps, bpd	: Bit position
da, das, dad, da1, da2	: Direct address
DF	: Direct flag When DF=0, RAM address upper 8 bits=0 When DF=1, RAM address upper 8 bits=XPh or YPh
CF	: Carry/borrow flag
ZF	: Zero flag
un, ln	: Upper/lower nibble (4 bits)
IP, IPh, IPm, IPl	: Instruction pointer h (4 bits), m (4 bits), l (8 bits)
FS	: Flag status byte (CZ, ZF, DF)
$label, label, /label	: Branch destination address

Table 12-1-1 MN1870 Instruction Set ①

Type	Mnemonic	
Data transfer	MOV	(DOUBLE OPERAND)
		(XP), (da)
		(da), (YP)
		XPl, imm
		XPh, imm
		YPl, imm
		YPh, imm
		XPl, (da)
		YPl, (da)
		(da), XPl
		(da), YPl
	MOVL	(dad), (das)
		(XP), (YP)
		XP, (da)
		(da), XP
		YP, (da)
		(da), YP
		(XP), imm16
		(da), imm16
		XP, imm16
		YP, imm16
	RDTBL	(XP), (YP)
	MOV1	(dad) bpd, (das) bps
	MOV1N	(dad) bpd, (das) bps
	XCH	XP, YP
		XP, SP
		(XP), (YP)
		(da1), (da2)
	XCH4	(SINGLE OPERAND)
	CLR	(SINGLE OPERAND)
		(da) bp
		(XP) bp
		DF
		CF
	SET	(da) bp
		(XP) bp
		DF
		CF

Type	Mnemonic	
Arithmetic	ADDC	(DOUBLE OPERAND)
	ADDD	(DOUBLE OPERAND)
	ADDR	XPl, XPl, imm
		YPl, YPl, imm
		XPl, (da), imm
		YPl, (da), imm
	INC	(SINGLE OPERAND)
		XP
		YP
	INCL	(da) (Note)
	DEC	(SINGLE OPERAND)
		XP
		YP
	DECL	(da) (Note)
	SUBC	(DOUBLE OPERAND)
	SUBD	(DOUBLE OPERAND)
	MUL	(XP)
		(XP), imm
	DIV	(XP)
		(XP), imm
	CMP	(DOUBLE OPERAND)
		XPl, imm
		YPl, imm
	CMPL	(XP), (YP)
	CMPM	(XP) mask, imm
		(da) mask, imm
		(XP) mask, (YP) mask
		(dad) mask, (das) mask
Logic	NOT	(SINGLE OPERAND)
	ROL	(SINGLE OPERAND)
	ROR	(SINGLE OPERAND)
	AND	(DOUBLE OPERAND)
	OR	(DOUBLE OPERAND)
	XOR	(DOUBLE OPERAND)

Type	Mnemonic	
Subroutine call	CALL	label
		/label
		XP
	RET	
	RETI	
Unconditional branch	BR	$label
		label
		/label
		XP
Unconditional skip	SKIP	
conditional branch	BC	$label
	BZ	$label
	BLE	$label
	BNC	$label
	BNZ	$label
	BGT	$label
Test and conditional branch	T1BNZ	(da) bp, $label
		(XP) bp, $label
	T1BZ	(da) bp, $label
		(XP) bp, $label
Compare and conditional branch	CMPBNE	(XP), imm, $label
		(da), imm, $label
	CMPBE	(XP), imm, $label
		(da), imm, $label
Loop	LOOP	$label
	RLOOP	XP, $label
		YP, $label
Stack manipulation	PUSH	(SINGLE OPERAND)
		XP
		YP
		imm
		FS
	POP	(SINGLE OPERAND)
		XP
		YP
		FS

Type	Mnemonic
Other	STOP
	HALT
	NOP

<Legend>

(SINGLE OPERAND): (XP)
(da)

(DOUBLE OPERAND): (XP), (YP)
(XP), imm
(dad), (das)
(da), imm

$label: Relative address specification (+127 to −128)

label: Short address specification (absolute address branch within 4-Kbyte page)

/label: Long address specification (64-Kbyte absolute address branch)

Note: Although INCL(da) and DECL(da) instructions are 16-bit arithmetic instructions, ZF is set if the value of the upper 8 bits of the result is X'00'.

☞ P9 "Note on Use of INCL (da) and DECL (da) "

Table 12-1-2 MN1870 Instruction Set ② (1/5)

Type	Mnemonic	Operation		CZ	in	Machine Language Code (Hex)	IUO
Data transfer	MOV XPl, imm	XPl ← imm				D8 : imm	110
	XPh, imm	XPh ← imm				D9 : imm	110
	YPl, imm	YPl ← imm				DA : imm	110
	YPh, imm	YPh ← imm				DB : imm	110
	XPl, (da)	XPl ← RAM (XPh • DF, da)			T	CD : da	121
	YPl, (da)	YPl ← RAM (YPh • DF, da)			T	CF : da	121
	(da), XPl	RAM (XPh • DF, da) ← XPl				CC : da	110
	(da), YPl	RAM (YPh • DF, da) ← YPl				CE : da	110
	(XP), (YP)	RAM (XP) ← RAM (YP)			T	54	*21
	(XP), imm	RAM (XP) ← imm				55 : imm	110
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (YPh • DF, das)			T	56 : das : dad	*120
	(da), imm	RAM (XPh • DF, da) ← imm				57 : imm : da	120
	(XP), (da)	RAM (XP) ← RAM (XPh • DF, da)			T	36 : da	121
	(da), (YP)	RAM (YPh • DF, da) ← RAM (YP)			T	37 : da	*20
	MOVL (XP), (YP)	RAM (XP) ← RAM (YP) RAM (XP+1) ← RAM (YP+1) XP ← XP+1 YP ← YP+1			T	5C	*41
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (YPh • DF, das) RAM (XPh • DF, dad+1) ← RAM (YPh • DF, das+1)			T	5E : das : dad	141
	XP, (da)	XPl ← RAM (XPh • DF, da) XPh ← RAM (XPh • DF, da+1)			T	38 : da	131
	(da), XP	RAM (XPh • DF, da) ← XPl RAM (XPh • DF, da+1) ← XPh				39 : da	*31
	YP, (da)	YPl ← RAM (YPh • DF, da) YPh ← RAM (YPh • DF, da+1)			T	3A: da	131
	(da), YP	RAM (YPh • DF, da) ← YPl RAM (YPh • DF, da+1) ← YPh				3B : da	*31
	(XP), imm16	RAM (XP) ← imm16_l RAM (XP+1) ← imm16_h XP ← XP+2				5D : imm16_l : imm16_h	120
	(da), imm16	RAM (XPh • DF, da) ← imm16_l RAM (XPh • DF, da+1) ← imm16_h				5F : imm16_l : da : imm16_h	130
	XP, imm16	XP ← imm16				31 : imm16_l : imm16_h	120
	YP, imm16	YP ← imm16				33 : imm16_l : imm16_h	120
	RDTBL (XP), (YP)	RAM (XP) ← RAM (YP)	ROM table look-up			FE	*40
	MOV1 (dad) bpd, (das) bps	RAM (XPh • DF, dad) bpd ← RAM (YPh, das) bps	move bit		LT	F2 : das : bpd4, bps4 : dad	141
	MOV1N (dad) bpd, (das) bps	RAM (XPh • DF, dad) bpd ← $\overline{\text{RAM (YPh, das) bps}}$	bit complement		LT	FA : das : bpd4, bps4 : dad	141

Table 12-1-2 MN1870 Instruction Set ② (2/5)

Type	Mnemonic	Operation	CZ	in	Machine Language Code (Hex)	IUO
Data transfer	XCH XP, YP	XP ↔ YP exchange			D7	*20
	XP, SP	XP ↔ SP			D5	*20
	(XP), (YP)	RAM (XP) ↔ RAM (YP)		LL	58	*41
	(da1), (da2)	RAM (XPh • DF, da1) ↔ RAM (XPh • DF, da2)		LL	5A : da2 : da1	141
	XCH4 (XP)	RAM (XP) un ↔ RAM (XP) ln exchange nibble		L	46	*21
	(da)	RAM (YPh • DF, da) un ↔ RAM (YPh • DF, da) ln		L	47 : da	121
	CLR (XP)	RAM (XP) ← 0 clear	1		4A	*21
	(da)	RAM (XPh • DF, da) ← 0 clear	1		4B : da	121
	(da) bp	RAM (XPh • DF, da) bp← 0 clear bit		L	10 + bp : da (bp = 0~7)	121
	(XP) bp	RAM (XP)bp ← 0 clear bit		L	B0 : bp	121
	DF	DF ← 0 clear DF			85	*1H
	CF	CF ← 0 clear CF	0		87	*1H
	SET (da) bp	RAM (XPh • DF, da) bp ← 1 set bit		L	18 + bp : da (bp = 0~7)	121
	(XP) bp	RAM (XP) bp ← 1 set bit		L	B8 :bp	121
	DF	DF ← 1 set DF			8D	*1H
	CF	CF ← 1 set CF	1		8F	*1H
Arithmetic	ADDC (XP), (YP)	RAM (XP) ← RAM (XP) + RAM (YP) + CF add with CF	CZ	LT	78	*31
	(XP), imm	RAM (XP) ← RAM (XP) + imm + CF	CZ	L	79 : imm	131
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (XPh • DF, dad) + RAM (YPh • DF, das) + CF	CZ	LT	7A : das :dad	131
	(da), imm	RAM (XPh • DF, da) ← RAM (XPh • DF, da) + imm + CF	CZ	L	7B : imm : da	131
	ADDD (XP), (YP)	RAM (XP) ln ← RAM (XP) ln + RAM (YP) ln + CF add decimal	CZ	LT	7C	*41
	(XP), imm4	RAM (XP) ln ← RAM (XP) ln + imm4 + CF	CZ	L	7D : imm4	141
	(dad), (das)	RAM (XPh • DF, dad) ln ← RAM (XPh • DF, dad) ln + RAM (YPh • DF, das) + CF	CZ	LT	7E : das :dad	141
	(da), imm4	RAM (XPh • DF, da) ln ← RAM (XPh • DF, da) ln + imm4 + CF	CZ	L	7F : imm4 : da	141
	ADDR XPl, XPl, imm	XPl ← XPl + imm	CZ		ED : imm	121
	YPl, YPl, imm	YPl ← YPl + imm	CZ		EF : imm	121
	XPl, (da), imm	XPl ← RAM (XPh • DF, da) + imm	CZ	T	E9 : imm :da	131
	YPl, (da), imm	YPl ← RAM (XPh • DF, da) + imm	CZ	T	EB : imm : da	131
	INC XP	XP ← XP + 1 increment			E5	*1H
	YP	YP ← YP + 1			E7	*1H
	(XP)	RAM (XP) ← RAM (XP) + 1	CZ	L	48	*21
	(da)	RAM (XPh • DF, da) ← RAM (XP • DF, da) + 1	CZ	L	49 : da	121
	DEC XP	XP ← XP – 1 decrement			E4	*1H
	YP	YP ← YP – 1			E6	*1H
	(XP)	RAM (XP) ← RAM (XP) – 1	CZ	L	40	*21
	(da)	RAM (XPh • DF, da) ← RAM (XPh • DF, da) – 1	CZ	L	41 : da	121

Table 12-1-2 MN1870 Instruction Set ② (2/5)

Type	Mnemonic	Operation	CZ	in	Machine Language Code (Hex)	IUO
Arithmetic	INCL (da)	RAM (XPh • DF, da) ← RAM (XPh • DF, da) + 1	CZ	L	CB : da	151
		RAM (XPh • DF, da + 1) ← RAM (XPh • DF, da + 1) + CF	(Note)			
	DECL (da)	RAM (XPh • DF, da) ← RAM (XPh • DF, da) – 1	CZ	L	C3 : da	151
		RAM (XPh • DF, da + 1) ← RAM (XPh • DF, da + 1) – CF	(Note)			
	SUBC (XP), (YP)	RAM (XP) ← RAM (XP) – RAM (YP) – CF subtract	CZ	LT	70	*31
	(XP), imm	RAM (XP) ← RAM (XP) – imm – CF	CZ	L	71 : imm	131
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (XPh • DF, dad) – RAM (YPh • DF, das) – CF	CZ	LT	72 : das : dad	131
	(da), imm	RAM (XPh • DF, da) ← RAM (XPh • DF, da) – imm – CF	CZ	L	73 : imm : da	131
	SUBD (XP), (YP)	RAM (XP) In ← RAM (XP) In – RAM (YP) In – CF • sub decimal	CZ	LT	74	*41
	(XP), imm4	RAM (XP) In ← RAM (XP) In – imm4 – CF	CZ	L	75 : imm4	141
	(dad), (das)	RAM (XPh • DF, dad) In ← RAM (XPh • DF, dad) In – RAM (YPh • DF, das) In – CF	CZ	LT	76 : das : dad	141
	(da), imm4	RAM (XPh • DF, da) In ← RAM (XPh • DF, da) In – imm4 – CF	CZ	L	77 : imm4 : da	141
	MUL (XP)	RAM (XP + 1), RAM (XP) – RAM (XP)*RAM (XP + 1) XP ← XP + 1		TT	59	*81
	(XP), imm	RAM (XP + 1), RAM (XP) – RAM (XP)*imm XP ← XP + 1		T	C9 : imm	181
	DIV (XP)	RAM (XP) ← RAM (XP + 1), RAM (XP) / RAM (YP) XP ← XP + 1		TT	51	*101
		RAM (XP + 1) ← remainder				
	(XP), imm	RAM (XP) ← RAM (XP + 1), RAM (XP) / imm XP ← XP + 1		T	C1 : imm	1101
		RAM (XP + 1) ← remainder				
	CMP XPl, imm	XPl – imm compare	CZ		D1 : imm	121
	YPl, imm	YPl – imm	CZ		D3 : imm	121
	(XP), (YP)	RAM (XP) - RAM (YP)	CZ	LT	60	*31
	(XP), imm	RAM (XP) - imm	CZ	L	61 : imm	131
	(da1), (da2)	RAM (XPh • DF, da1) - RAM (YPh • DF, da2)	CZ	LT	62 : da2 : da1	131
	(da), imm	RAM (XPh • DF, da) - imm	CZ	L	63 : imm : da	131
	CMPL (XP), (YP)	RAM (XP) - RAM (YP)	CZ	TT	84	*51
		RAM (XP + 1) - RAM (YP + 1) – CF				
		XP ← XP + 2				
		YP ← YP + 2				
	CMPM (XP) mask, imm	[RAM (XP) . and. mask] – imm compare with mask	CZ	T	44 : mask : imm	131
	(da) mask, imm	[RAM (XPh • DF, da) . and. mask] – imm	CZ	T	45 : da : mask : imm	141
	(XP) mask, (YP) mask	[RAM (XP) . and. mask] – [RAM (YP). and. mask]	CZ	TT	50 : mask	141
	(da1) mask,(da2) mask	[RAM (XPh • DF, da1) . and. mask] – [RAM (YPh • DF, da2)). and. mask]	CZ	TT	52 : da2 : mask : da1	151
Logic	NOT (XP)	RAM (XP) ← $\overline{\text{RAM (XP)}}$ complement	Z	L	42	*21
	(da)	RAM (XPh • DF, da) ← $\overline{\text{RAM (XPh • DF, da)}}$	Z	L	43 : da	121
	ROL (XP)	CF ← RAM ( ) 7 : RAM ( ) bp ← RAM ( ) bp–1 : RAM ( ) 0 ← CF	C	L	4C	*21
	(da)	[RAM ( ) = RAM (XP) or RAM (XPh • DF, da)] rotate left with CF	C	L	4D : da	121

Table 12-1-2 MN1870 Instruction Set ② (4/5)

Type	Mnemonic	Operation	CZ	in	Machine Language Code (Hex)	IUO
Logic	ROR (XP)	CF ← RAM ( ) 0 : RAM ( ) bp–1 ← RAM ( ) bp : RAM ( ) 7 ← CF	C	L	4E	*21
	(da)	[RAM ( ) = RAM (XP) or RAM (XPh • DF, da)] rotate right with CF	C	L	4F : da	121
	AND (XP), (YP)	RAM (XP) ← RAM (XP) .and. RAM (YP)	Z	LT	64	*31
	(XP), imm	RAM (XP) ← RAM (XP) .and. imm	Z	L	65 : imm	131
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (XPh • DF, dad) .and. RAM (YPh • DF, das)	Z	LT	66 : das : dad	131
	(da), imm	RAM (XPh • DF, da) ← RAM (XPh • DF, da) .and. imm	Z	L	67 : imm : da	131
	OR (XP), (YP)	RAM (XP) ← RAM (XP) .or. RAM (YP)	Z	LT	6C	*31
	(XP), imm	RAM (XP) ← RAM (XP) .or. imm	Z	L	6D : imm	131
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (XPh • DF, dad) .or. RAM (YPh • DF, das)	Z	LT	6E : das : dad	131
	(da), imm	RAM (XPh • DF, da) ← RAM (XPh • DF, da) .or. imm	Z	L	6F : imm : da	131
	XOR (XP), (YP)	RAM (XP) ← RAM (XP) .xor. RAM (YP) exclusive OR	Z	LT	68	*31
	(XP), imm	RAM (XP) ← RAM (XP) .xor. imm	Z	L	69 : imm	131
	(dad), (das)	RAM (XPh • DF, dad) ← RAM (XPh • DF, dad) .xor. RAM (YPh • DF, das)	Z	LT	6A : das : dad	131
	(da), imm	RAM (XPh • DF, da) ← RAM (XPh • DF, da) .xor. imm	Z	L	6B : imm : da	131
Subroutine call	CALL label	call absolute (4Kbyte) IP ← IPh, label RAM (SP–1) ← IPh, IPm RAM (SP–2) ← IPl SP ← SP–2			90 + label – m : label – l	*30
	/label	call absolute (64Kbyte) IP ← / label RAM (SP–1) ← IPh, IPm RAM (SP–2) ← IPl SP ← SP–2			F7 : label – h, m : label – l	*30
	XP	IP ← XP RAM (SP–1) ← IPh, IPm RAM (SP–2) ← IPl SP ← SP–2			FD	*30
	RET	IPl ← RAM (SP) : IPh, IPm ← RAM (SP+1) : SP ← SP+2		L	F4	*40
	RETI	FS ← RAM (SP) IPl ← RAM (SP + 1) IPh, IPm ← RAM (SP + 2) SP ← SP + 3	CZ	L	F5	*50
Unconditional branch	BR $label	IP ← IP + 2 + $label branch relative (+127––128)			88 : $label	120
	label	IP ← IPh, label branch absolute (4Kbyte/page)			A0 + label – m : label – l	120
	/ label	IP ← / label branch absolute long (64Kbyte)			F6 : label – h, m : label – l	120
	XP	IP ← XP			FC	*20
Unconditional skip	SKIP	IP ← IP + 2 skip next one byte			80	120
Conditional branch	BC $label	if CF = 1 then IP ← IP + 2 + $label / else IP ← IP + 2			81 : $label	120
	BZ $label	if ZF = 1 then IP ← IP + 2 + $label / else IP ← IP + 2			82 : $label	120

Table 12-1-2 MN1870 Instruction Set ② (5/5)

Type	Mnemonic	Operation	CZ	in	Machine Language Code (Hex)	IUO
Condition-al branch	BLE $label	if CF = 1. or. ZF = 1 then IP ← IP + 2 + $label / else IP ← IP + 2			83 : $label	120
	BNC $label	if CF = 0 then IP ← IP + 2 + $label / else IP ← IP + 2			89 : $label	120
	BNZ $label	if ZF = 0 then IP ← IP + 2 + $label / else IP ← IP + 2			8A : $label	120
	BGT $label	if CF = 0. and. ZF = 0 then IP ← IP + 2 + $label / else IP ← IP + 2			8B : $label	120
Test and condition-al branch	T1BNZ (da) bp, $label	if RAM (XPh • DF, da) bp = 1 then IP ← IP + 3 + $label / else IP ← IP + 3		T	20 + bp : da : $label	130
	(XP) bp, $label	if RAM (XP) bp = 1 then IP ← IP + 3 + $label / else IP ← IP + 3		T	B2 : bp : $label	130
	T1BZ (da) bp, $label	if RAM (XPh • DF, da) bp = 0 then IP ← IP + 3 + $label / else IP ← IP + 3		T	28 + bp : da : $label	130
	(XP) bp, $label	if RAM (XP) bp = 0 then IP ← IP + 3 + $label / else IP ← IP + 3		T	BA : bp : $label	130
Compare and condi-tional branch	CMPBNE (XP), imm, $label	if RAM (XP) ≠ imm then IP ← IP + 3 + $label / else IP ← IP + 3		T	F0 : imm : $label	130
	(da), imm, $label	if RAM (XP • DF, da) ≠ imm then IP ← IP + 4 + $label / else IP ← IP + 4		T	F1 : imm : da : $label	140
	CMPBE (XP), imm, $label	if RAM (XP) imm then IP ← IP + 3 + $label / else IP ← IP + 3		T	F8 : imm : $label	130
	(da), imm, $label	if RAM (XPh • DF, da) = imm then IP ← IP + 4 + $label / else IP ← IP + 4		T	F9 : imm : da : $label	140
Loop	LOOP $label	RAM (SP) ← RAM (SP) ≠ 1 if RAM (SP) ≠ 0 then IP ← IP + 2 + $label / else IP ← IP + 2 : SP ← SP + 1		L	E0 : $label	*30
	RLOOP XP, $label	RAM (SP) ← RAM (SP) – 1 : XP ← XP + 1 if RAM (SP) ≠ 0 then IP ← IP + 2 + $label / else IP ← IP + 2 : SP ← SP + 1		L	E1 : $label	*30
	YP, $label	RAM (SP) ← RAM (SP) – 1 : YP ← YP + 1 if RAM (SP) ≠ 0 then IP ← IP + 2 + $label / else IP ← IP + 2 : SP ← SP + 1		L	E3 : $label	*30
Stack manipu-lation	PUSH XP	RAM (SP–1) ← XPh : RAM (SP–2) ← XPl : SP ← SP – 2 push stack			BC	*31
	YP	RAM (SP–1) ← YPh : RAM (SP–2) ← YPl : SP ← SP – 2			BE	*31
	(XP)	RAM (SP–1) ← RAM (XP) : SP ← SP – 1		T	BF	*31
	(da)	RAM (SP–1) ← RAM (XPh • DF, da) : SP ← SP – 1		T	BD : da	*31
	imm	RAM (SP–1) ← imm : SP ← SP – 1			FB : imm	121
	FS	RAM (SP–1) ← FS : SP ← SP – 1			BB	*21
	POP XP	XPl ← RAM (SP) : XPh ← RAM (SP + 1) : SP ← SP + 2 pop stack		L	B4	*31
	YP	YPl ← RAM (SP) : YPh ← RAM (SP + 1) : SP ← SP + 2		L	B6	*31
	(XP)	RAM (XP) ← RAM (SP) : SP ← SP + 1		L	B7	*21
	(da)	RAM (XPh • DF, da) ← RAM (SP) : SP ← SP + 1		L	B5 : da	*20
	FS	FS ← RAM (SP) : SP ← SP + 1	CZ	L	B3	*21
Other	NOP	IP ← IP + 1 no operation			00	*1H
	HALT	set CPU HALT			19 : 16 : 21 : 16 : FD	
	STOP	set CPU STOP			18 : 16 : 20 : 16 : FD	

12-2 MN1870 Series Opcode Map

Table 12-2-1 MN1870 Series Opcode Map

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP															
1	CLR (da) bp								SET (da) bp							
2	T1BNZ (da) bp, $label								T1BZ (da)bp, $label							
3		MOVL XP, imm16		MOVL YP, imm16			MOV (XP), (da) | (da), (YP)		MOVL XP, (da) | (da), XP | YP, (da) | (da), YP							
4	DEC (XP) | (da) C, Z		NOT (XP) | (da) Z		CMPM (XP)mask, imm | (da)mask, imm C, Z		XCH4 (XP) | (da)	INC (XP) | (da) C, Z		CLR (XP) | (da) ZF="1"		ROL (XP) | (da) C		ROR (XP) | (da) C	
5	CMPM (XP)mask,(YP)mask C, Z	DIV (XP)	CMPM (da1)mask,(da2)mask C, Z		MOV (XP), (YP) | (XP), imm | (dad), (das) | (da), imm				XCH (XP), (YP)	MUL (XP)	XCH (da1), (da2)		MOVL (XP), (YP) | (XP), imm16 | (dad), (das) | (da), imm16			
6	CMP (XP), (YP) | (XP), imm | (da1), (da2) | (da), imm C, Z				AND (XP), (YP) | (XP), imm | (da1), (da2) | (da), imm Z				XOR (XP), (YP) | (XP), imm | (da1), (da2) | (da), imm Z				OR (XP), (YP) | (XP), imm | (da1), (da2) | (da), imm Z			
7	SUBC (XP), (YP) | (XP), imm | (da1), (da2) | (da), imm C, Z				SUBD (XP), (YP) | (XP), imm4 | (da1), (da2) | (da), imm4 C, Z				ADDC (XP), (YP) | (XP), imm | (da1), (da2) | (da), imm C, Z				ADDD (XP), (YP) | (XP), imm4 | (da1), (da2) | (da), imm4 C, Z			
8	SKIP	BC $label	BZ $label	BLE $label	CMPL (XP), (YP) C, Z	CLR DF DF="0"		CLR CF CF="0"	BR $label	BNC $label	BNZ $label	BGT $label		SET DF DF="1"		SET CF CF="1"
9	CALL label															
A	BR label															
B	CLR (XP) bp	POP FS C, Z	T1BNZ (XP) bp, $label	FS CZ	XP	POP (da)	YP	(XP)	SET (XP)dp	PUSH FS	T1BZ (XP) bp, $label	FS	XP	PUSH (da)	YP	(XP)
C		DIV (XP), imm		DECL (da) (note) C, Z						MUL (XP), imm		INCL (da) (note) C, Z	MOV (da), XPl | XPl, (da) | (da), YPl | YPl, (da)			
D	CMP XP1, imm | YP1, imm C, Z					XCH XP, SP		XCH XP, YP	MOV XPl, imm | XPh, imm | YPl, imm | YPh, imm							
E	LOOP $label	RLOOP XP, $label	LOOP $label	RLOOP YP, $label	DEC XP	INC XP	DEC YP	INC YP	ADDR XPl, (da), imm | YPl, (da), imm | XPl, XPl, imm | YPl, YPl, imm C, Z							
F	CMPBNE( ), imm. $label (XP) | (da)		MOV1 (dad)bpd, (das)bps		RET	RETI CS	BR /label	CALL /label	CMPBE( ), imm. $label (XP) | (da)		MOV1N (dad)bpd,(das)bps	PUSH imm	BR XP	CALL XP	RDTBL (XP), (YP)	

12-3 List of Special-Function Registers

Address	Register Name	Bit Abbreviations								Reference Page
		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
X'000'	Port 0 (PORT0)	P07	P06	P05	P04	P03	P02	P01	P00	80
		①	CMOS 3-state input/output port							
X'001'	Port 1 (PORT1)	P17	P16	P15	P14	P13	P12	P11	P10	80
		CMOS 3-state input/output port								
X'002'	Port 2 (PORT2)	P27	P26	P25	P24	P23	P22	P21	P20	80
		CMOS 3-state input/output port								
X'003'	Port 3 (PORT3)				P34	P33	P32	P31	P30	80
					CMOS 3-state input/output port					
X'004'	Port 4 (PORT4)	P47	P46	P45	P44	P43	P42	P41	P40	80
		CMOS 3-state input/output port								
X'005'	Port 5 (PORT5)	P57	P56	P55	P54	P53	P52	P51	P50	80
		High-voltage output port (P-channel open-drain)								
X'006'	Port 6 (PORT6)	P67	P66	P65	P64	P63	P62	P61	P60	80
		High-voltage output port (P-channel open-drain)								
X'007'	Port 7 (PORT7)	P77	P76	P75	P74	P73	P72	P71	P70	80
		High-voltage output port (P-channel open-drain)								
X'008'	Port 8 (PORT8)	P87	P86	P85	P84	P83	P82	P81	P80	80
		High-voltage output port (P-channel open-drain)								
X'009'	Port 9 (PORT9)					P93	P92	P91	P90	80
						CMOS 3-state input/output port				
X'00A'	Port/FLP select register 0 (PFSR0)	P67SEL	P66SEL	P65SEL	P64SEL	P63SEL	P62SEL	P61SEL	P60SEL	164
		Port 6/FLP pin function selection								
X'00B'	Port/FLP select register 1 (PFSR1)	P87SEL	P86SEL	P85SEL	P84SEL	P83SEL	P82SEL	P81SEL	P80SEL	165
		Port 8/FLP pin function selection								
X'00C'	Port/FLP select register 2 (PFSR2)							P7SEL	P5SEL	166
								③	②	
X'00D'	Digit number/dimmer register (DNDR)	DN3	DN2	DN1	DN0	DR3	DR2	DR1	DR0	167
		Digit number setting data				Dimmer setting data				
X'00E'	FLP clock mode register (FCMR)						DSPAUTO	DSPCLK	DSPSTART	168
							⑥	⑤	④	
X'00F'	Not used									—
X'010'	Interrupt mode control register (IRQM)				IEMASK	EDGEDIR3	EDGEDIR2	EDGEDIR1	EDGEDIR0	58
					⑪	⑩	⑨	⑧	⑦	

Notes:
① N-channel open-drain input/output port
② Port 5/FLP pin function selection
③ Port 7/FLP pin function selection
④ Display stop/display start
⑤ Display-time-per-digit setting
⑥ Serial display/automatic repeat display selection
⑦ IRQ1 valid edge specification
⑧ IRQ0 valid edge specification
⑨ IRQ2valid edge specification
⑩ IRQ3 valid edge specification
⑪ Interrupt enable mask flag

Address	Register Name	Bit Abbreviations								Reference Page
		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
X'011'	Not used									—
X'012'	Interrupt request flag register 0 (IF0)	IFTC3	IFTC2	IFTC1	IFTC0	IFIRQ3	IFIRQ2	IFIRQ1	IFIRQ0	59
		Interrupt request flags								
X'013'	Interrupt request flag register 1 (IF1)	IFRSV	IFADT	IFKYS	IFSIF2	IFSIF1	IFSIF0	IFTC5	IFTC4	59
		Interrupt request flags								
X'014'	Interrupt enable flag register 0 (IE0)	IETC3	IETC2	IETC1	IETC0	IEIRQ3	IEIRQ2	IEIRQ1	IEIRQ0	61
		Interrupt enable flags								
X'015'	Interrupt enable flag register 1 (IE1)	IERSV	IEADT	IEKYS	IESIF2	IESIF1	IESIF0	IETC5	IETC4	63
		Interrupt enable flags								
X'016'	CPU mode register (CPUM)		CLKSEL			OSCM1	OSCM0	HALTM	STOPM	66
			System clock selection			Oscillation control		HALT mode specification	STOP mode specification	
X'017'	Not used									—
X'018'	Port 0 direction control register (P0DIR)		P06ON	P05ON	P04ON	P03ON	P02ON	P01ON	P00ON	80
			Port 0 input/output direction control							
X'019'	Port 1 direction control register (P1DIR)	P17ON	P16ON	P15ON	P14ON	P13ON	P12ON	P11ON	P10ON	80
		Port 1 input/output direction control								
X'01A'	Port 2 direction control register (P2DIR)	P27ON	P26ON	P25ON	P24ON	P23ON	P22ON	P21ON	P20ON	80
		Port 2 input/output direction control								
X'01B'	Port 3 direction control register (P3DIR)				P34ON	P33ON	P32ON	P31ON	P30ON	80
					Port 3 input/output direction control					
X'01C'	Port 4 direction control register (P4DIR)	P47ON	P46ON	P45ON	P44ON	P43ON	P42ON	P41ON	P40ON	80
		Port 4 input/output direction control								
X'01D'	Port 9 direction control register (P9DIR)					P93ON	P92ON	P91ON	P90ON	80
						Port 9 input/output direction control				
X'01E' X'01F'	Not used									—
X'020'	Transmit/receive shift buffer 0 (SIBUF0)	SIBUF07	SIBUF06	SIBUF05	SIBUF04	SIBUF03	SIBUF02	SIBUF01	SIBUF00	143
		Serial interface 0 transmit/receive shift buffer								
X'021'	Transmit/receive shift buffer 1 (SIBUF1)	SIBUF17	SIBUF16	SIBUF15	SIBUF14	SIBUF13	SIBUF12	SIBUF11	SIBUF10	143
		Serial interface 1 transmit/receive shift buffer								
X'022'	Serial interface mode register 0 (SIM0)	SIFOE0	SIFCON0	SBTC0	SBOC0	BITDIR0	CM02	CM01	CM00	147
		⑫	Parallel connection specification	Output form specification	Output form specification	Start bit specification	Transfer clock specification			
X'023'	Serial interface mode register 1 (SIM1)	SIFOE1		SBTC1	SBOC1	BITDIR1	CM12	CM11	CM10	148
		⑬		Output form specification	Output form specification	Start bit specification	Transfer clock specification			
X'024'	Serial interface bit counter 0 (SBC0)	LTI0	CLKPL0	STCE0		BUSY0	SBC02	SBC01	SBC00	144
		Input latch/through	Clock polarity specification	Start condition		Busy flag	Serial interface 0 bit counter			
X'025'	Serial interface bit counter 1 (SBC1)			STCE1		BUSY1	SBC12	SBC11	SBC10	145
				Start condition		Busy flag	Serial interface 1 bit counter			

Notes: ⑫ Serial interface 0 input/output enable
⑬ Serial interface 1 input/output enable

Address	Register Name	Bit Abbreviations								Reference Page
		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
X'026'	Not used									—
X'027'	Remote control/noise filter control register (RMC)		RMCEN	RMSYN	RMOEN	RMC3	RMC2	RMC1	RMC0	202, 213
			Carrier use specification	Synchronous mode	P15/RMOUT selection	⑮		⑭		
X'028'	Programmable timer counter 0 (timer latch 0/ binary counter 0) (TC0/BC0)	TC07/ BC07	TC06/ BC06	TC05/ BC05	TC04/ BC04	TC03/ BC03	TC02/ BC02	TC01/ BC01	TC00/ BC00	116
		Read: Binary counter value Write: Reload value setting								
X'029'	Timer mode register 0 (TM0)	TC0EN	TC0MOD	TC0OUT	TC0SYN	TC0PS1	TC0PS0	TC0CLK1	TC0CLK0	119
		Timer 0 enable	Pulse width measure-ment mode	Timer output specification	Synchro-nous mode	Timer 0 prescaler scaling factor selection		Timer 0 clock source selection		
X'02A'	Programmable timer counter 1 (timer latch 1/ binary counter 1) (TC1/BC1)	TC17/ BC17	TC16/ BC16	TC15/ BC15	TC14/ BC14	TC13/ BC13	TC12/ BC12	TC11/ BC11	TC10/ BC10	116
		Read: Binary counter value Write: Reload value setting								
X'02B'	Timer mode register 1 (TM1)	TC1EN	TC3EN	TC3CLK1	TC3CLK0	TC1OUT	TC1CLK	TC1PS1	TC1PS0	120
		Timer 1 enable	Timer 3 enable	Timer 3 clock source selection		⑰	⑯	Timer 1 prescaler scaling factor selection		
X'02C'	Timer mode register 2 (TM2)	WDEN	BUZFQ	BUZEN	TC2EN	TC2FLG	TC2CLK	TC2MPX1	TC2MPX0	121
		Watch-dog timer enable	Buzzer frequency selection	Buzzer enable	Timer 2 enable	Realtime clock flag	Timer 2 clock source selection	Timer 2 interrupt cycle selection		
X'02D'	Programmable timer counter 3 (timer latch 3/ binary counter 3) (TC3/BC3)	TC37/ BC37	TC36/ BC36	TC35/ BC35	TC34/ BC34	TC33/ BC33	TC32/ BC32	TC31/ BC31	TC30/ BC30	117
		Read: Binary counter value Write: Reload value setting								
X'02E'	A/D control register (ADC)				ADST	ADLS	ADCHS2	ADCHS1	ADCHS0	194
					A/D start	Ladder resistor on/off control	Input channel selection			
X'02F'	A/D buffer (ADBUF)	ADBUF7	ADBUF6	ADBUF5	ADBUF4	ADBUF3	ADBUF2	ADBUF1	ADBUF0	195
		A/D conversion result storage								
X'030'	PWM control register (PWMC)			PWM2OE	PWM1OE	PWM1E	PWM0E	PWM0LD	PWM0OE	186
				P17/PWM2 selection	P16/PWM1 selection	PWM1 enable	PWM0 enable	Latch data transfer	P34/PWM0 selection	

Notes: ⑭ IRQ1 pin noise filter sampling clock specification
⑮ IRQ2 pin noise filter sampling clock specification
⑯ Timer 1 clock source selection
⑰ Timer 1 output specification

Address	Register Name	Bit Abbreviations								Reference Page
		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
X'031'	PWM0 data register upper 7 bits (PWMHR)		PWMHR6	PWMHR5	PWMHR4	PWMHR3	PWMHR2	PWMHR1	PWMHR0	187
			14-bit PWM0 data register upper 7 bits							
X'032'	PWM0 data register lower 7 bits (PWMLR)		PWMLR6	PWMLR5	PWMLR4	PWMLR3	PWMLR2	PWMLR1	PWMLR0	187
			14-bit PWM0 data register lower 7 bits							
X'033'	PWM0 data latch upper 7 bits (PWMHL)		PWMHL6	PWMHL5	PWMHL4	PWMHL3	PWMHL2	PWMHL1	PWMHL0	187
			14-bit PWM0 data latch upper 7 bits							
X'034'	PWM0 data latch lower 7 bits (PWMLL)		PWMLL6	PWMLL5	PWMLL4	PWMLL3	PWMLL2	PWMLL1	PWMLL0	187
			14-bit PWM0 data latch lower 7 bits							
X'035'	Transmit/receive shift buffer 2 (SIBUF2)	SIBUF27	SIBUF26	SIBUF25	SIBUF24	SIBUF23	SIBUF22	SIBUF21	SIBUF20	143
		Serial interface 2 transmit/receive shift buffer								
X'036'	Serial interface mode register 2 (SIM2)	SIFOE2		SBTC2	SBOC2	BITDIR2	CM22	CM21	CM20	149
		⑱		Output form specification	Output form specification	Start bit specification	Transfer clock specification			
X'037'	Serial interface bit counter 2 (SBC2)	LTI2	CLKPL2	STCE2		BUSY2	SBC22	SBC21	SBC20	146
		Input latch/ through	Clock polarity specification	Start condition		Busy flag	Serial interface 0 bit counter			
X'038'	Automatic data transfer control register (ADTC)			DAPINC	SAPINC	BURST	ADTC2	ADTC1	ADTC0	227
				DAP incre-ment	SAP incre-ment	Transfer mode	Transfer factor specification			
X'039'	Transfer byte counter (ADTB)	ADTB7	ADTB6	ADTB5	ADTB4	ADTB3	ADTB2	ADTB1	ADTB0	228
		Transfer byte number setting								
X'03A'	Data transfer target address register lower 8 bits (TAPL)	TAPL7	TAPL6	TAPL5	TAPL4	TAPL3	TAPL2	TAPL1	TAPL0	229
		Data transfer target address lower 8 bits								
X'03B'	Data transfer target address register upper 8 bits (TAPH)	TAPH7	TAPH6	TAPH5	TAPH4	TAPH3	TAPH2	TAPH1	TAPH0	229
		Data transfer target address upper 8 bits								
X'03C'	Data transfer source address register lower 8 bits (SAPL)	SAPL7	SAPL6	SAPL5	SAPL4	SAPL3	SAPL2	SAPL1	SAPL0	229
		Data transfer source address lower 8 bits								
X'03D'	Data transfer source address register upper 8 bits (SAPH)	SAPH7	SAPH6	SAPH5	SAPH4	SAPH3	SAPH2	SAPH1	SAPH0	229
		Data transfer source address upper 8 bits								
X'03E'	Data transfer destination address register lower 8 bits (DAPL)	DAPL7	DAPL6	DAPL5	DAPL4	DAPL3	DAPL2	DAPL1	DAPL0	229
		Data transfer destination address lower 8 bits								
X'03F'	Data transfer destination address register upper 8 bits (DAPH)	DAPH7	DAPH6	DAPH5	DAPH4	DAPH3	DAPH2	DAPH1	DAPH0	230
		Data transfer destination address upper 8 bits								

Notes: ⑱ Serial interface 2 input/output enable

Address	Register Name	Bit Abbreviations								Reference Page
		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
X'040'	Port 0 pull-up control register (P0RON)		P06RON		P04RON	P03RON		P01RON	P00RON	80
		Port 0 pull-up resistor on/off control								
X'041'	Port 1 pull-up control register (P1RON)	P17RON	P16RON	P15RON	P14RON	P13RON	P12RON	P11RON	P10RON	80
		Port 1 pull-up resistor on/off control								
X'042'	Port 2 pull-up control register (P2RON)							P2HRON	P2LRON	80
								Port 2 pull-up resistor on/off control (nibble-wise)		
X'043'	Port 3 pull-up control register (P3RON)				P34RON	P33RON	P32RON	P31RON	P30RON	80
					Port 3 pull-up resistor on/off control					
X'044'	Port 4 pull-down control register (P4RON)	P47RON	P46RON	P45RON	P44RON	P43RON	P42RON	P41RON	P40RON	80
		Port 4 pull-down resistor on/off control								
X'045'	Port 9 pull-up control register (P9RON)						P92RON	P91RON		80
							Port 9 pull-up resistor on/off control			
X'046' X'047'	Not used									—
X'048'	Programmable timer counter 4 (timer latch 4/binary counter 4) (TC4/BC4)	TC47/BC47	TC46/BC46	TC45/BC45	TC44/BC44	TC43/BC43	TC42/BC42	TC41/BC41	TC40/BC40	117
		Read: Binary counter value Write: Reload value setting								
X'049'	Timer mode register 4 (TM4)	TC4EN	TC4SYN	TC4CLK1	TC4CLK0	TC4OUT				122
		Timer 4 enable	Synchronous mode	Timer 4 clock source selection		Timer 4 output specification				
X'04A'	Programmable timer counter 5 (timer latch 5/binary counter 5) (TC5/BC5)	TC57/BC57	TC56/BC56	TC55/BC55	TC54/BC54	TC53/BC53	TC52/BC52	TC51/BC51	TC50/BC50	118
		Read: Binary counter value Write: Reload value setting								
X'04B'	Timer mode register 5 (TM5)	TC5EN	TC35OUT	TCO35C		TC5PS1	TC5PS0	TC5CLK1	TC5CLK0	123
		Timer 5 enable	Synchronous output specification	Synchronous output clock selection		Timer 5 prescaler scaling factor selection		Timer 5 clock source selection		
X'04C' X'04D'	Not used									—
X'04E'	PWM2 data register (PWMR2)	PWMR27	PWMR26	PWMR25	PWMR24	PWMR23	PWMR22	PWMR21	PWMR20	188
		PWM2 data register								
X'04F'	Not used									—

12-4 MN18P76423 (One-Time Programmable Microcomputer with On-Chip EPROM)

12-4-1 Overview

The MN18P76423 is a one-time programmable microcomputer with 64K × 8-bit (64-Kbyte) EPROM in place of the mask ROM of the MN1872423/MN1873223/MN1874023/MN1874823.

The MN18P76423-FBis a resin-sealed model, permitting data to be written to EPROM once and EPROM data can not be erased.

The PX-AP1876423-FBC is sealed in a windowed ceramic package, allowing data to be erased by exposure of ultraviolet light.

Both plastic and ceramic packages are of 84-pin flat type.

12-4-2 On-Chip EPROM

The MN18P76423 has on-chip 64-Kbyte electrically programmable EPROM. The Intel 27C256 specification (Intel high-speed (II)), with a +12.5V programming voltage (V_{PP}), is used for programming and verification of the on-chip EPROM.

The MN18P76423 can also be programmed with a general-purpose PROM programmer by converting the MN18P76423's 84 pins to the normal 28-pin EPROM arrangement.

Placing the MN18P76423 in on-chip EPROM mode suspends its microcomputer functions, allowing the on-chip EPROM to be programmed. The pin configuration is EPROM mode is shown in figure 12-4-1.

When the Intel 27C256 specification is used for MN18P76423 programming and verification, two write operations are needed, one for addresses X'0000'~X'7FFF' (lower 32 bits) and one for addresses X'8000" ~X'FFFF' (upper 32 bits). (With a PROM programmer for which addresses X'8000'~X'FFFF' cannot be designated in 27C265 mode, designate addresses X'0000'~X'7FFF'.)

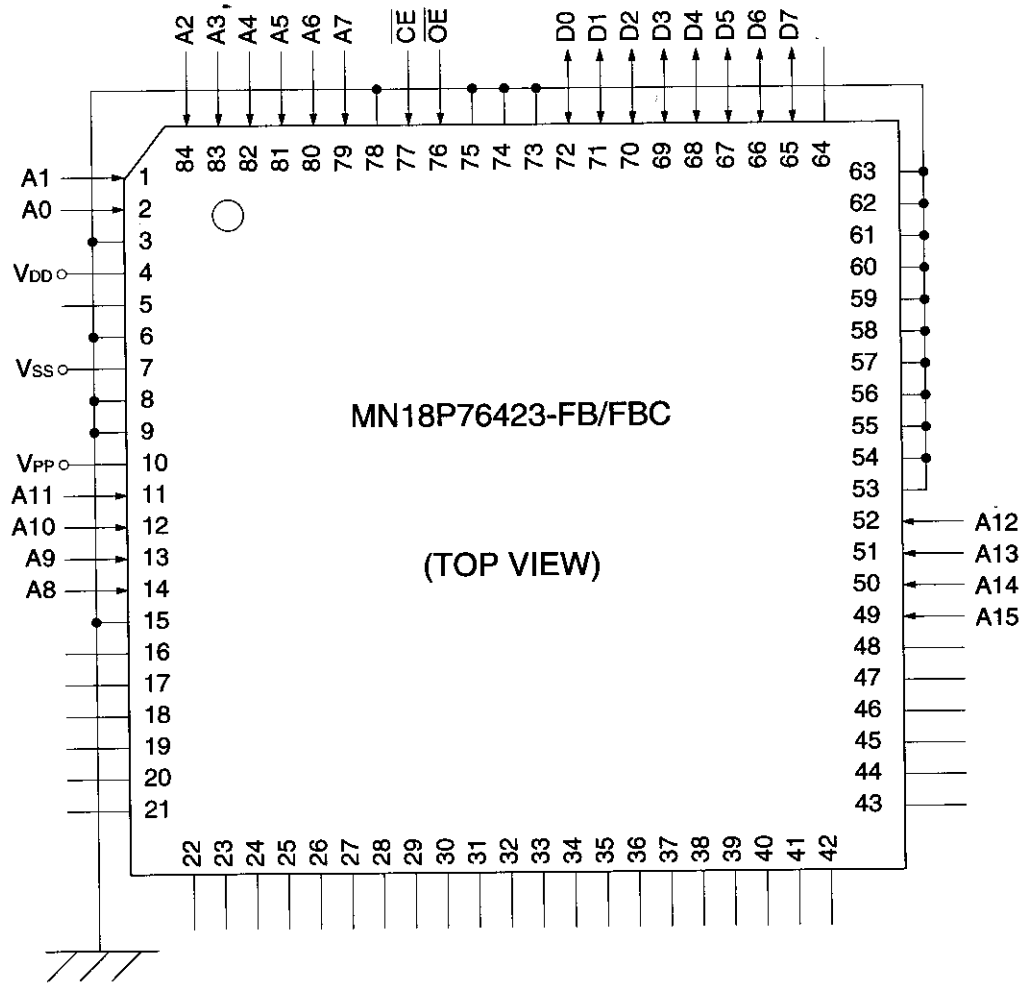


Figure 12-4-1 Pin Configuration in EPROM Mode

(1) EPROM Mode Setting

■ 27C256 mode

EPROM mode setting includes selection of the mode for programming/verifying with the $\overline{OE}$ pin and $\overline{CE}$ pin after the prescribed voltages are applied to the supply voltage pin (V_{DD}) and program voltage pin (V_{PP}). 27C256 mode EPROM settings are listed in table 12-4-1.

Table 12-4-1 EPROM Mode Settings (27C256 Mode)

Mode \ Pins	V_{DD}	Vpp	$\overline{CE}$	$\overline{OE}$	D0~D7	A0~A14
Program	+6V	+12.5V	"L"	"H"	Data input	Address input
Verify	+6V	+12.5V	"H"	"L"	Data output	Address input
Program/ verify disabled	+6V	+12.5V	"H"	"H"	Hi-Z	—
Read	+5V	+5V	"L"	"L"	Data output	Address input
Output disabled	+5V	+5V	"L"	"H"	Hi-Z	—
Standby	+5V	+5V	"H"	—	Hi-Z	—

(2) Programming On-Chip EPROM

■ 27C256 mode

① Programming

Program mode is entered by setting the supply voltage to +6V, then applying +12.5V to the V_{PP} pin, and forcing the $\overline{OE}$ pin high and the $\overline{CE}$ pin low. The 8-bit data input from the data input/output pins (D0~D7) is written to the on-chip EPROM address specified by the address input pins (A0~A14).

② Verification

Forcing the $\overline{CE}$ pin high and the $\overline{OE}$ pin low after programming causes the written data to be output from the data input/output pins (D0~D7), enabling the data to be verified.

☞ For program/verify input/output timing, see "12-4-2 (3) On-Chip EPROM Programming Electrical Characteristics."

Following the flowchart in figure 12-4-2 for on-chip EPROM programming enables programming and verification to be executed fast and reliably.

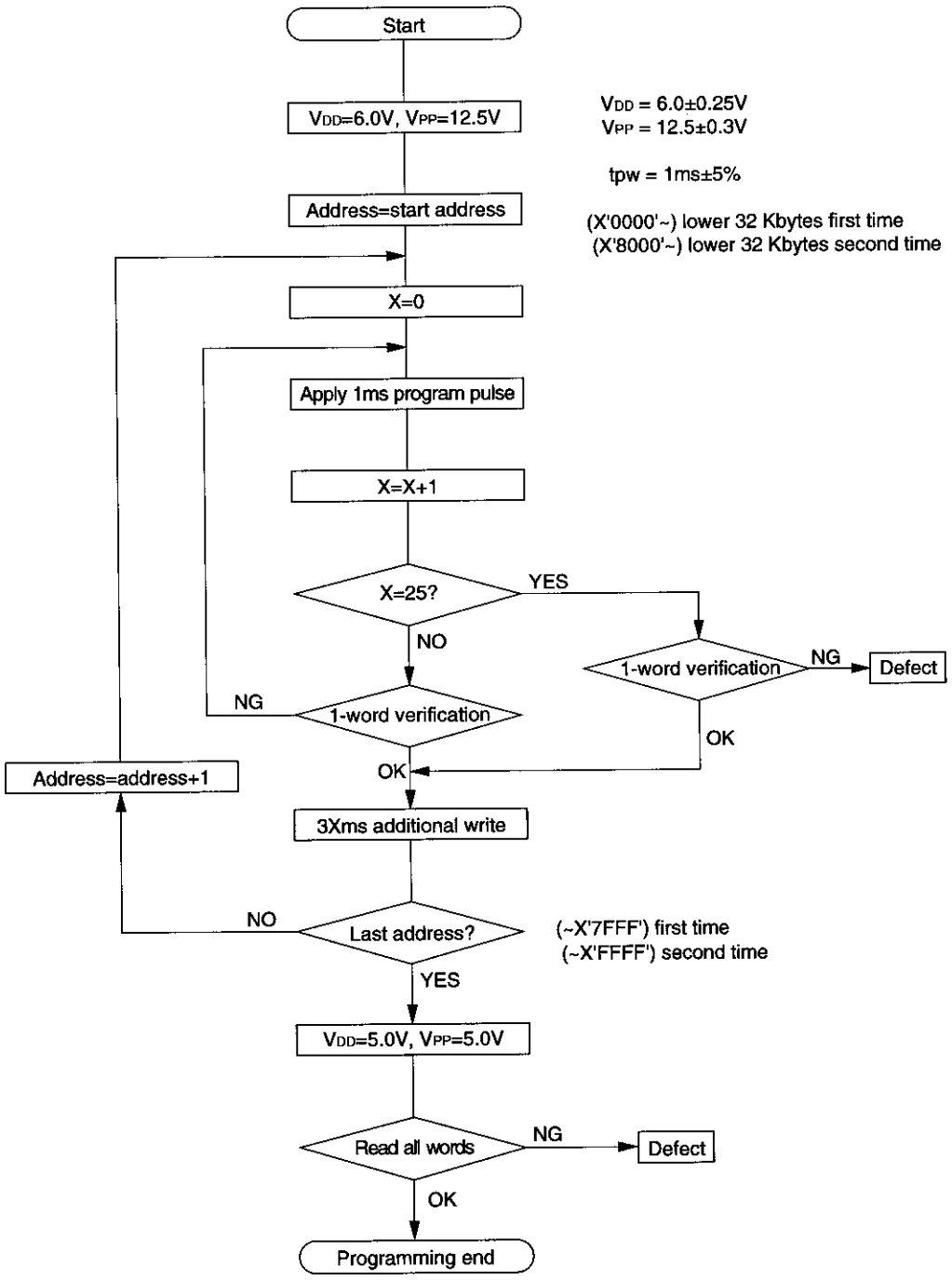


Figure 12-4-2 On-Chip EPROM Programming Flowchart (27C256 Mode)

(3) On-Chip EPROM Programming Electrical Characteristics

1. *Wait for the V_{DD} power supply (+6V) to settle before applying the V_{PP} power supply (+12.5V), and cut V_{PP} power off before V_{DD} power cut off.*
2. *Ensure that the V_{PP} pin voltage does not exceed +13.5V, including overshoot.*
3. *On no account remove the device while +12.5V is being applied to the Vpp pin, as this may affect the reliability of the device.*
4. *When $\overline{CE}$=V_{IL}, change the Vpp pin from V_{IL} to +12.5V, or from +12.5V to V_{IL}.*

■ DC characteristics (27C256 mode)

V_{DD}=6V±0.25V, V_{PP}=12.5V±0.3V, T_a=25°C±5°C

Item		Test Conditions	Allowable Values			Unit
			Min	Typ	Max	
Input leakage current	I_{LI}	V_{LI}=0V~V_{DD}	——	——	±10	µA
Output voltage	V_{OL}	I_{OL}=2.1mA	——	——	0.45	V
	V_{OH}	I_{OH}= -400µA	2.4	——	——	V
Input voltage	V_{IL}		V_{SS}	——	0 8	V
	V_{IH}		V_{DD}x0 7	——	V_{DD}	V
V_{DD} supply current (program/verify)	I_{DD}		——	——	TBD	mA
V_{PP} supply current (program)	I_{PP}	$\overline{CE}$=V_{IL}	——	——	TBD	mA

■ AC characteristics (27C256 mode)

V_{DD}=6V±0.25V, V_{PP}=12.5V±0.3V, T_a=25°C±5°C

Item	Test Conditions	Allowable Values			Unit
		Min	Typ	Max	
Address setup time	t_{AS}	2	—	—	µs
Output enable setup time	t_{OES}	2	—	—	µs
Data setup time	t_{DS}	2	—	—	µs
Address hold time	t_{AH}	0	—	—	µs
Data hold time	t_{DH}	2	—	—	µs
Vcc setup time	t_{VCS}	2	—	—	µs
V_{PP} setup time	t_{VPS}	2	—	—	µs
Program pulse width	t_{PW}	0 95	1	1 05	ms
Additional program pulse width	t_{APW}	2.85	—	78 75	ms

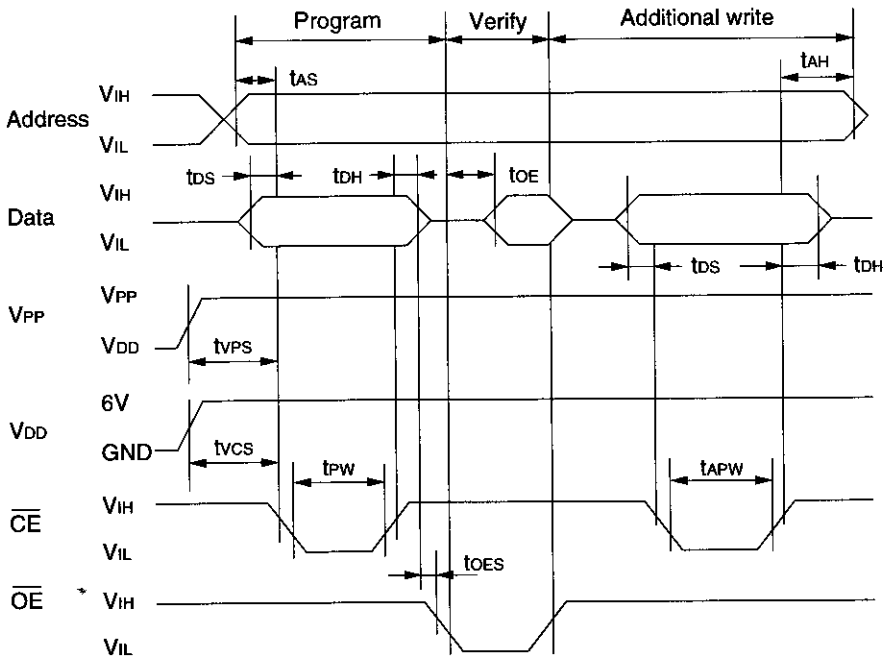


Figure 12-4-3 Input/Output Timing during Programming (27C256 Mode)

(4) Erasing On-Chip EPROM

A version of the MN18P76423 (the PX-AP18764323-FBC) is available in a ceramic package with a window that allows on-chip EPROM data to be erased by exposure of ultraviolet light. Data is erased by ultraviolet light with a wavelength of 254nm. The necessary quantity of radiation is 15W•s/cm^2. The window should be kept clean to ensure effective erasure.

Erasing Data in a Windowed Package Version

Data in an-chip EPROM product with a windowed package is erased (changed from 0 to 1) by exposing the chip to ultraviolet light (of 2537A wavelength) through the window.

(1) Erasing procedure

1. Ensure that the window is free of oil, adhesive, or other stains.
   *If necessary, clean the window with alcohol or a similar cleaning agent that will not damage the package.
2. Expose the package to ultraviolet light.
   Expose to a commercial ultraviolet lamp for 15 to 20 minutes at a distance of 2 to 3 cm (package surface illuminance: 12000 uW/cm^2). This will provide 10W•s/cm^2 ultraviolet irradiation, ensuring complete erasure of the data.
   *If the lamp has a filter, remove this before use.
   *Fitting a mirror reflector to the lamp will increase the illuminance by 40% to 80%, shortening the erase time.
   *Check the life of the lamp and ensure that full illumination is provided.

(2) Amount of ultraviolet irradiation

The recommended quantity of ultraviolet irradiation is 10W•s/cm^2. This figure includes a considerable safety margin, and is several times the quantity considered necessary to erase all bits. This figure should be strictly observed to ensure dependable erasure throughout the temperature and supply voltage ranges.

(3) Usage Notes

Data in windowed-package products with on-chip EPROM may be erased by exposure to light of a wavelength shorter than 4000A. Although the effect is far less than with ultraviolet light, dissipation of the data charge may be caused by fluorescent light or sunlight. Exposure to such irradiation may therefore affect system reliability in the long term. If a chip is used where it will be exposed to light of these wavelengths, the window should be covered with a seal to shield the EPROM.

Although light with a wavelength of 4000 to 5000A or more does not affect data retention, exposure to extremely bright light may cause abnormal circuit operation for reasons related to the characteristics of semiconductors in general. Therefore, protective measures should be taken if a chip is exposed to continuous irradiation of this kind, even in the case of light with a wavelength of 4000A or more.

(5) On-Chip EPROM Usage Notes

① When programming with a PROM programmer, check that the PROM programmer socket, programming adapter, and CPU are properly connected, to prevent the risk of damage to the CPU.

② The MN18P76423's programming voltage is 12.5V. Selecting the Fujitsu specification and applying 21.0V will destroy the chip. When using a PROM programmer, select the Intel 27C256 specification (V_{PP}=+12.5V) and program the upper and lower 32Kbytes separately.

When data erasure is performed by exposing ultraviolet light to a windowed package, the EPROM contents are changed to all 1s.

12-4-3 Appearance of MN18P76423 Products

(1) Packages

① MN18P7423-FB

The outer dimensions of the plastic package version of the MN18P76423 are identical to those of the MN1872423/MN1873223/MN1874023/MN1874823. See "1-7 Package Dimensions for details."

② PX-AP18764323-FBC

Outer views of the PX-AP18764323-FBC (84-pin ceramic flat package) are shown in figure 12-4-4. Note, in particular, the position of pin 1.

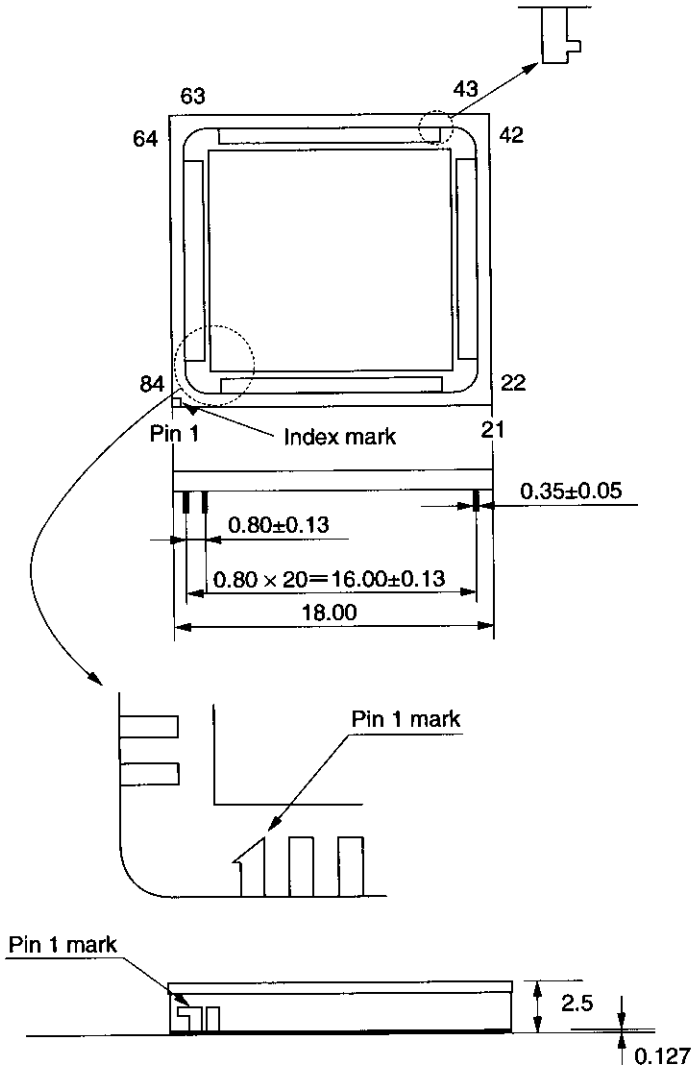


Figure 12-4-4 External Views of PX-AP1876423-FBC

(2) Programming adapter

Panasonic/Matsushita offers a programming adapter for converting the 84-pin package to the 28-pin 27C256 specification when writing data to the MN18P76423's on-chip EPROM. The part number of the programming adapter is OTP84QFP-1876423. The programming adapter has a switch (SW3) for selecting upper or lower addresses in 27C256 mode.

1. Use in 27C256 mode

   Setting SW3 to the right side enables addresses X'0000'~X'7FFF' to be written to.

   Setting SW3 to the left side enables addresses X'8000'~X'FFFF' to be written to.

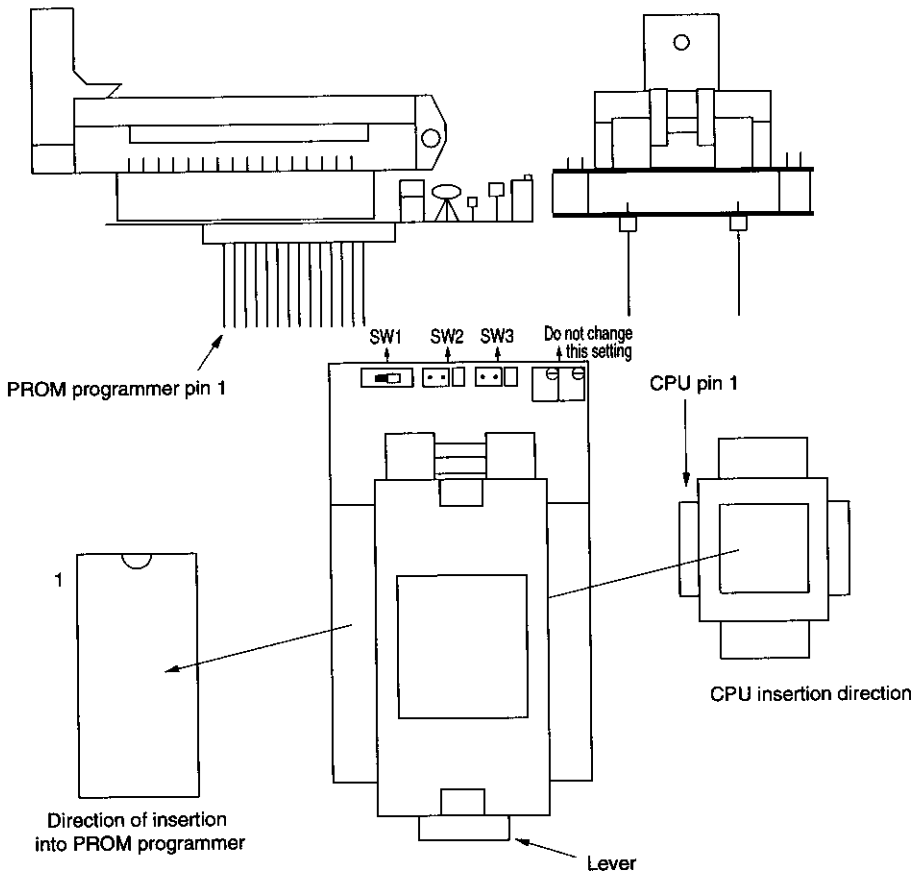


Figure 12-4-5 Connecting Programming Adapter

12-4-4 MN18P76423 Usage Notes

① Note that the MN18P76423 electrical characteristic shown in table 12-4-2 is different from the MN1872423/MN1873223/MN1874023/MN1874823.

Table 12-4-2 Difference in Electrical Characteristics

Model	Item	Test Conditions	Allowable Values			Unit
			Min	Typ	Max	
MN1872423/MN1873223/ MN1874023/MN1874823	Supply voltage	32kHz≤fx≤65kHz	2.7		5.5	V
MN18P76423			3.5		5.5	V

② The MN18P76423 does not have pull-down resistors incorporated into the high-voltage ports (P67~P60, P87~P80).

③ When program writing is completed, fix a lightproof shield over the glass areas above and at the side of the CPU to prevent erasure of data by ultraviolet light.

④ The ROM and RAM sizes of the MN18P76423 differ from those of the MN1872423/MN1873223/MN1874023/MN1874823. When ordering ROM, therefore, make sure that actual ROM and/or RAM sizes are used.
The on-chip ROM and RAM sizes of each model are shown in table 12-4-3.

Table 12-4-3 ROM and RAM Sizes of Each Model

Model	ROM	RAM
MN1872423	24,576 (24K) bytes (X'0000'~X'5FFF')	512 bytes (X'050'~X'22F', X'7E0'~X'7FF')
MN1873223	32.768 (32K) bytes (X'0000'~X'7FFF')	1024 bytes (X'050'~X'42F', X'7E0'~X'7FF')
MN1874023	40,960 (40K) bytes (X'0000'~X'9FFF')	1024 bytes (X'050'~X'42F', X'7E0'~X'7FF')
MN1874823	49,512 (48K) bytes (X'0000'~X'BFFF')	1024 bytes (X'050'~X'42F', X'7E0'~X'7FF')
MN18P76423	65,536 (64K) bytes (X'0000'~X'FFFF')	1968 bytes (X'050'~X'7FF')

The MN18P76423's oscillation mode can be changed according to the written value of bit 7 of EPROM address X'FFFF'.

Bit 7 of Address X'FFFF'	"0"	"1"
Oscillation mode	Type A	Type B

Type B is set when the product is shipped.

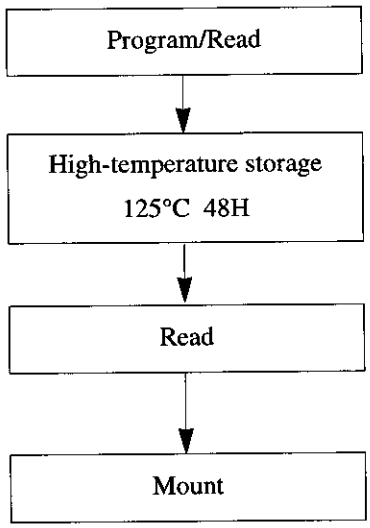
12-4-5 Notes on MN18P76423 Quality

(1) The following defects characteristic of one-time programmable microcomputers occur in the MN18P76423:

- Write defects
- Retention defects

In order to eliminate products with these defects, a write check and screening are carried out in the wafer stage. However, since write tests cannot be conducted for performance deficiencies caused by flaws that arise in the assembly stage onward, write defects and retention defects develop, occurring in a fixed ratio.

(2) To eliminate retention defects, high-temperature storage screening of the MN18P76423 is recommended before programming and mounting.



MN1872423/3223/4023/4823
LSI User's Manual

October,1996 1st Edition

Issued by Matsushita Electric Industrial Co., Ltd.
Matsushita Electronics Corporation

Semiconductor Company, Matsushita Electronics Corporation

Nagaokakyo, Kyoto, 617-8520 Japan
Tel: (075) 951-8151
http://www.mec.panasonic.co.jp

SALES OFFICES

■ U.S.A. SALES OFFICE

Panasonic Industrial Company [PIC]

- **New Jersey Office:**
  2 Panasonic Way, Secaucus, New Jersey 07094
  Tel: 201-392-6173
  Fax: 201-392-4652
- **Milpitas Office:**
  1600 McCandless Drive, Milpitas, California 95035
  Tel: 408-945-5630
  Fax: 408-946-9063
- **Chicago Office:**
  1707 N. Randall Road, Elgin, Illinois 60123-7847
  Tel: 847-468-5829
  Fax: 847-468-5725
- **Atlanta Office:**
  1225 Northbrook Parkway, Suite 1-151,
  Suwanee, Georgia 30174
  Tel: 770-338-6940
  Fax: 770-338-6849
- **San Diego Office:**
  9444 Balboa Avenue, Suite 185
  San Diego, California 92123
  Tel: 619-503-2940
  Fax: 619-715-5545

■ CANADA SALES OFFICE

Panasonic Canada Inc. [PCI]
5700 Ambler Drive Mississauga, Ontario, L4W 2T3
Tel: 905-624-5010
Fax: 905-624-9880

■ GERMANY SALES OFFICE

Panasonic Industrial Europe G.m.b.H. [PIEG]

- **Munich Office:**
  Hans-Pinsel-Strasse 2 85540 Haar
  Tel: 89-46159-156
  Fax: 89-46159-195

■ U.K. SALES OFFICE

Panasonic Industrial Europe Ltd. [PIEL]

- **Electric component Group:**
  Willoughby Road, Bracknell, Berkshire RG12 8FP
  Tel: 1344-85-3773
  Fax: 1344-85-3853

■ FRANCE SALES OFFICE

Panasonic Industrial Europe G.m.b.H. [PIEG]

- **Paris Office:**
  270, Avenue de President Wilson
  93218 La Plaine Saint-Denis Cedex
  Tel: 14946-4413
  Fax: 14946-0007

■ ITALY SALES OFFICE

Panasonic Industrial Europe G.m.b.H. [PIEG]

- **Milano Office:**
  Via Lucini N19, 20125 Milano
  Tel: 2678-8266
  Fax: 2668-8207

■ TAIWAN SALES OFFICE

Panasonic Industrial Sales Taiwan Co.,Ltd. [PIST]

- **Head Office:**
  6th Floor, Tai Ping & First Building No.550. Sec.4,
  Chung Hsiao E. Rd. Taipei 10516
  Tel: 2-2757-1900
  Fax: 2-2757-1906
- **Kaohsiung Office:**
  6th Floor, Hsien 1st Road Kaohsiung
  Tel: 7-223-5815
  Fax: 7-224-8362

■ HONG KONG SALES OFFICE

Panasonic Shun Hing Industrial Sales (Hong Kong) Co., Ltd. [PSI(HK)]
11/F, Great Eagle Centre, 23 Harbour Road,
Wanchai, Hong Kong.
Tel: 2529-7322
Fax: 2865-3697

■ SINGAPORE SALES OFFICE

Panasonic Semiconductor of South Asia [PSSA]
300 Beach Road # 16-01
The Concourse Singapore 199555
Tel: 390-3688
Fax: 390-3689

■ MALAYSIA SALES OFFICE

Panasonic Industrial Company (Malaysia) Sdn. Bhd.

- **Head Office: [PICM]**
  Tingkat 16B Menara PKNS PJ No.17,Jalan Yong
  Shook Lin 46050 Petaling Jaya Selangor Darul Ehsan
  Malaysia
  Tel: 03-7516606
  Fax: 03-7516666
- **Penang Office:**
  Suite 20-17,MWE PLAZA No.8,Lebuh Farquhar,10200
  Penang Malaysia
  Tel: 04-2625550
  Fax: 04-2619989
- **Johore Sales Office:**
  39-01 Jaran Sri Perkasa 2/1,Taman Tampoi
  Utama,Tampoi 81200 Johor Bahru,Johor Malaysia
  Tel: 07-241-3822
  Fax: 07-241-3996

■ CHINA SALES OFFICE

Panasonic SH Industrial Sales (Shenzhen) Co., Ltd. [PSI(SZ)]
7A-107, International Business & Exhibition Centre,
Futian Free Trade Zone, Shenzhen 518048
Tel: 755-359-8500
Fax: 755-359-8516

Panasonic Industrial (Shanghai) Co., Ltd. [PICS]
1F, Block A, Development Mansion, 51 Ri Jing Street,
Wai Gao Qiao Free Trade Zone, Shanghai 200137
Tel: 21-5866-6114
Fax: 21-5866-8000

■ THAILAND SALES OFFICE

Panasonic Industrial (Thailand) Ltd. [PICT]
252/133 Muang Thai-Phatra Complex Building,31st
Fl.Rachadaphisek Rd.,Huaykwang,Bangkok 10320
Tel: 02-6933407
Fax: 02-6933423

■ KOREA SALES OFFICE

Panasonic Industrial Korea Co., Ltd. [PIKL]
Hanil Group Bldg.11th Fl.,191 Hangangro 2ga,
Youngsans-ku,Seoul 140-702,Korea
Tel: 82-2-795-9600
Fax: 82-2-795-1542

■ PHILIPPINES SALES OFFICE

National Panasonic Sales Philippines [NPP]
102 Laguna Boulevard Laguna Technopark Sta.
Rosa. Laguna 4026 Philippines
Tel: 02-520-3150
Fax: 02-843-2778

080600
Printed in JAPAN