TOSHIBA

8-BIT MICROPROCESSOR TLCS-Z80,85

1990

TOSHIBA AMERICA ELECTRONIC COMPONENTS, INC.

TABLE OF CONTENTS

PART 1 TLCS-Z80 LSI DEVICES

CHAPTER 1 TLCS-Z80 FAMILY

CHAPTER 2 TLCS-Z80 ASSP FAMILY

PART 2 TLCS-85 LSI DEVICES

○ 8-BIT MICROPROCESSOR

TLCS-Z80 FAMILY MPUs

Series		TLCS-Z80			
Item \ Product Name / Unit		TMPZ84C00AP-6 TMPZ84C00AM-6 TMPZ84C00AT-6	TMPZ84C00AP-8 TMPZ84C00AM-8 TMPZ84C00AT-8	TMPZ84C01F	TMPZ84C02AF-6
Minimum Instruction Cycle Time	μS	0.6	0.5	1.0	0.6
Supply Voltage	V	4.5~5.5		4.5~5.5※	4.5~5.5
Supply Current Typ.	mA	15	20	15	20
Operating Temp.	℃	−40~85			
No. of Pins	—	40/44	40/44	44	44
Clock rate	MHz	DC~6	DC~8	DC~4	DC~6

060489

Series		TLCS-Z80 (ASSP)		
Item \ Product Name / Unit		TMPZ84C011AF-6	TMPZ84C013AT-6	TMPZ84C015AF-6
Minimum Instruction Cycle Time	μS	0.6	0.6	0.6
Supply Voltage	V	4.5~5.5		
Supply Current Typ.	mA	22	22	25
Operating Temp.	℃	−40~85		
No. of Pins	—	100	84	100
Clock rate	MHz	DC~6	DC~6	DC~6

060489

(Notes) ※ : Operatable under low supply voltage (2.7V (Min)) ;
Postfix M: SOP (Small Outline Package), P : DIP (Dual-in-Line Package)
T : PLCC (Plastic Leaded Chip Carrier), F : QFP (Quad Flat Package)

TLCS-Z80 FAMILY PERIPHERALs

Series		TLCS-Z80				
Item / Unit / Product Name		TMPZ84C10AP-6 TMPZ84C10AM-6 TMPZ84C10AT-6	TMPZ84C20AP-6 TMPZ84C20AM-6 TMPZ84C20AT-6	TMPZ84C30AP-6 TMPZ84C30AM-6 TMPZ84C30AT-6	TMPZ84C40AP-6 TMPZ84C40AM-6	TMPZ84C41AP-6 TMPZ84C41AM-6
Minimum Instruction Cycle Time	μS	—	—	—	—	—
Supply Voltage	V	4.5~5.5				
Supply Current Typ.	mA	6	3	4	4	4
Operating Temp.	℃	−40~85				
No. of Pins	—	40 / 44	40 / 44	28 / 44	40	40
Clock rate	MHz	DC~6	DC~6	DC~6	DC~6	DC~6

060489

Series		TLCS-Z80			
Item / Unit / Product Name		TMPZ84C42AP-6 TMPZ84C42AM-6	TMPZ84C43AF-6 TMPZ84C44AT-6	TMPZ84C61AP-6	
Minimum Instruction Cycle Time	μS	—	—	—	
Supply Voltage	V	4.5~5.5			
Supply Current Typ.	mA	4	4	3	4
Operating Temp.	℃	−40~85			
No. of Pins	—	40	44	16	16
Clock rate	MHz	DC~6	DC~6	DC~6	DC~8

060489

(Notes) Postfix P : DIP (Dual-in-line Package); T : PLCC (Plastic Leaded Chip Carrier)
F : QFP (Quad Flat Package); M: SOP (Small Outine Package)

TLCS-Z80 FAMILY 8MHz VERSION (PERIPHERALS & ASSP)

Series		TLCS-Z80 (PERIPHERALS)				
Item \ Unit / Product Name		TMPZ84C20AP-8	TMPZ84C30AP-8	TMPZ84C40AP-8	TMPZ84C41AP-8	TMPZ84C42AP-8
Minimum Instruction Cycle Time	μS	—				
Supply Voltage	V	4.75~5.25				
Supply Current Typ.	mA	4	5	5	5	5
Operating Temp.	℃	−10~70				
No. of Pins	—	40				
Clock rate	MHz	8				

060489

Series		TLCS-Z80 (ASSP)	
Item \ Unit / Product Name		TMPZ84C015AF-8	TMPZ84C013AT-8
Minimum Instruction Cycle Time	μS	0.5	
Supply Voltage	V	4.75~5.25	
Supply Current Typ.	mA	35	31
Operating Temp.	℃	−10~70	
No. of Pins	—	100	84
Clock rate	MHz	8	

060489

(Notes) Postfix P : DIP (Dual-in-line Package)
F : QFP (Quad Flat Package)

○ 8-BIT MICROPROCESSOR

TLCS-85 FAMILY (1/2)

Series		TLCS-85 (NMOS)				
Item \ Unit \ Product Name		TMP8085AP-2	TMP8085AHP-2	TMP8155P-2	TMP8156P-2	TMP8237AP-5
Minimum Instruction Cycle Time	μS	1.3 / 0.8	1.3 / 0.8	—	—	—
Supply Voltage	V	4.75~5.25	4.5~5.5	4.75~5.25		4.75~5.25
Supply Current Typ.	mA	170	135	180	180	150
Operating Temp.	℃	0~70				
No. of Pins	—	40	40	40	40	40
Clock rate	MHz	3 / 5	3 / 5	3 / 5	3 / 5	3 / 5

060489

Series		TLCS-85 (NMOS)				
Item \ Unit \ Product Name		TMP8251AP	TMP8253P-5	TMP8255AP-5	TMP8259AP	TMP8279P-5
Minimum Instruction Cycle Time	μS	—	—	—	—	—
Supply Voltage	V	4.75~5.25			4.5~5.5	
Supply Current Typ.	mA	100	140	120	85	120
Operating Temp.	℃	0~70				
No. of Pins	—	28	24	40	28	40
Clock rate	MHz	3	2.5	3 (TRD = 200ns)	3 (TRD = 200ns)	3

060489

(Note) Postfix P : DIP (Dual-in-line Package)

TLCS-85 FAMILY (2/2)

Series		TLCS-85 (CMOS)				
Item \ Unit \ Product Name		TMP82C51AP-2 TMP82C51AM-2	TMP82C51AP-10 TMP82C51AM-10	TMP82C53P-2	TMP82C54P-2 TMP82C54M-2	TMP82C55AP-2 TMP82C55AM-2
Supply Voltage	V	4.5~5.5				
Supply Current Typ.	mA	0.8	4	2 <0.5μA>	3	2 <0.5μA>
Operating Temp.	℃	−40~85				
No. of Pins	—	28	28	24	24	40
Clock rate	MHz	5	10	5	10	5 (TRD = 140ns)

060489

Series		TLCS-85 (CMOS)				
Item \ Unit \ Product Name		TMP82C55AP-10 TMP82C55AM-10	TMP82C255AN-2 TMP82C255AN-10	TMP82C265AF-2 TMP82C265AF-10	TMP82C59AP-2 TMP82C59AM-2	TMP82C79P-2 TMP82C79M-2
Supply Voltage	V	4.5~5.5				
Supply Current Typ.	mA	2 <0.5μA>	3	3	0.8	2
Operating Temp.	℃	−40~85				
No. of Pins	—	40	64	80	28	40
Clock rate	MHz	10 (TRD = 100ns)	8 / 10	8 / 10	8 (TRD = 120ns)	5

060489

Series		TLCS-85 (CMOS)	
Item \ Unit \ Product Name		TMP82C37AP-5 TMP82C37AM-5	*TMP82C37BP-8 *TMP82C37BM-8
Supply Voltage	V	4.5~5.5	
Supply Current Typ.	mA	5	8
Operating Temp.	℃	−40~85	
No. of Pins	—	40	40
Clock rate	MHz	5	8

060489

(Notes) * : under development; < > : Hold current;
Postfix P : DIP (Dual-in-line Package), F : QFP (Quad Flat Package)
N : Plastic shrink DIP, M : SOP (Small Outline Package)

TOSHIBA

PART 1 8-BIT MICROPROCESSOR

CHAPTER 1 TLCS-Z80 FAMILY

TOSHIBA CORPORATION

CONTENTS

TMPZ84C00AP-6 / TMPZ84C00AM-6 / TMPZ84C00AT-6
TMPZ84C00AP-8 / TMPZ84C00AM-8 / TMPZ84C00AT-8
TLCS-Z80 MPU : 8-BIT MICROPROCESSOR

1. OUTLINE AND FEATURES

The TMPZ84C00A is an 8-bit microprocessor (hereinafter referred to as MPU), which provides low power operation and high performance.

Built into the TMPZ84C00A are bus control, memory control and timing control circuits in addition to paired 6 general purpose registers, accumulator, flag registers and an arithmetic-and-logic unit.

The TMPZ84C00A is fabricated using Toshiba's CMOS Silicon gate Technology.

The pricipal functions and features of the TMPZ84C00A are as follows.

Table 1.1 Operating Frequency and Supply Current

Produce Name	Operating Frequency	Supply Current (TYP.)	
		AT RUN	AT STAND BY
TMPZ84C00AP-6/AM-6/AT-6	6MHz	15mA	0.5μA
TMPZ84C00AP-8/AM-8/AT-8	8MHz	20mA	0.5μA

060489

(1) Commands compatible with the Zilog Z80 MPU.

(2) Low power consumption

(3) DC to 8MHz operation (at 5V ± 10%)

(4) Single 5V power supply (5V ± 10%)

(5) Operating temperature (− 40°C to 85°C)

(6) Powerful set of 158 instrucitons available

(7) Powerful interrupt function

(a) Non-maskable interrupt terminal ($\overline{NMI}$)

(b) Maskable interrupt terminal ($\overline{INT}$)

The following 3 modes are selectable:

- 8080 compatible interrupt mode (interrupt by Non-Z80 family peripheral LSI) (Mode 0)
- Restart interrupt (Mode 1)

- Daisy chain structure interrupt using Z80 family peripheral LSI (Mode 2)

(8) An auxiliary resister provided to each of general purpose registers

(9) Two index registers

(10) 10 addressing modes

(11) Built-in refresh circuit for dynamic memory

(12) Molded in 40-pin DIP package (P), 40-pin SOP package (M) and 44-pin PLCC package (T).
Further, in the following text and explanations for charts and tables, hexadecimal numbers are directly used without giving an identification to explanation of address, etc. to the extent not to cause confusions.

Note : Z80 is a trademark of Zilog Inc., U.S.A.

2. PIN ASSIGNMENT AND FUNCTIONS

The pin assignment and I/O pin names and brief functions of TMPZ84C00A are shown below.

2.1 PIN ASSIGNMENT (TOP View)

The pin assignment of the TMPZ84C00A are as shown in Figure 2.1 and Figure 2.2.

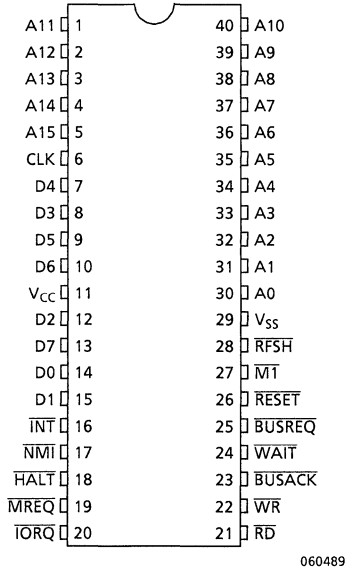


Figure 2.1 DIP, SOP Pin Assignment

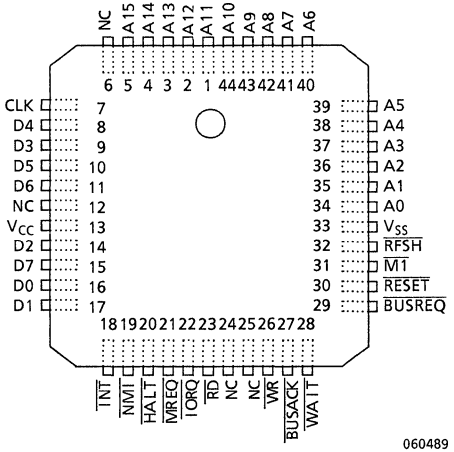


Figure 2.2 PLCC Package Pin Assignment

2.2 PIN NAMES AND FUNCTIONS

I/O pin names and functions are as shown Table 2.1.

Table 2.1 Pin names and Functions (1/2)

Pin	Q'ty (Number)	Type	Function
D0-D7	8	Input/output 3-state	The 8-bit bi-directional data bus.
A0-A15	16	Output 3-state	The 16-bit address bus. These pins specify memory and I/O port addresses. During a refresh cycle, the refresh address is output.
$\overline{M1}$	1	Output	The Machine Cycle 1 signal. In an operation code fetch cycle, this pin goes "0" with the $\overline{MREQ}$ signal. At the execution of a 2-byte operation code, this pin goes "0" for each operation code fetch. In a maskable interrupt acknowledge cycle, this pin goes "0" with the $\overline{IORQ}$ signal.
$\overline{RD}$	1	Output 3-state	The Read signal. It indicates that the MPU is ready for accepting data from memory or I/O device. The data from the addressed memory or I/O devices is gated by this signal onto the MPU data bus.
$\overline{WR}$	1	Output 3-state	The Write signal. This signal is output when the data to be stored in the addressed memory or I/O device is on the data bus.
$\overline{MREQ}$	1	Output 3-state	The Memory Request signal. When the execution address for memory access is on the address bus, this pin goes "0" . During a memory refresh cycle, this pin also goes "0" with $\overline{RFSH}$ signal.
$\overline{IORQ}$	1	Output 3-state	The Input/Output Request signal. This pin goes "0" when the address for an I/O read or write operation is on the low-order 8 bits (A0 through A7) of the address bus. The $\overline{IORQ}$ signal is also output with the $\overline{M1}$ signal at interrupt acknowledge to tell an I/O device that the interrupt response vector can be placed on the data bus.
CLK	1	Input	The Single-phase Clock Input. When the clock input is placed in the DC state (continued "1" or "0" level), this pin stops operating and holds the state of that time.

060489

(2/2)

Pin	Q'ty (Number)	Type	Function
$\overline{RESET}$	1	Input	The Reset signal input. $\overline{RESET}$ signal is used for initialization MPU and must be kept in active state ("0") for a period of at least 3 clocks.
$\overline{INT}$	1	Input	The Maskable Interrupt signal. An interrupt is caused by the peripheral LSI. An interrupt is acknowledged when the interrupt enable flip-flop (IFF) is set to "1" by software. The $\overline{INT}$ pin is normally wire-ORed and requires an external pullup resistor for these applications.
$\overline{WAIT}$	1	Input	The Wait Request signal. This signal indicates to the MPU that the addressed memory or I/O device is not ready for data transfer. As long as this signal is "0", the MPU is in the Wait state.
$\overline{BUSREQ}$	1	Input	The bus Request signal. The $\overline{BUSREQ}$ signal forces the MPU address bus, data bus, and control signals $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ to be placed in the high-impedance state. This signal is normally Wire-ORed and requires an external pullup resistor for these applications.
$\overline{BUSACK}$	1	Output	The Bus Acknowledge signal. In response to the $\overline{BUSREQ}$ signal, the $\overline{BUSACK}$ signal indicates to the requesting peripheral LSI that the MPU address bus, data bus, and control signals $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and WR have been put in the high-impedance state.
$\overline{HALT}$	1	Output	The Halt signal. This pin goes "0" when the MPU has executed a Halt instruction and is in the Halt state.
$\overline{RFSH}$	1	Output	The refresh signal. When the dynamic memory refresh address is on the low-order 8 bits of the address bus, this signal goes "0" . At the same time, the $\overline{MREQ}$ signal also goes active ("0").
$\overline{NMI}$	1	Input	The Non-maskable Interrupt Request signal. This interrupt request has a higher priority than the maskable interrupt and is not dependent on the interrupt enable flip-flop (IFF) state.
NC (PLCC only)	4	–	Not connected internaly. Please use by open.
V_{CC}	1	power supply	+ 5 V
V_{SS}	1	power supply	0 V

060489

3. FUNCTIONAL DESCRIPTION

The system configuration, functions and basic operation of the TMPZ84C00A are described here.

3.1 BLOCK DIAGRAM

The block diagram of the internal configuration is shown in Figure 3.1.

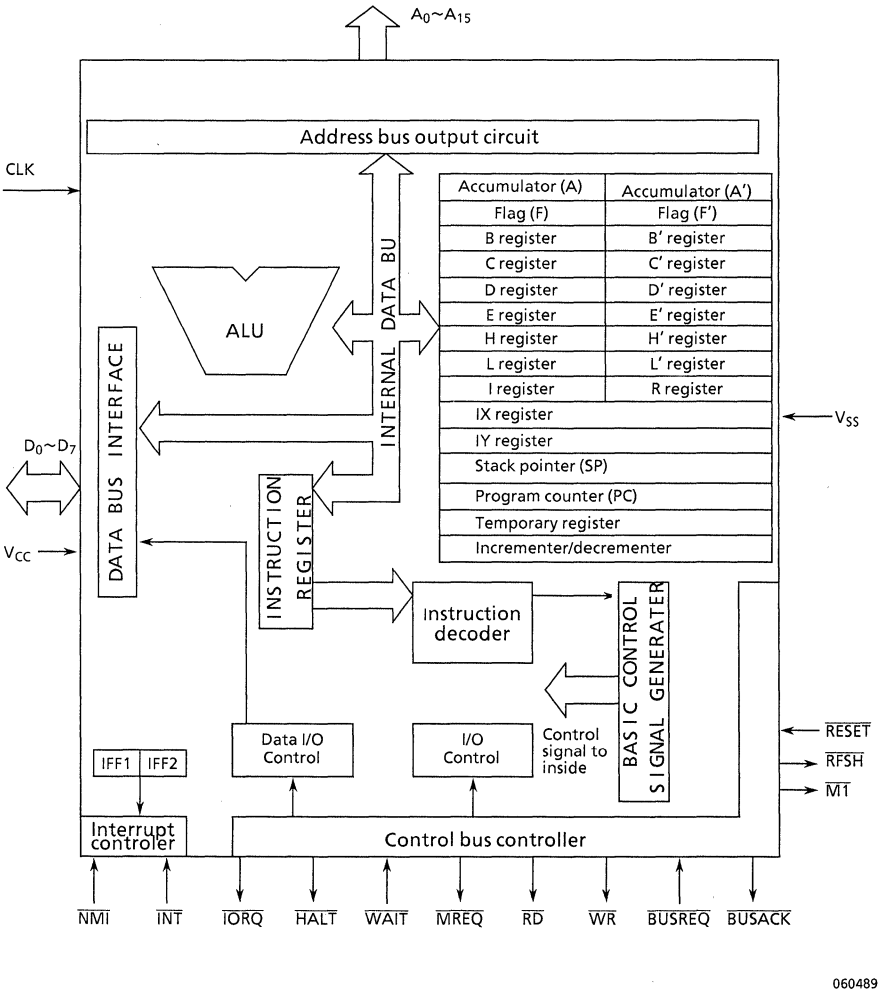


Figure 3.1 Block Diagram

3.2 SYSTEM CONFIGURATION

The MPU has the configuration shown in Figure 3.1. The address signal is put on the address bus via the address buffer. The data bus is controlled for input or output by the data bus interface. Both the address and data buses are put in the high-impedance state by the $\overline{BUSEQ}$ signal input to make them available for other peripheral LSIs. The Opcode read from memory via the data bus is written to the instruction register. This Opcode is decoded by the instruction decoder. According to the result of the decoding, control signals are sent to the relevant devices. Receiving these control signals, the ALU performs arithmetic operations. The register array temporarily hold the information required to perform operation.

The following describes the MPU's main components and functions which the user must understand to operate the TMPZ84C00A.

[1] Internal Register Groups

The configuration of the internal register groups is as follows:

(1) Main registers

A, F, B, C, D, E, H, L

(2) Alternate registers

A', F', B', C', D', E', H', L'

(3) Special purpose registers

I, R, IX, IY, SP, PC

Figure 3.3 shows the configuration of the internal register groups. The register groups, each being of a static RAM, consists of eighteen 8-bit registers and four 16-bit registers. The following describes the function of each register:

(1) Main registers (A, F, B, C, D, E, H, L)

(a) Accumulator (A)

The accumulator is an 8-bit register used for arithmetic and data transfer operations.

(b) Flag register (F) (see Fig. 3.2)

The flag register is an 8-bit register to hold the result of each arithmetic operation. Actually, the 6 of the 8 bits are set ("1")/reset ("0") according to the condition specified by an instruction.

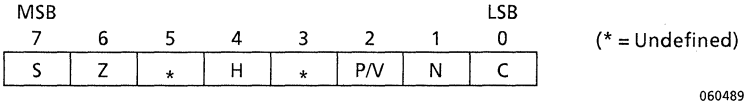


Figure 3.2 Flag Register Configuration

The following 4 bits are directly available to the programmer for setting the jump, call and return instruction conditions:

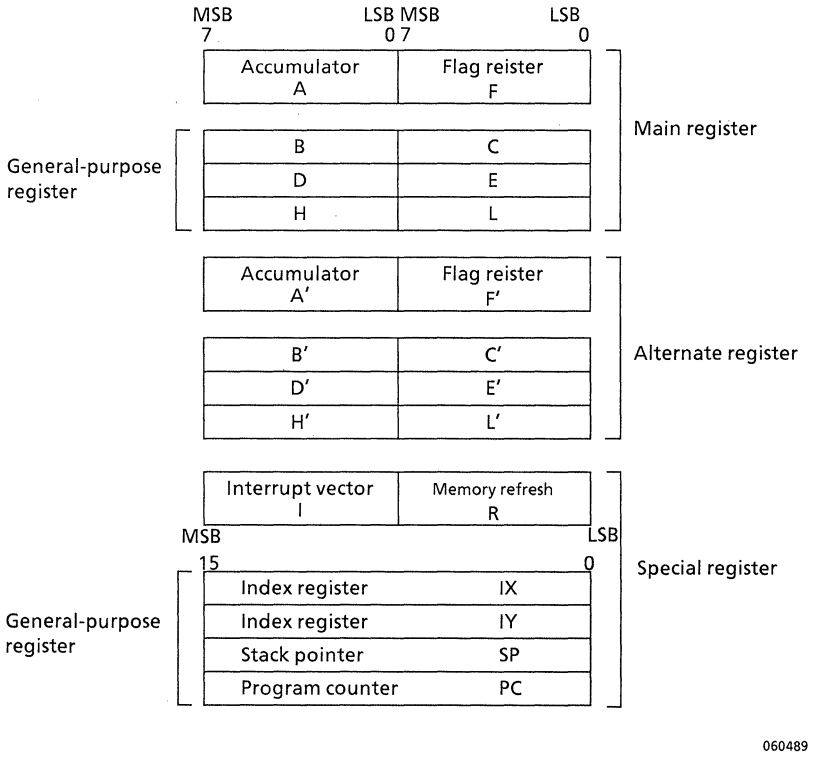


Figure 3.3 Flag Register Configuration

- Sign flag (S)

  When the result of an operation is negative, the S flag is set to "1" . Actually, the content of bit 7 of accumulator is stored in this flag.

- Zero flag (Z)

  When all bits turn out to be "0" s after operation, the Z flag is set to "1" . Otherwise, it is set to "0".

With a block search instruction (CPI, CPIR, CPD or CPDR), the Z flag is set to "1" if the source data and the accumulator data match.

With a block I/O instruction (INI, IND, OUTI or OUTD), the Z flag is set to "1" if the content of the B register used as the byte counter is "0" at the end of comparison.

- Parity/overflow flag (P/V)

This flag has two functions. One is the parity flag (P) that indicates the result of a logical operation (AND A, B etc.). The P flag is set to "1" if the parity is even as a result of the operation on signed values by two's complement. It is reset to "0" if the parity is odd. With a block search instruction (CPI, CPIR, CPD or CPDR) and a block transfer instruction (LDI or LDD), the P flag indicates the state of the byte counter (register pair B and C). It is set to "1" if the byte counter is not "0" and reset to "0" when the byte counter becomes "0" (at the end of comparison or data transfer). The content of the interrupt enable flip-flop (IFF) is saved to the P flag when the contents of the R register or I register are transferred to the accumulator.

The other use of the P/V flag is the overflow flag (V) that indicates whether an overflow has occurred or not as a result of an arithmetic operation. The V flag is set to "1" when the value in the accumulator gets out of a range of the maximum value +127 and the minimum value −128 and therefore cannot be correctly represented as a two's complement notation.

Whether the P/V flag operates as the P flag or V flag is determined by the type of the instruction executed.

- Carry flag (C)

The C flag is set to "1" if a carry occurs from bit 7 of the accumulator or a borrow occurs as a result of an operation.

The following two flags are not available to the programmer for the test and set ("1")/reset ("0") purposes. They are internally used by the MPU for BCD arithmetic operations.

  - Half carry flag (H)

The H flag is used for holding the carry or borrow from the low-order 4 bits of a BCD operation result. When a DAA instruction (decimal adjust) is executed, the MPU automatically uses the H flag to adjust the result of a decimal addition or subtraction.

  - Add/subtract flag (N)

In BCD operation, algorithm is different between addition and subtraction. The N flag indicates whether the executed operation is addition or subtraction.

For change of the flag state depending on the instruction, see 3.4 "TMPZ84C00A Instruction Set".

(c) General-purpose registers (B, C, D, E, H, L)

General-purpose registers consist of 8 bits each. They are used as 16-bit register pairs (BC, DE, HL) as well as independent 8-bit registers to supplement the accumulator. The B register and the register pair BC are used as counters when a block I/O, block transfer, or search instruction is executed. The register pair HL has various memory addressing features as compared with the register pairs BC and DE.

(2) Alternate registers (A', F', B', C', D', E', H', L')

The configuration of the alternate registers is exactly the same as that of the main registers. There is no instruction that handles the alternate registers directly. The data in the alternate registers are processed by moving them into the main registers by means of exchange instructions as shown below:

EX AF, AF' (A↔A', F↔F')

EXX (B↔B', C↔C', D↔D', E↔E', H↔H', L↔L')

When a high-speed interrupt response has been requested within the system, these instruction can be used to quickly move the contents of the accumulator, flag registers, and general-purpose registers into the corresponding registers. This eliminates the need for transferring the register contents to/from the external stack during execution of the interrupt handling routine, thereby shortening the interrupt servicing time greatly.

(3) Special purpose registers (I, R, IX, IY, SP, PC)

(a) Interrupt page address register (I)

The TMPZ84C00A provides two kinds of interrupts :maskable interrupt (INT) and non-maskable interrupt (NMI). The maskable interrupt provides three modes (0, 1, and 2) in which the interrupt is handled. These modes can be selected by instructions IM0, IM1, and IM2 respectively. In Mode 2, any memory location can be called indirectly depending on the interrupt. For this purpose, the I register stores the high-order 8 bits of the indirect address. The low-order 8 bits are supplied from the interrupting peripheral LSI. This scheme permits calling the interrupt handling routine from any memory location in an extremely short access time. For the details of interrupts, see [4] "Interrupt Capability".

(b) Memory refresh register (R)

The R register is used as the memory refresh counter when the dynamic RAM is used for memory. This permits using of the dynamic memory in the same manner as the static memory. The Low-order 7 bits of this 8-bit register is automatically incremented for each instruction fetch. While the MPU decodes and executes the fetched instruction, the contents of the R register are synchronized with the refresh signal to place the low-order 8 bits on the address bus. This operation is all performed by the MPU and, therefore, dose not need a special processing by program. The MPU operation is not delayed by this operation. During refresh, the contents of the I register are placed on the high-order 8 bits of the address bus.

(c) Index registers (IX, IY)

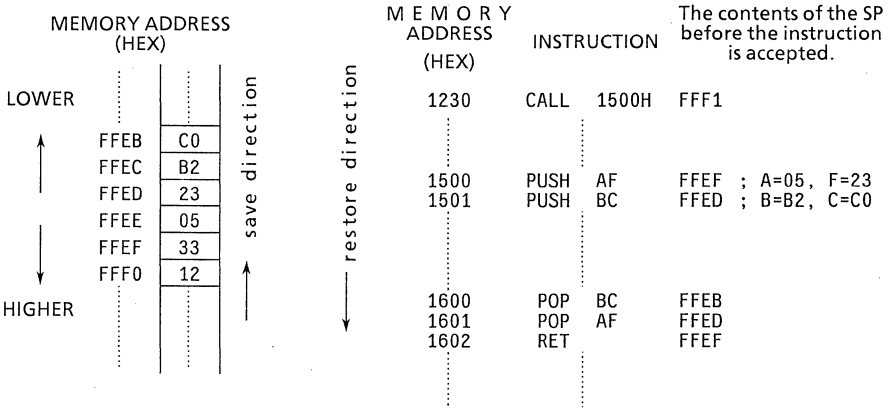
The two independent index registers IX and IY hold the 16-bit base address when used in the index addressing mode. In this addressing mode, the memory address obtained by adding the contents of an index register to the displacement value (for example, LD IX+40H) is specified. This mode is convenient for using data tables. Also these registers can be used separately for memory addressing and data retaining registers.

(d) Stack pointer (SP)

The stack pointer is a 16-bit register to provide the start address information in the stack area in the external RAM. The content of the stack pointer is decremented at the execution of a CALL instruction or PUSH instruction or interrupt handling and is incremented at the execution of a return instruction or POP instruction. At the execution of a CALL instruction or interrupt handling, the current content of the program counter is saved into the stack. At the execution of a return instruction, the content is restored from the stack to the program counter. These operations are all performed by the MPU automatically. However, the other registers are not saved or restored automatically. For the storing of the contents of these registers, an exchange instruction (EX or EXX) for alternate register, a PUSH or a POP instructions must be used. When a PUSH instruction is executed, the contents of the specified register are saved into the stack. When a POP instruction is executed, the contents of the stack are moved to the specified register.

These data are restored on a last-in, first-out basis. Use of the stack permits processing of multiple-level interrupts, deep subroutine nestings, and various data manipulation very easily. The stack pointer is not initialized in the hardware approach. Therefore, it is required to allocate the stack area in RAM to specify initialization (at the highest address of the stack area) in the initial program.

(ex)



The foregoing example shows the stack pointer and stack operations in which the instructions starting with the CALL at address 1230H and ending with the RET at address 1602H have been executed. However, it is assumed that there is no instruction or interrupt other than shown above that uses the stack during the execution. When the value the stack pointer before executing the CALL instruction at address 1230H indicates address FFF1H, address 1233H is stored at addresses FFF0H and FFEFH because the CALL instruction consists of 3 bytes, then the stack pointer is decremented. Similarly, the data are saved or restored sequentially according to the instructions. These stack and stack pointer operations are all performed automatically.

(e) Program counter (PC)

The program counter holds, in 16 bits, the memory address of the instruction to be executed next. The MPU fetches the instruction from the memory location indicated by the program counter. When the content of the program counter is put on the address bus, the program counter is incremented automatically. However, it is not incremented with a jump instruction, a call instruction, or interrupt processing. Instead, the specified new address is set on it. With a return instruction, the content restored from the stack is set on the program counter. These operations are all performed automatically and therefore, no care is required for programming.

[2] Halt Capability

When a HALT instruction has been executed, the MPU is put in the halt state. The halt capability can be used to halt the MPU against the external interrupts, thereby reducing the power dissipation. In the halt state the states of MPU's internal registers are retained. The halt state is cleared by reset or when an interrupt is accepted. For the details of halt operation, see [3] "Basic Timing".

(1) Halt operation

When a HALT instruction has been executed, the MPU sets the $\overline{\text{HALT}}$ signal to "0" to indicate that the MPU is going to be put in the halt state. Actually, the MPU in the halt state automatically continues executing NOP instructions if there is the system clock input. However, the program counter is not incremented. This keeps the refresh signal generated when the dynamic memory is used. During halt, the MPU's internal states are retained. By using TLCS-Z80's clock generator/controller (TMPZ84C60P or TMPZ84C61AP), the clock input control for these halt operations is realized easily.

(2) Releasing the halt state

The halt state is cleared by accepting an interrupt (the $\overline{\text{INT}}$ or $\overline{\text{NMI}}$ signal input) or by reset (the $\overline{\text{RESET}}$ signal input). When an interrupt is accepted, the halt state is cleared and the interrupt handling routine is executed. However, a maskable interrupt (INT) cannot be accepted unless the interrupt enable flip-flop (IFF) is set.

Note that when the halt state is cleared by the $\overline{\text{RESET}}$ signal, the MPU is reset and the program counter is set to "0".

[3] RESET Signal

Holding the $\overline{\text{RESET}}$ pin at the low level ("0") under the following conditions, the MPU's internal states are reset:

(1) The supply voltage level is within the operational voltage range.

(2) System clock stabilization.

(3) Holding the $\overline{\text{RESET}}$ signal at the low level ("0") for at least 3 full clock cycles. When the $\overline{\text{RESET}}$ signal goes high ("1"), the MPU starts executing instructions from address 0000H after at least 2T state dummy cycles.
When reset, the MPU performs the following processing:

(1) Program counter

0000H is set.

(2) Interrupt

The interrupt enable flip-flop (IFF) is reset to "0" to disable the maskable interrupt. For the maskable interrupt processing, mode 0 is specified.

(3) Control output

All control outputs are made inactive ("1"). Therefore, the halt state is also cleared.

(4) Interrupt page address register (I register)

The content of the R register becomes 00H.

(5) Refresh register (R register)

The content of the R register becomes 00H.

The contents of the registers other than above and the external memory do not change.

Therefore, they must be initialized as required.

[4] Interrupt Capability

The interrupt capability is used to suspend the execution of the current program and execute the processing of the requested peripheral LSI. Normally, this interrupt processing routine contains the data exchange and transfer of status and control information between the MPU and the peripheral LSI. When this routine has been completed, the MPU returns to the active state before the interrupt has been accepted.

The TMPZ84C00A provides the non-maskable interrupt (NMI) and maskable interrupt (INT) capabilities which are detected by the $\overline{\text{NMI}}$ and $\overline{\text{INT}}$ interrupt request signals, respectively. A non-maskable interrupt, when caused by a peripheral LSI, is accepted unconditionally. This interrupt is used to support critical functions such as the protection of the system from unpredictable happening including power failure. A maskable interrupt can be enabled or disabled by program. For example, if the timer is used and, therefore, an interrupt is not desired, the system can be programmed to disable the interrupt. Table 3.1 lists the processing by interrupt source.

(1) Interrupt enable/disable

A non-maskable interrupt cannot be disabled by program, while a maskable interrupt can be enabled or disabled by program. The MPU has the interrupt enable flip-flop (IFF). A maskable interrupt can be enabled or disabled by setting this flip-flop to "1" (set) or "0" (reset) through an EI instruction (enable) or a DI instruction (disable) in program. Actually, the IFF consists of two flip-flops IFF1 and IFF2. IFF1 is used to select between the enable and disable of a maskable interrupt. IFF2 holds the state of IFF1 before a maskable interrupt has been accepted. Both IFF1 and IFF2 are reset to "0" when any of the following conditions occurs, disabling an interrupt:

- MPU reset
- Execution of DI instruction
- Acceptance of maskable interrupt

Both IFF1 and IFF2 are set to "1" when the the following condition occurs, enabling an interrupt:

- Execution of EI instruction

Actually, the waiting maskable interrupt request is accepted after the execution of the instruction that follows the EI instruction.
This delay by one instruction is caused by accepting an interrupt after completion of the execution of a return instruction if the instruction following the EI instruction is a return instruction.
In the above operation, the contents of IFF1 and IFF2 are the same.

Table 3.1 Processing by Interrupt Source

Interrupt Source	Priority	Programmed condition		Vector address	Interrupt return instruction
Non-maskable interrupt (the falling edge of $\overline{NMI}$)	1	None		Address 66H	RETN
Maskable interrupt ($\overline{INT}$ becomes "0" at instruction's last clock)	2	IFF = 1	Mode 0	Instruction from peripheral LSI. Normally, CALL or RST instruction.	(Note) RET I
			Mode 1	Address 38H.	
			Mode 2	The address indicated by the data table (memory) at the address specified by I register (high-order 8 bits) and data from peripheral LSI (low-order 8 bits, LSB = "0").	

060489

Note : Mode 0 applies when the instruction from peripheral LSI is CALL or RST instruction.

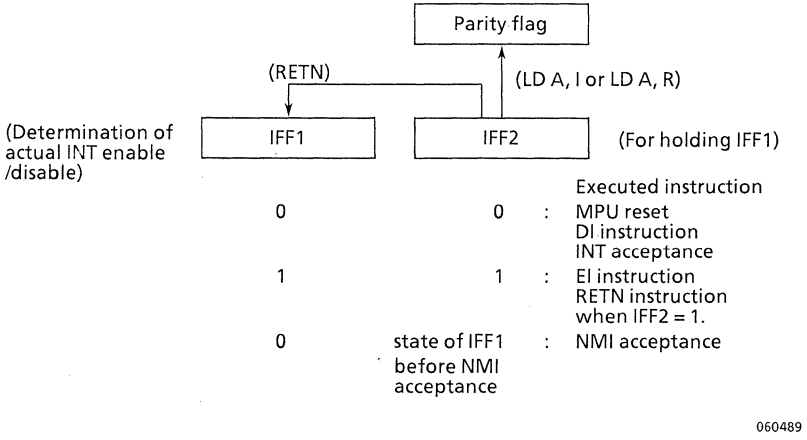


Figure 3.4 Interrupt Enable Flip-Flop (IFF)

When a non-maskable interrupt has been accepted, IFF1 is reset to "0" (interrupt disable) until an EI or RETN instruction is executed, so as to prevent from accepting the next interrupt. For this purpose, the state (interrupt enable/disable) of IFF1 immediately before non-maskable interrupt acceptance must be stored. This state is copied into IFF2 upon acceptance of a non-maskable interrupt. The content of IFF2 is copied into the parity flag at the execution of the following instructions, so that the copied data can be tested or stored:

- The load instruction (LD A, I) to load the contents of the I register into the accumulator.
- The load instruction (LD A, R) to load the contents of the R register into the accumulator.

When the return instruction (RETN) from the non-maskable interrupt is executed, the contents of the current IFF2 are copied back to IFF1. If an operation which changes the contents of IFF2 (due to the execution of EI or DI instruction, for example) has not been performed during interrupt handling, IFF1 automatically returns to the state immediately before the interrupt acceptance. Table 3.2 lists the states of IFF1 and IFF2 after execution of interrupt-related instructions.

Table 3.2 State of IFF1 and IFF2

Operation sequence	IFF1	IFF2	Remarks
NPU reset	0	0	
EI	1	1	
NMI acceptance	0	1	
LD A, I	*	*	Parity flag←IFF2
RETN	1	1	IFF1←IFF2
LD A, R	*	*	Parity flag←IFF2
INT acceptance	0	0	
RETI	*	*	
EI	1	1	
NMI acceptance	0	1	
DI	0	0	
RETN	*	*	

Note : * = no change

060489

(2) Interrupt processing

With a non-maskable interrupt, the internal NMI flip-flop is set to "1" on the falling edge of the interrupt signal, $\overline{NMI}$. The state of this flip-flop is sampled on the rising edge of the last clock of each instruction to accept an interrupt. A maskable interrupt is accepted if the interrupt signal $\overline{INT}$ is low ("0") on the rising edge of the last clock of each instruction and the interrupt enable state (IFF = 1 and $\overline{BUSREQ}$ signal = inactive ("1")) is on. The following is the processing to be performed after a non-maskable interrupt and a maskable interrupt are accepted:

(a) Non-maskable interrupt (NMI)

When a non-maskable interrupt has been accepted, the MPU performs the following processing:

1 The internal NMI flip-flop is reset to "0".

2 IFF1 is reset to "0", disabling the maskable interrupt.

The contents of the IFF1 immediately before the interrupt acceptance are copied into the IFF2.

3 The contents of the current program counter are saved into the stack.

4 The instructions starting from non-maskable interrupt vector address 66H are executed.

A non-maskable interrupt processing program terminates after executing the RETN instruction. This return instruction performs the followings:

1 The contents of the current IFF2 are copied into IFF1.

2 The contents of the program counter are restored from the stack.

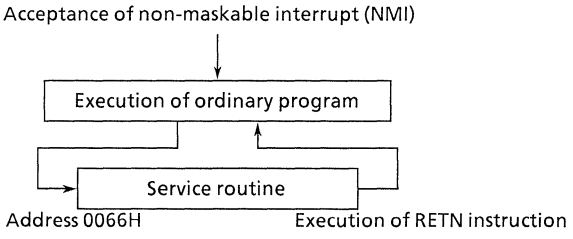


Figure 3.5 Non-Maskable Interrupt Processing

(b) Maskable interrupt (INT)

When a maskable interrupt has been accepted, the MPU performs the following processings:

1 Both IFF1 and IFF2 are reset to "0", disabling the maskable interrupts.

2 The contents of the current program counter are saved into the stack.

3 A maskable interrupt is serviced in one of the three modes 0, 1 and 2. A mode is selected by executing the instruction IM0, IM1or IM2 before the interrupt is serviced. The instructions are executed starting from the vector address corresponding to the selected mode.

- Mode 0

In mode 0, the interrupting peripheral LSI puts a restart instruction (RST) or a call instruction (CALL) on the data bus and the MPU executes the interrupt service routine according to that instruction. At reset, this mode is automatically set.

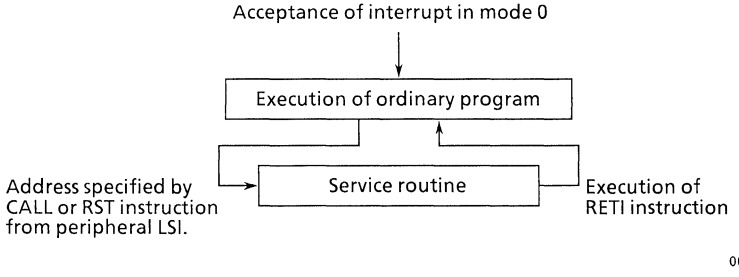


Figure 3.6 Interrupt Processing in Mode 0

- Mode 1

When an interrupt is accepted in mode 1, restart is performed from address 0038H. Therefore, the service routine for this interrupt is programmed from the address 0038H .

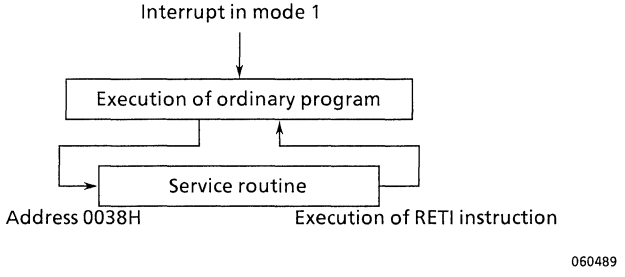


Figure 3.7 Interrupt Processing in Mode 1

- Mode 2

The interrupt processing in mode 2 requires a 16-bit pointer consisting of the high-order 8 bits of the I register and the low-order 8 bits (with the LSB="0") of the data fetched from the TLCS-Z80 family peripheral LSI. Therefore, the necessary value must be loaded in the I register beforehand. This pointer is used to specify the memory address in the table. The contents of the specified address and the next address provide the start address of the service routine. Therefore, use of this mode requires the table of the service routine's start address (16 bits) to be set at appropriate location under software control . This location can be anywhere in memory.

The LSB of the table pointer is set to "0" because a 2-byte data is needed to specify the service routine start address in 16 bits and start that address from an even-number address. In the table, the start address begins with the low-order byte followed by the high-order byte as shown in Figure 3.8.

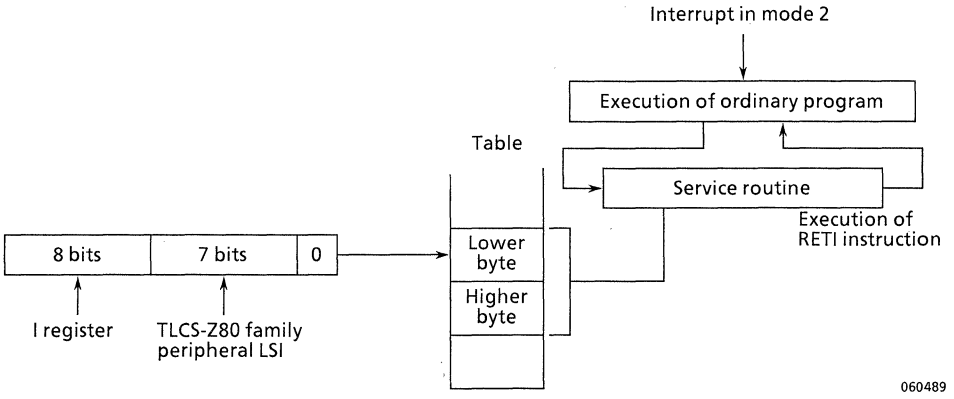


Figure 3.8 Interrupt Processing in Mode 2

Mode 2 is used in the daisy chain interrupt processing using TLCS-Z80 family LSI. TLCS-Z80 family peripheral LSIs all contain the interrupt priority controller in daisy chain structure. In this interrupt structure, the interrupt request signals are connected one after another and given priorities for processing when two or more maskable interrupt requests occur at a time. Only the interrupt vector from the peripheral LSI having the highest priority is put on the data bus. By receiving the interrupt vector in mode 2, the processing for that peripheral LSI can be performed. When an interrupt requested by a peripheral LSI having a priority higher than that of the current peripheral LSI during the execution of the interrupt processing routine, the higher priority interrupt can be enabled by the EI instruction to form an interrupt nesting.

The maskable interrupt processing program terminates by executing an RETI instruction. This return instruction performs the following processings:

- Restores the content of the program counter from the stack.
- Notifies the requesting peripheral LSI of the termination of interrupt processing.

3.3 MPU STATUS TRANSITION DIAGRAM AND BASIC TIMING

The following describes the MPU status transition and the basic timing of each MPU operation.

[1] Instruction Cycle

Each TMPZ84C00A instruction is executed by combining the basic operations of memory read/write, input/output, bus request/acknowledge, and interrupt. These basic operations are performed synchronizing with the system clock (the CLK signal).

One clock period is called a state (T). The smallest unit of each basic operation is called a machine cycle (M). Each instruction consists of 1 to 6 machine cycles and each machine cycle consists of 3 to 6 clock states basically. However, the number of clock states in a machine cycle can be increased by the $\overline{WAIT}$ signal described later on. Figure 3.9 shows an example of the basic timing of a 3-machine-cycle instruction.

The first machine cycle (M1) of each instruction is the cycle in which the Opcode of the instruction to be executed next is read (this is called the Opcode fetch cycle). The Opcode fetch cycle basically consists of 4 to 6 clock states. In the machine cycle that follows the Opcode fetch cycle, data is transferred between the MPU and the memory or peripheral LSIs. This operation basically consists of 3 to 5 clock states.

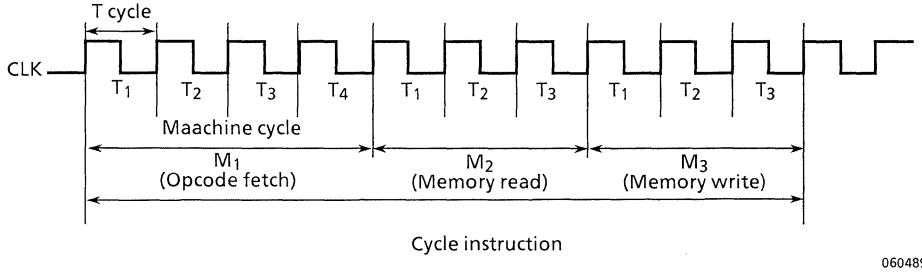


Figure 3.9 Example of MPU Basic Timing (3-Machine-Cycle Instruction)

[2] Status Transition Diagram

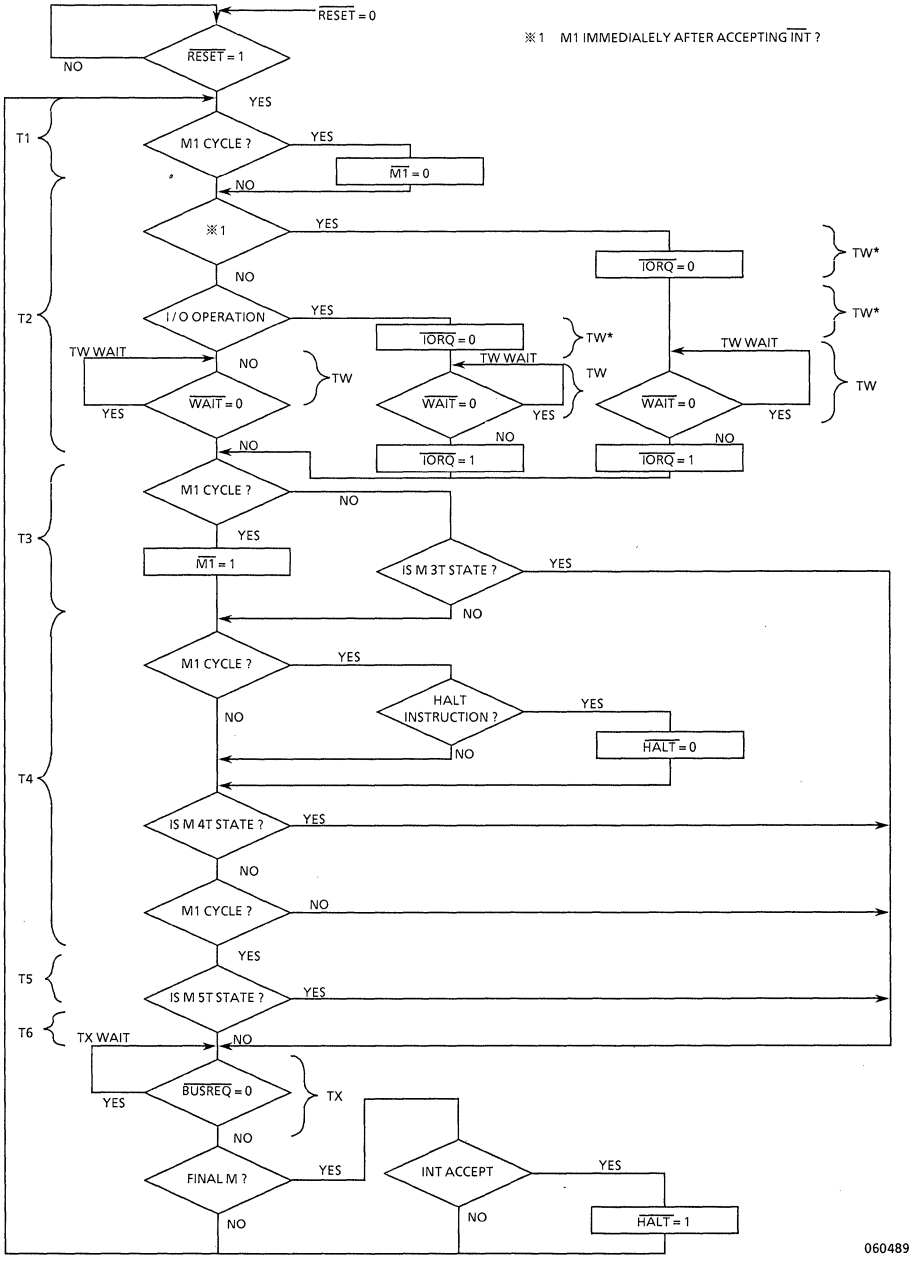


Figure 3.10 Status Transition Diagram

[3] Basic Timing

(1) Opcode fetch cycle (M1)

In the Opcode fetch cycle, MPU fetches an Opcode in the machine-language codes in memory. This is also called the M1 cycle because it is the first machine cycle to execute each instruction.

Figure 3.11 shows the basic timing of a basic Opcode fetch cycle.

In clock state T1, the content of the program counter is put on the address bus. The $\overline{M1}$ signal goes "0", indicating to the MPU that this is the Opcode fetch cycle. At the same time, $\overline{MREQ}$ and $\overline{RD}$ signals go "0". When the $\overline{MREQ}$ signal goes "0", the address signal has already been stabilized. Therefore, this signal can be used for the memory chip enable signal. The $\overline{RD}$ signal indicates that the MPU is ready to accept the data from memory. By these signals, the MPU accesses memory to fetch the Opcode in the instruction register. The MPU samples the $\overline{WAIT}$ signal on the falling edge of clock state T2. If the $\overline{WAIT}$ signal is "0" on the falling edge of clock state T2 and the following wait state (TW), the next state becomes clock state TW. Figure 3.12 shows the delay state of the Opcode fetch cycle caused by the $\overline{WAIT}$ signal.

The data (Opcode) on the data bus is fetched on the rising edge of clock state T3 then, the $\overline{MREQ}$, $\overline{RD}$, and $\overline{M1}$ signals go "1". In clock state T3, a memory refresh address is put on the low-order 8 bits of the address bus and the $\overline{RFSH}$ signal goes "0" and the $\overline{MREQ}$ signal goes "0" again. This signal indicates that the memory refresh cycle is on. At this time, the contents of the I register are put on the high-order 8 bits of the address bus and the 8 bits of the R register are put on the low-order 8 bits of the address bus. By using the $\overline{RFSH}$ and $\overline{MREQ}$ signals, memory refresh is performed in clock state T3 and T4. However, the $\overline{RD}$ signal remains "1" because the contents of the memory refresh address are not put on the data bus.

In clock state T4, the $\overline{MREQ}$ signal returns to "1". The refresh address is kept output until the rising edge of the clock state T1 in the next machine cycle, keeping the $\overline{RFSH}$ signal set to "0". The cycle delay state caused by setting the $\overline{WAIT}$ signal to "0" is the same in the memory read/write, input/output, and maskable interrupt acknowledge cycles. The diagram of the cycle delay state caused by the $\overline{WAIT}$ signal set to "0" is omitted in the following description.

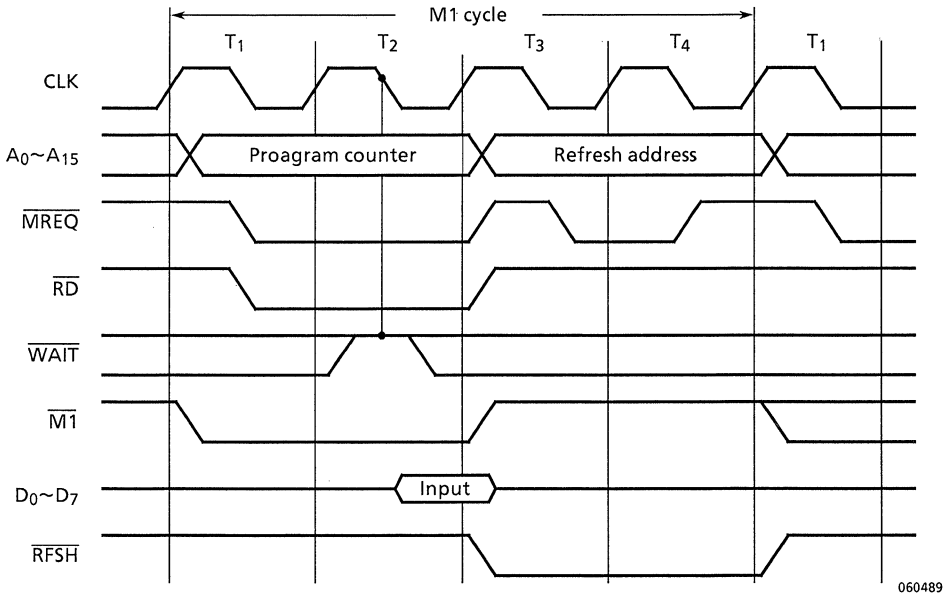


Figure 3.11 Opcode Fetch Timing

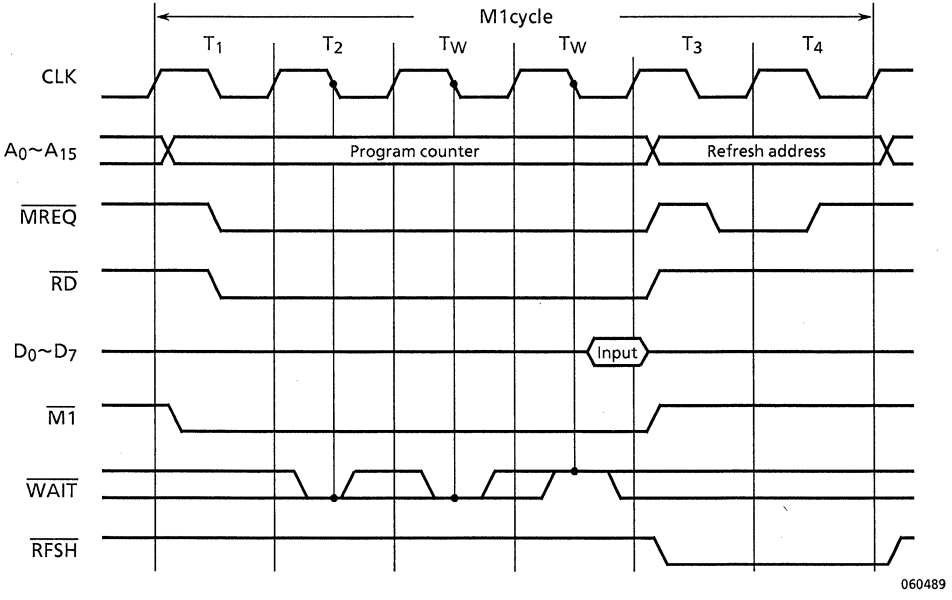


Figure 3.12 Opcode Fetch Timing Incliding Wait State

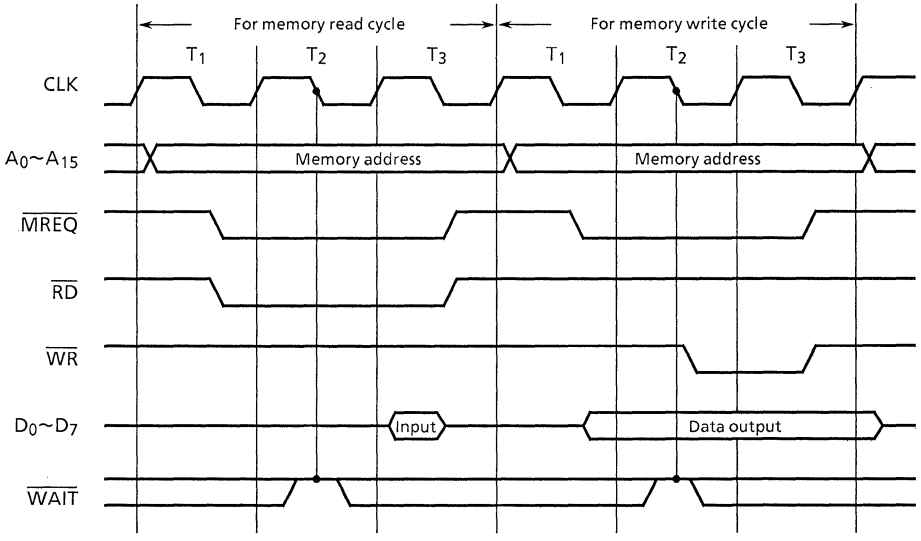
(2) Memory read/write operations

Figure 3.13 shows the basic timing of memory read/write operations (except for the Opcode fetch cycle) in the same diagram for convenience.

In each operation, the memory address signal to read/write data on the address bus is output in clock clock state T1. The operation in which the $\overline{WAIT}$ signal is sampled in clock state T2 and the following TW state is the same as the Opcode fetch cycle.

In memory read, memory data is put on the data bus by the address, $\overline{MREQ}$, and $\overline{RD}$ signals. The MPU reads this data.

In memory write, the memory address signal is put on the address bus then the $\overline{MREQ}$ signal is set to "0" to put the write data onto the data bus. When the data bus has been stabilized, the $\overline{WR}$ signal is output in clock state T2. The WR signal can be used as the memory write signal.



060489

Figure 3.13 Memory Read/Write Cycle Timing

(3) Input/output operations

Figure 3.14 shows the basic timing of input/output operations. The feature of the I/O operation timing is that, regardless of the state of the $\overline{WAIT}$ signal in clock state T2, the I/O cycle automatically goes in the wait state (TW*) after clock T2. The $\overline{WAIT}$ signal is sampled on the falling edge of TW*. If the $\overline{WAIT}$ signal is "0" on the falling edges of TW* and the following clock state, the I/O operation enters into clock state TW*. Clock state TW* is inserted because the $\overline{IORQ}$ signal goes "0" in clock state T2, so that it is too late to sample the $\overline{WAIT}$ signal after decoding the I/O port address. In each of input and output operations, the I/O port address is put on the low-order 8 bits of the address bus in clock state T1. On the high-order 8 bits, the contents of the accumulator or B register are output. In clock state T2, the $\overline{IORQ}$ signal goes "0" instead of the $\overline{MREQ}$ signal. The $\overline{IORQ}$ signal can be used as the chip enable signal for a peripheral LSI.

In an input operation, the contents of the input port are read onto the data bus by the address, $\overline{IORQ}$, or $\overline{RD}$ signals. The MPU reads this data.

In an output operation, the output port address and the output data are respectively put on the address bus and data bus in clock state Tl, then the $\overline{IORQ}$ and $\overline{WR}$ signals go "0" in clock state T2. The $\overline{WR}$ signal can be used as the output port write signal.

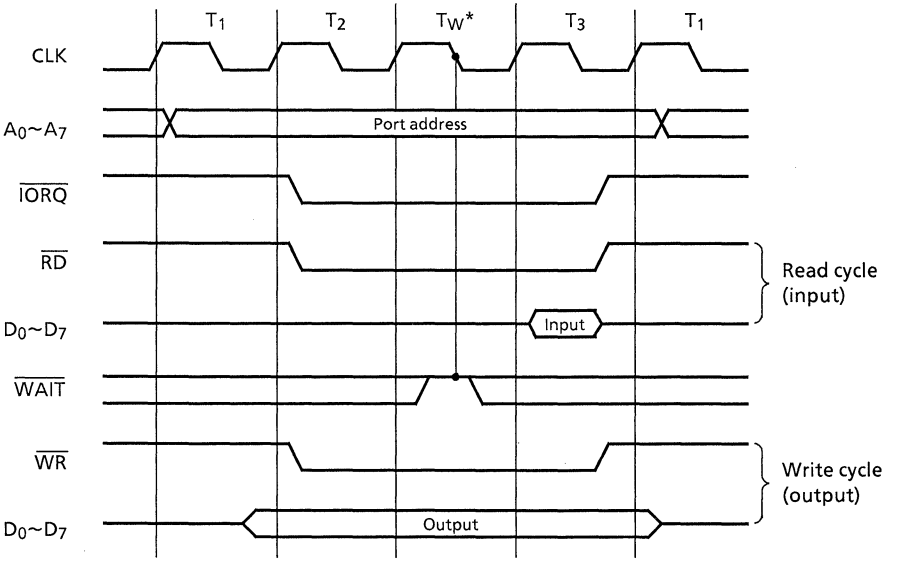


Figure 3.14 I/O Operating Timing

(4) Bus request and bus acknowledge operations

Figure 3.15 shows the basic timing of bus request and bus acknowledge operations.

The address bus (A0 through A15), data bus (D0 through D7), $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ signals controlled by the MPU can be put in the high-impedance state (floating) to electrically disconnect them from the MPU. This operation, after sampling the $\overline{BUSREQ}$ signal on the rising edge of the last clock of each machine cycle, starts on the rising edge of the next clock if this signal is found "0".

Subsequently, these buses are controlled by external peripheral LSIs. For example, data can be directly transferred between memory and these peripheral LSIs. This state is cleared if the $\overline{BUSREQ}$ signal is found "1" after sampling it on the rising edge of each subsequent clock state (TX), and enters into the next machine cycle. During the floating state, the $\overline{BUSACK}$ signal goes "0" to indicate it to the peripheral LSIs.

In this state, however, no memory refresh is performed and, therefore, the $\overline{RFSH}$ signal is set to "1". Hence, to maintain this state for a long time with a system using dynamic memory, memory refresh must be performed by the external controller.

Note that, in the floating state, neither maskable interrupt (INT) nor non-maskable interrupt (NMI) can be accepted.

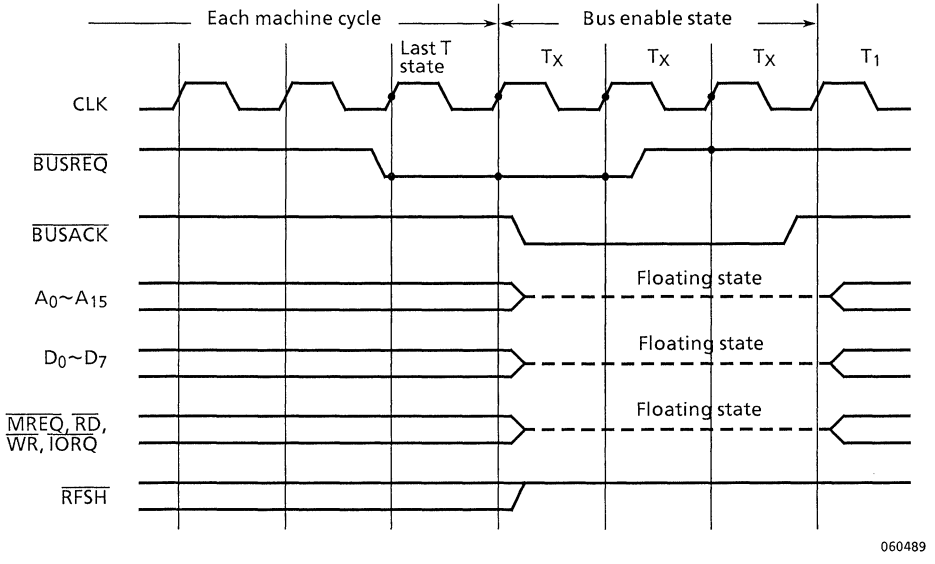


Figure 3.15 Bus Request and Bus Acknowledge Timing

(5) Maskable interrupt acknowledge operation

Figure 3.16 shows the basic timing of the maskable interrupt acknowledge.

The MPU samples the maskable interrupt request signal ($\overline{INT}$) on the rising edge of the last clock of each instruction execution. If the $\overline{INT}$ signal is found "0", a maskable interrupt is accepted except in the following cases:

- The interrupt enable flip-flop is reset to "0".
- The $\overline{BUSREQ}$ signal is "0".

When a maskable interrupt has been accepted, a special Opcode fetch cycle is generated. In this cycle, 2 clock states of wait state (TW*) is automatically inserted after the clock state T2. The $\overline{WAIT}$ signal is sampled on the falling edges of the second clock state TW* and the following clock state TW and, if the $\overline{WAIT}$ signal is found "0", the instruction cycle enters in the next clock state TW. In this Opcode fetch cycle, the $\overline{IORQ}$ signal goes "0" in the first TW* state instead of the $\overline{MREQ}$ signal while, in a normal Opcode fetch cycle, the $\overline{MREQ}$ signal goes "0" in clock state T1. This indicates to the maskable interrupt requesting LSI that the 8-bit interrupt vector can be put on the data bus. The MPU reads this data to perform interrupt processing. Therefore, the contents of the program counter put on the address bus are not used. Unlike an ordinary I/O operation, the $\overline{RD}$ signal does not go "0".

In clock state T3, the memory refresh address signal is put on the address bus for memory refresh like normal Opcode fetch cycle and the $\overline{RFSH}$ signal goes "0". In the subsequent machine cycles (M2 and M3) , the contents of the current program counter are saved into the stack. In machine cycles M4 and M5, the contents of the I register (the high-order 8 bits) and the contents of the address indicated by the address of the vector (the low-order 8 bits) from the peripheral LSI are fetched in the program counter.

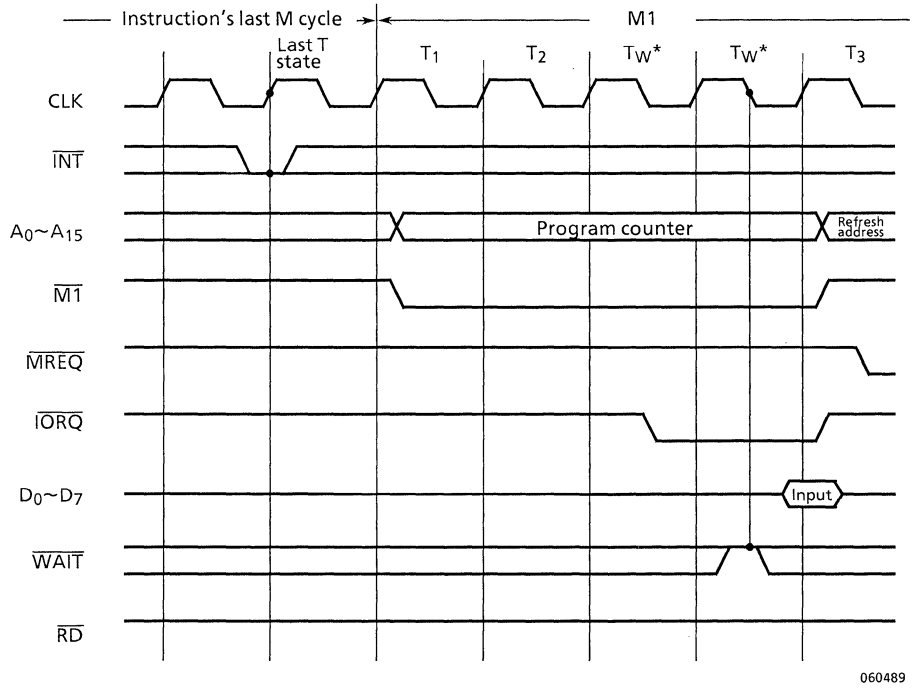


Figure 3.16 Maskable Interrupt Acknowledge Timing

(6) Non-maskable interrupt acknowledge operation

Figure 3.17 shows the basic timing of non-maskable interrupt acknowledge.

When the non-maskable interrupt request signal ($\overline{NMI}$) goes low, the internal non-maskable flip-flop is set to "1". The $\overline{NMI}$ signal is detected in any timing of each instruction. However, the internal NMI flip-flop is sampled on the rising edge of the last clock of each instruction. Therefore, the $\overline{NMI}$ signal should go low by the last clock state of an instruction.

The Opcode fetch cycle for non-maskable interrupt request acknowledge is generally the same as the ordinary Opcode fetch cycle. However, the Opcode on the data bus at the time is ignored. The contents of the current program counter are saved into the stack in the subsequent machine cycles (M2 and M3) . In the following machine cycle, the operation jumps to address 0066H, the non-maskable interrupt vector address. The machine cycles after these depend on the contents of the fetched Opcode.

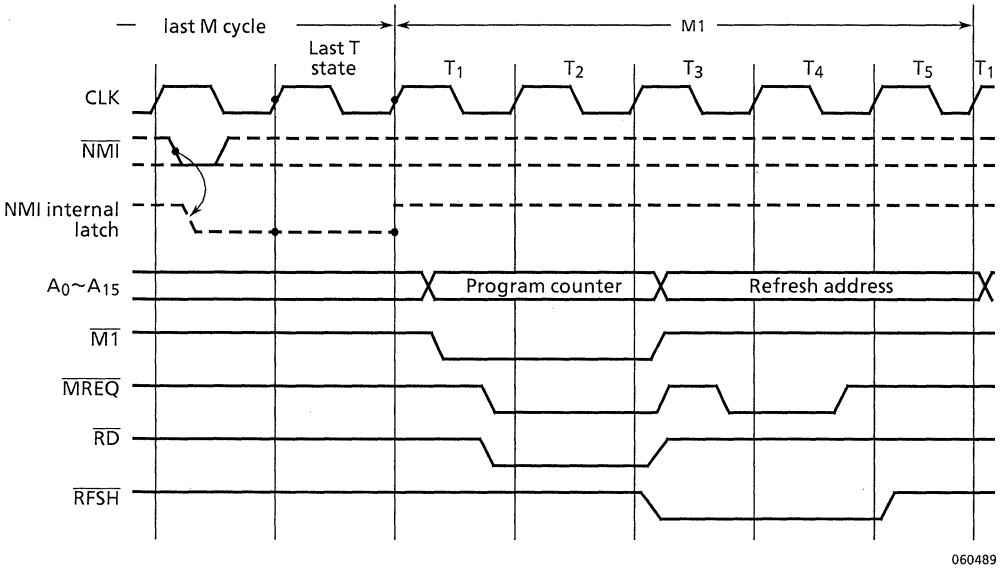


Figure 3.17 Non-Maaskable Inpterrupt Acknowledge Timing

(7) Halt operation

When a HALT instruction is fetched in the Opcode fetch cycle, the MPU sets the $\overline{\text{HALT}}$ signal to "0" synchronized with the falling edge of clock state T4 to indicate it to the peripheral LSI and stops operating. If the system clock is kept supplied in the halt state, the MPU continues executing NOP instructions. This is done to output refresh signals when the dynamic memory is used. The NOP instruction execution cycle is the same as the ordinary Opcode fetch cycle except the data on the data bus are ignored.

The halt state is cleared when an interrupt is accepted or the $\overline{\text{RESET}}$ signal is set to "0" to reset the MPU. Figure 3.18 shows the halt state clear operation by interrupt acknowledge. An interrupt is sampled on the rising edge of the last clock (clock state T4) of the NOP instruction. A maskable interrupt can be accepted when the $\overline{\text{INT}}$ signal is "0". A non-maskable interrupt is accepted when the internal NMI flip-flop which is set on the falling edge of the $\overline{\text{NMI}}$ signal is set at "1". However, it is required that the interrupt enable flip-flop is set to "1" for a maskable interrupt to be accepted. The interrupt processing for the accepted interrupt starts from the next cycle.

However, when the supply of the system clock has been stopped by the power down operation, it is required to restart the supply of the system clock and input the $\overline{INT}$ signal until the execution of one instruction is completed or the $\overline{RESET}$ signal until 3 clocks are input. Figure 3.19 shows the timing of clearing the halt state caused by power down. By using TLCS-Z80's clock generator/controller (TMPZ84C60P or TMPZ84C61AP), above-stated operation is realized easily.

For the reset operation, see (8) "Reset operation". Note that the $\overline{INT}$ and $\overline{NMI}$ signals are shown on the same diagram in Figures 3.18 and 3.19 for convenience.

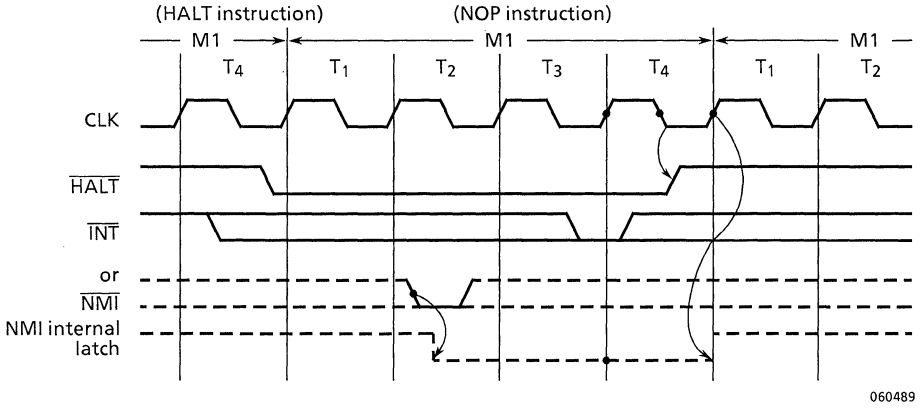


Figure 3.18 Timing of Clearing Halt State Caused by Interrupt Acknowledge

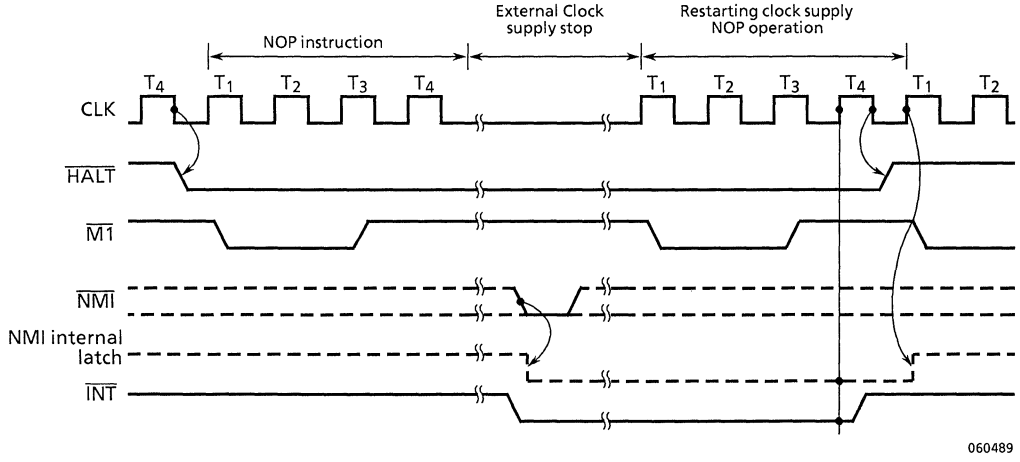


Figure 3.19 Timing of Clearing Halt State Caused By Power Down

(8) Reset operation

Figure 3.20 shows the basic timing of reset operation.

To reset the MPU, the $\overline{RESET}$ signal must be kept at "0" for at least 3 clocks. When the $\overline{RESET}$ signal goes "1", instruction execution starts from address 0000H after a dummy cycle of at least 2 clock states.

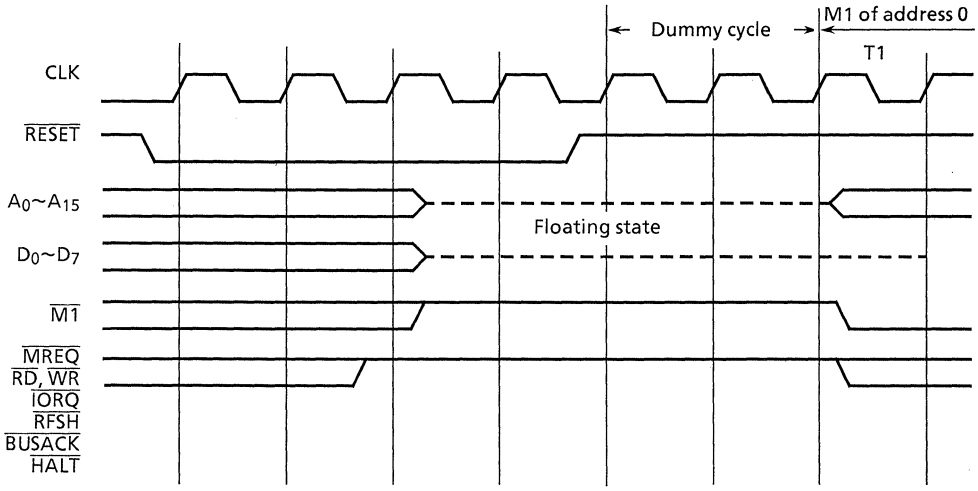


Figure 3.20 Reset Timing

3.4 TMPZ84C00A INSTRUCTION SET

This subsection lists the TMPZ84C00A instruction codes and their functions. The table below lists the symbols and abbreviations used to describe the instruction set. The symbols which require special attention are described in the locations in which they appear.

- Symbols (1/2)

Classification	Symbol	Meaning
Register	r, g	Register B, C, D, E, H, L, A,
	t	Register pair BC, DE, HL
		Stack pointer SP
	q	Register pair BC, DE, HL, AF
	p	Register pair BC, DE
		Index register IX
		Stack pointer SP
	s	Register pair BC, DE
		Index register IY
		Stack pointer SP
	t_H	Higher register of register pair (B, D, H)
		Higher 8 bits of stack pointer (SP)
	q_H	Higher register of register pair (B, D, H, A)
	IX_H	Higher 8 bits of index register IX
	IY_H	Higher 8 bits of index register IY
	PC_H	Higher 8 bits of program counter (PC)
	t_L	Lower register of register pair (C, E, L)
		Lower 8 bits of stack pointer (SP)
	q_L	Lower register of register pair (C, E, L, F)
	IX_L	Lower 8 bits of index register IX
	IY_L	Lower 8 bits of index register IY
	PC_L	Lower 8 bits of program counter (PC)
	rb	Bit b (0-7) of register (B, C, D, E, H, L, A)

120489

- Symbols (2/2)

Classification	Symbol	Meaning
Memory	mn	Memory address represented in 16 bits. m indicates higher 8 bits and n, lower 8 bits.
	$(HL)_b$	Bit b (0-7) of the contents of the memory address indicated by register pair HL.
	$(IX+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IX.
	$(IY+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IY.
Flag change symbol	0	Reset to "0" by operation.
	1	Set to "1" by operation.
	-	No change
	*	Affected by operation
	X	Undefined
	P	Handled as parity flag. P = 0: odd parity P = 1: even parity
	V	Handled as overflow flag. V = 0: No overflow V = 1: Overflow
Operator	←	Transfer
	↔	Exchange
	+	Add
	−	Subtract
	∧	Logical and between bits.
	∨	Logical or between bits.
	⊕	Exclusive or between bits
Others	IFF	Interrupt enable flip-flop
	CY	Carry flag
	Z	Zero flag

120489

TMPZ84C00A Instruction Set (1/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT DATA LOAD	LD r,g	01 rrr ggg	40+r×8+g	r←g	–	–	X	–	X	–	–	–	1	4
	LD r,n	00 rrr 110 nn nnn nnn	06+r×8 n	r←n	–	–	X	–	X	–	–	–	2	7
	LD r,(HL)	01 rrr 110	46+r×8	r←(HL)	–	–	X	–	X	–	–	–	2	7
	LD r,(IX+d)	11 011 101 01 rrr 110 dd ddd ddd	DD 46+r×8 d	r←(IX+d)	–	–	X	–	X	–	–	–	5	19
	LD r,(IY+d)	11 111 101 01 rrr 110 dd ddd ddd	FD 46+r×8 d	r←(IY+d)	–	–	X	–	X	–	–	–	5	19
	LD (HL),r	01 110 rrr	70+r	(HL)←r	–	–	X	–	X	–	–	–	2	7
	LD (IX+d),r	11 011 101 01 110 rrr dd ddd ddd	DD 70+r d	(IX+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),r	11 111 101 01 110 rrr dd ddd ddd	FD 70+r d	(IY+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (HL),n	00 110 110 nn nnn nnn	36 n	(HL)←n	–	–	X	–	X	–	–	–	3	10
	LD (IX+d), n	11 011 101 00 110 110 dd ddd ddd nn nnn nnn	DD 36 d n	(IX+d)←n	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),n	11 111 101 00 110 110 dd ddd ddd nn nnn nnn	FD 36 d n	(IY+d)←n	–	–	X	–	X	–	–	–	5	19
	LD A,(BC)	00 001 010	0A	A←(BC)	–	–	X	–	X	–	–	–	2	7
	LD A,(DE)	00 011 010	1A	A←(DE)	–	–	X	–	X	–	–	–	2	7
	LD A,(mn)	00 111 010 nn nnn nnn mm mmm mmm	3A n m	A←(mn)	–	–	X	–	X	–	–	–	4	13
	LD (BC),A	00 000 010	02	(BC)←A	–	–	X	–	X	–	–	–	2	7
	LD (DE),A	00 010 010	12	(DE)←A	–	–	X	–	X	–	–	–	2	7
	LD (mn),A	00 110 010 nn nnn nnn mm mmm mmm	32 n m	(mn)←A	–	–	X	–	X	–	–	–	4	13
	LD A,I	11 101 101 01 010 111	ED 57	A←I	*	*	X	0	X	IFF	0	–	2	9
	LD A,R	11 101 101 01 011 111	ED 5F	A←R	*	*	X	0	X	IFF	0	–	2	9
	LD I,A	11 101 101 01 000 111	ED 47	I←A	–	–	X	–	X	–	–	–	2	9
	LD R,A	11 101 101 01 001 111	ED 4F	R←A	–	–	X	–	X	–	–	–	2	9
16-BIT DATA LOAD	LD t,mn	00 tt0 001 nn nnn nnn mm mmm mmm	01+t×10 n m	t←mn	–	–	X	–	X	–	–	–	3	10
	LD IX,mn	11 011 101 00 100 001 nn nnn nnn mm mmm mmm	DD 21 n m	IX←mn	–	–	X	–	X	–	–	–	4	14

r	rrr
g	ggg
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

Note : r,g means any of the registers A, B, C, D, E, H, L.
IFF in "Flag" column indicates that the content of the interrupt enable flip-flop is copied into the P/V flag.

120489

TMPZ84C00A Instruction Set (2/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
16-BIT DATA LOAD	LD IY,mn	11 111 101 00 100 001 nn nnn nnn mm mmm mmm	FD 21 n m	IY←mn	–	–	X	–	X	–	–	–	4	14
	LD HL,(mn)	00 101 010 nn nnn nnn mm mmm mmm	2A n m	H←(mn+1) L←(mn)	–	–	X	–	X	–	–	–	5	16
	LD t,(mn)	11 101 101 01 tt1 011 nn nnn nnn mm mmm mmm	ED 4B+t×10 n m	t_H←(mn+1) t_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IX,(mn)	11 011 101 00 101 010 nn nnn nnn mm mmm mmm	DD 2A n m	IX_H←(mn+1) IX_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IY,(mn)	11 111 101 00 101 010 nn nnn nnn mm mmm mmm	FD 2A n m	IY_H←(mn+1) IY_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD (mn),HL	00 100 010 nn nnn nnn mm mmm mmm	22 n m	(mn+1)←H (mn)←L	–	–	X	–	X	–	–	–	5	16
	LD (mn),t	11 101 101 01 tt0 011 nn nnn nnn mm mmm mmm	ED 43+t×10 n m	(mn+1)←t_H (mn)←t_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IX	11 011 101 00 100 010 nn nnn nnn mm mmm mmm	DD 22 n m	(mn+1)←IX_H (mn)←IX_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IY	11 111 101 00 100 010 nn nnn nnn mm mmm mmm	FD 22 n m	(mn+1)←IY_H (mn)←IY_L	–	–	X	–	X	–	–	–	6	20
	LD SP,HL	11 111 001	F9	SP←HL	–	–	X	–	X	–	–	–	1	6
	LD SP,IX	11 011 101 11 111 001	DD F9	SP←IX	–	–	X	–	X	–	–	–	2	10
	LD SP,IY	11 111 101 11 111 001	FD F9	SP←IY	–	–	X	–	X	–	–	–	2	10
	PUSH q	11 qq0 101	C5+q×10	(SP−2)←q_L,(SP−1)←q_H, SP←SP−2	–	–	X	–	X	–	–	–	3	11
	PUSH IX	11 011 101 11 100 101	DD E5	(SP−2)←IX_L,(SP−1)←IX_H SP←SP−2	–	–	X	–	X	–	–	–	4	15
	PUSH IY	11 111 101 11 100 101	FD E5	(SP−2)←IY_L,(SP−1)←IY_H SP←SP−2	–	–	X	–	X	–	–	–	4	15
	POP q	11 qq0 001	C1+q×10	q_H←(SP+1),q_L←(SP), SP←SP+2	–	–	X	–	X	–	–	–	3	10
	POP IX	11 011 101 11 100 001	DD E1	IX_H←(SP+1),IX_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
	POP IY	11 111 101 11 100 001	FD E1	IY_H←(SP+1),IY_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
*1	EX DE,HL	11 101 011	EB	DE↔HL	–	–	X	–	X	–	–	–	1	4
	EX AF,AF'	00 001 000	08	AF↔AF'	–	–	X	–	X	–	–	–	1	4
	EXX	11 011 001	D9	BC↔BC',DE↔DE',HL↔HL'	–	–	X	–	X	–	–	–	1	4

t	tt
BC	00
DE	01
HL	10
SP	11

q	qq
BC	00
DE	01
HL	10
AF	11

Note : t is any of the register pairs BC, DE, HL, SP.
q is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively. (Ex) $BC_L = C$, $AF_H = A$.

*1 : EXCHANGE

120489

TMPZ84C00A Instruction Set (3/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES	
EXCHANGE	EX (SP),HL	11 100 011	E3	H⇔(SP+1),L⇔(SP)	–	–	X	–	X	–	–	–	5	19	
	EX (SP),IX	11 011 101 11 100 011	DD E3	IX_H⇔(SP+1) IX_L⇔(SP)	–	–	X	–	X	–	–	–	6	23	
	EX (SP),IY	11 111 101 11 100 011	FD E3	IY_H⇔(SP+1) IY_L⇔(SP)	–	–	X	–	X	–	–	–	6	23	
BLOCK TRANSFER BLOCK SEARCH	LDI	11 101 101 10 100 000	ED A0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDIR	11 101 101 10 110 000	ED B0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	LDD	11 101 101 10 101 000	ED A8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDDR	11 101 101 10 111 000	ED B8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	CPI	11 101 101 10 100 001	ED A1	A−(HL) HL←HL+1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPIR	11 101 101 10 110 001	ED B1	A−(HL),HL←HL+1,BC←BC−1 Repeat until A=(HL) or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
	CPD	11 101 101 10 101 001	ED A9	A−(HL) HL←HL−1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPDR	11 101 101 10 111 001	ED B9	A−(HL),HL←HL−1,BC←BC−1 Repeat until A=(HL) or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
8-BIT ARITHMETIC AND LOGICAL	ADD A,r	10 000 rrr	80+r	A←A+r	*	*	X	*	X	V	0	*	1	4	
	ADD A,n	11 000 110 nn nnn nnn	C6 n	A←A+n	*	*	X	*	X	V	0	*	2	7	
	ADD A,(HL)	10 000 110	86	A←A+(HL)	*	*	X	*	X	V	0	*	2	7	
	ADD A,(IX+d)	11 011 101 10 000 110 dd ddd ddd	DD 86 d	A←A+(IX+d)	*	*	X	*	X	V	0	*	5	19	
	ADD A,(IY+d)	11 111 101 10 000 110 dd ddd ddd	FD 86 d	A←A+(IY+d)	*	*	X	*	X	V	0	*	5	19	
	ADC A,r	10 001 rrr	88+r	A←A+r+CY	*	*	X	*	X	V	0	*	1	4	
	ADC A,n	11 001 110 nn nnn nnn	CE n	A←A+n+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(HL)	10 001 110	8E	A←A+(HL)+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(IX+d)	11 011 101 10 001 110 dd ddd ddd	DD 8E d	A←A+(IX+d)+CY	*	*	X	*	X	V	0	*	5	19	
	ADC A,(IY+d)	11 111 101 10 001 110 dd ddd ddd	FD 8E d	A←A+(IY+d)+CY	*	*	X	*	X	V	0	*	5	19	
	SUB r	10 010 rrr	90+r	A←A−r	*	*	X	*	X	V	1	*	1	4	
	SUB n	11 010 110 nn nnn nnn	D6 n	A←A−n	*	*	X	*	X	V	1	*	2	7	
	SUB (HL)	10 010 110	96	A←A−(HL)	*	*	X	*	X	V	1	*	2	7	
	SUB (IX+d)	11 011 101 10 010 110 dd ddd ddd	DD 96 d	A←A−(IX+d)	*	*	X	*	X	V	1	*	5	19	
	SUB (IY+d)	11 111 101 10 010 110 dd ddd ddd	FD 96 d	A←A−(IY+d)	*	*	X	*	X	V	1	*	5	19	

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : *M P/V flag is 0 if the result of BC − 1 = 0, otherwise P/V = 1.
*N Z flag is 1 if A = (HL), otherwise Z = 0.
[ ] indicates the total condition of the number of cycles and states indicated by arrow.
r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C00A Instruction Set (4/9)

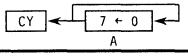
ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT ARITHMETIC AND LOGICAL	SBC A,r	10 011 rrr	98+r	A←A−r−CY	*	*	X	*	X	V	1	*	1	4
	SBC A,n	11 011 110 nn nnn nnn	DE n	A←A−n−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(HL)	10 011 110	9E	A←A−(HL)−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(IX+d)	11 011 101 10 011 110 dd ddd ddd	DD 9E d	A←A−(IX+d)−CY	*	*	X	*	X	V	1	*	5	19
	SBC A,(IY+d)	11 111 101 10 011 110 dd ddd ddd	FD 9E d	A←A−(IY+d)−CY	*	*	X	*	X	V	1	*	5	19
	AND r	10 100 rrr	A0+r	A←A∧r	*	*	X	1	X	P	0	0	1	4
	AND n	11 100 110 nn nnn nnn	E6 n	A←A∧n	*	*	X	1	X	P	0	0	2	7
	AND (HL)	10 100 110	A6	A←A∧(HL)	*	*	X	1	X	P	0	0	2	7
	AND (IX+d)	11 011 101 10 100 110 dd ddd ddd	DD A6 d	A←A∧(IX+d)	*	*	X	1	X	P	0	0	5	19
	AND (IY+d)	11 111 101 10 100 110 dd ddd ddd	FD A6 d	A←A∧(IY+d)	*	*	X	1	X	P	0	0	5	19
	OR r	10 110 rrr	B0+r	A←A∨r	*	*	X	0	X	P	0	0	1	4
	OR n	11 110 110 nn nnn nnn	F6 n	A←A∨n	*	*	X	0	X	P	0	0	2	7
	OR (HL)	10 110 110	B6	A←A∨(HL)	*	*	X	0	X	P	0	0	2	7
	OR (IX+d)	11 011 101 10 110 110 dd ddd ddd	DD B6 d	A←A∨(IX+d)	*	*	X	0	X	P	0	0	5	19
	OR (IY+d)	11 111 101 10 110 110 dd ddd ddd	FD B6 d	A←A∨(IY+d)	*	*	X	0	X	P	0	0	5	19
	XOR r	10 101 rrr	A8+r	A←A∀r	*	*	X	0	X	P	0	0	1	4
	XOR n	11 101 110 nn nnn nnn	EE n	A←A∀n	*	*	X	0	X	P	0	0	2	7
	XOR (HL)	10 101 110	AE	A←A∀(HL)	*	*	X	0	X	P	0	0	2	7
	XOR (IX+d)	11 011 101 10 101 110 dd ddd ddd	DD AE d	A←A∀(IX+d)	*	*	X	0	X	P	0	0	5	19
	XOR (IY+d)	11 111 101 10 101 110 dd ddd ddd	FD AE d	A←A∀(IY+d)	*	*	X	0	X	P	0	0	5	19
	CP r	10 111 rrr	B8+r	A−r	*	*	X	*	X	V	1	*	1	4
	CP n	11 111 110 nn nnn nnn	FE n	A−n	*	*	X	*	X	V	1	*	2	7
	CP (HL)	10 111 110	BE	A−(HL)	*	*	X	*	X	V	1	*	2	7
	CP (IX+d)	11 011 101 10 111 110 dd ddd ddd	DD BE d	A−(IX+d)	*	*	X	*	X	V	1	*	5	19
	CP (IY+d)	11 111 101 10 111 110 dd ddd ddd	FD BE d	A−(IY+d)	*	*	X	*	X	V	1	*	5	19
	INC r	00 rrr 100	04+r×8	r←r+1	*	*	X	*	X	V	0	−	1	4
	INC (HL)	00 110 100	34	(HL)←(HL)+1	*	*	X	*	X	V	0	−	3	11
	INC (IX+d)	11 011 101 00 110 100 dd ddd ddd	DD 34 d	(IX+d)←(IX+d)+1	*	*	X	*	X	V	0	−	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C00A Instruction Set (5/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT ARITHMETIC AND LOGICAL	INC (IY+d)	11 111 101 00 110 100 dd ddd ddd	FD 34 d	(IY+d)←(IY+d)+1	*	*	X	*	X	V	0	–	6	23
	DEC r	00 rrr 101	05+r×8	r←r−1	*	*	X	*	X	V	1	–	1	4
	DEC (HL)	00 110 101	35	(HL)←(HL)−1	*	*	X	*	X	V	1	–	3	11
	DEC (IX+d)	11 011 101 00 110 101 dd ddd ddd	DD 35 d	(IX+d)←(IX+d)−1	*	*	X	*	X	V	1	–	6	23
	DEC (IY+d)	11 111 101 00 110 101 dd ddd ddd	FD 35 d	(IY+d)←(IY+d)−1	*	*	X	*	X	V	1	–	6	23
GENELAL-PURPOSE ARITHMETIC AND MPU CONTROL	DAA	00 100 111	27	Decimal adjust accumulator	*	*	X	*	X	P	–	*	1	4
	CPL	00 101 111	2F	A←$\overline{A}$	–	–	X	1	X	–	1	–	1	4
	NEG	11 101 101 01 000 100	ED 44	A←0−A	*	*	X	*	X	V	1	*	2	8
	CCF	00 111 111	3F	CY←$\overline{CY}$	–	–	X	X	X	–	0	*	1	4
	SCF	00 110 111	37	CY←1	–	–	X	0	X	–	0	1	1	4
	NOP	00 000 000	00	no operation	–	–	X	–	X	–	–	–	1	4
	HALT	01 110 110	76	MPU Halted	–	–	X	–	X	–	–	–	1	4
	DI	11 110 011	F3	IFF←0	–	–	X	–	X	–	–	–	1	4
	EI	11 111 011	FB	IFF←1	–	–	X	–	X	–	–	–	1	4
	IM 0	11 101 101 01 000 110	ED 46	Set interrupt mode 0	–	–	X	–	X	–	–	–	2	8
	IM 1	11 101 101 01 010 110	ED 56	Set interrupt mode 1	–	–	X	–	X	–	–	–	2	8
	IM 2	11 101 101 01 011 110	ED 5E	Set interrupt mode 2	–	–	X	–	X	–	–	–	2	8
16-BIT ARITHMETIC	ADD HL,t	00 tt1 001	09+t×10	HL←HL+t	–	–	X	X	X	–	0	*	3	11
	ADC HL,t	11 101 101 01 tt1 010	ED 4A+t×10	HL←HL+t+CY	*	*	X	X	X	V	0	*	4	15
	SBC HL,t	11 101 101 01 tt0 010	ED 42+t×10	HL←HL−t−CY	*	*	X	X	X	V	1	*	4	15
	ADD IX,p	11 011 101 00 pp1 001	DD 09+p×10	IX←IX+p	–	–	X	X	X	–	0	*	4	15
	ADD IY,s	11 111 101 00 ss1 001	FD 09+s×10	IY←IY+s	–	–	X	X	X	–	0	*	4	15
	INC t	00 tt0 011	03+t×10	t←t+1	–	–	X	–	X	–	–	–	1	6
	INC IX	11 011 101 00 100 011	DD 23	IX←IX+1	–	–	X	–	X	–	–	–	2	10
	INC IY	11 111 101 00 100 011	FD 23	IY←IY+1	–	–	X	–	X	–	–	–	2	10
	DEC t	00 tt1 011	0B+t×10	t←t−1	–	–	X	–	X	–	–	–	1	6
	DEC IX	11 011 101 00 101 011	DD 2B	IX←IX−1	–	–	X	–	X	–	–	–	2	10
	DEC IY	11 111 101 00 101 011	FD 2B	IY←IY−1	–	–	X	–	X	–	–	–	2	10
ROTATE	RLCA	00 000 111	07		–	–	X	0	X	–	0	*	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

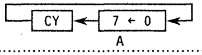
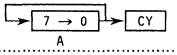
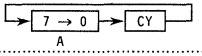
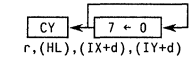
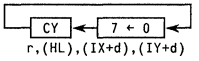
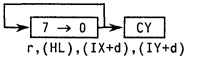
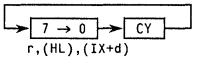
p	pp
BC	00
DE	01
IX	10
SP	11

s	ss
BC	00
DE	01
IY	10
SP	11

Note : ss is any of the register pairs BC, DE, HL, SP. PP is any of the register pairs BC, DE, IX, SP.
rr is any of the register pairs BC, DE, IY, SP.

120489

TMPZ84C00A Instruction Set (6/9)

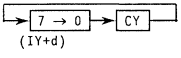
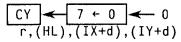
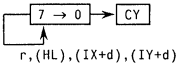
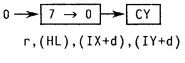
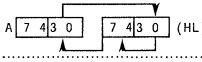
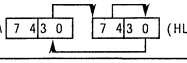
ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RLA	00 010 111	17		–	–	X	0	X	–	0	*	1	4
	RRCA	00 001 111	0F		–	–	X	0	X	–	0	*	1	4
	RRA	00 011 111	1F		–	–	X	0	X	–	0	*	1	4
	RLC r	11 001 011 00 000 rrr	CB 00+r		*	*	X	0	X	P	0	*	2	8
	RLC (HL)	11 001 011 00 000 110	CB 06		*	*	X	0	X	P	0	*	4	15
	RLC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 000 110	DD CB d 06	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	RLC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 000 110	FD CB d 06		*	*	X	0	X	P	0	*	6	23
	RL r	11 001 011 00 010 rrr	CB 10+r		*	*	X	0	X	P	0	*	2	8
	RL (HL)	11 001 011 00 010 110	CB 16		*	*	X	0	X	P	0	*	4	15
	RL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 010 110	DD CB d 16	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	RL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 010 110	FD CB d 16		*	*	X	0	X	P	0	*	6	23
	RRC r	11 001 011 00 001 rrr	CB 08+r		*	*	X	0	X	P	0	*	2	8
	RRC (HL)	11 001 011 00 001 110	CB 0E		*	*	X	0	X	P	0	*	4	15
	RRC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 001 110	DD CB d 0E	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	RRC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 001 110	FD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RR r	11 001 011 00 011 rrr	CB 18+r		*	*	X	0	X	P	0	*	2	8
	RR (HL)	11 001 011 00 011 110	CB 1E		*	*	X	0	X	P	0	*	4	15
	RR (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 011 110	DD CB d 1E	r,(HL),(IX+d)	*	*	X	0	X	P	0	*	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C00A Instruction Set (7/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RR (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 011 110	FD CB d 1E		*	*	X	0	X	P	0	*	6	23
	SLA r	11 001 011 00 100 rrr	CB 20+r		*	*	X	0	X	P	0	*	2	8
	SLA (HL)	11 001 011 00 100 110	CB 26		*	*	X	0	X	P	0	*	4	15
	SLA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 100 110	DD CB d 26		*	*	X	0	X	P	0	*	6	23
	SLA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 100 110	FD CB d 26		*	*	X	0	X	P	0	*	6	23
	SRA r	11 001 011 00 101 rrr	CB 28+r		*	*	X	0	X	P	0	*	2	8
	SRA (HL)	11 001 011 00 101 110	CB 2E		*	*	X	0	X	P	0	*	4	15
	SRA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 101 110	DD CB d 2E		*	*	X	0	X	P	0	*	6	23
	SRA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 101 110	FD CB d 2E		*	*	X	0	X	P	0	*	6	23
	SRL r	11 001 011 00 111 rrr	CB 38+r		*	*	X	0	X	P	0	*	2	8
	SRL (HL)	11 001 011 00 111 110	CB 3E		*	*	X	0	X	P	0	*	4	15
	SRL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 111 110	DD CB d 3E		*	*	X	0	X	P	0	*	6	23
	SRL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 111 110	FD CB d 3E		*	*	X	0	X	P	0	*	6	23
	RLD	11 101 101 01 101 111	ED 6F		*	*	X	0	X	P	0	–	5	18
	RRD	11 101 101 01 100 111	ED 67		*	*	X	0	X	P	0	–	5	18
BIT SET RESET AND TEST	BIT b,r	11 001 011 01 bbb rrr	CB 40+b×8+r	$Z \leftarrow \overline{r_b}$	X	*	X	1	X	X	0	–	2	8
	BIT b,(HL)	11 001 011 01 bbb 110	CB 46+b×8	$Z \leftarrow \overline{(HL)_b}$	X	*	X	1	X	X	0	–	3	12

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

*1

*1

Note : *1 : Rotate digit left and right between the accmulator and location (HL).
The content of the upper half of the accumulator is unaffected.
The notation $(HL)_b$ indicates bit $_b$ (0 to 7) within the contents of the HL register pair.
The notation r_b indicates bit $_b$ (0 to 7) within the r register.

120489

TMPZ84C00A Instruction Set (8/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
BIT SET RESET AND TEST	BIT b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 01 bbb 110	DD CB d 46+b×8	$Z \leftarrow \overline{(IX+d)_b}$	X	*	X	1	X	X	0	–	5	20
	BIT b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 01 bbb 110	FD CB d 46+b×8	$Z \leftarrow \overline{(IY+d)_b}$	X	*	X	1	X	X	0	–	5	20
	SET b,r	11 001 011 11 bbb rrr	CB C0+b×8+r	$r_b \leftarrow 1$	–	–	X	–	X	–	–	–	2	8
	SET b,(HL)	11 001 011 11 bbb 110	CB C6+b×8	$(HL)_b \leftarrow 1$	–	–	X	–	X	–	–	–	4	15
	SET b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 11 bbb 110	DD CB d C6+b×8	$(IX+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
	SET b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 11 bbb 110	FD CB d C6+b×8	$(IY+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
	RES b,r	11 001 011 10 bbb rrr	CB 80+b×8+r	$r_b \leftarrow 0$	–	–	X	–	X	–	–	–	2	8
	RES b,(HL)	11 001 011 10 bbb 110	CB 86+b×8	$(HL)_b \leftarrow 0$	–	–	X	–	X	–	–	–	4	15
	RES b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 10 bbb 110	DD CB d 86+b×8	$(IX+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
	RES b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 10 bbb 110	FD CB d 86+b×8	$(IY+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
JUMP	JP mn	11 000 011 nn nnn nnn mm mmm mmm	C3 n m	PC←mn	–	–	X	–	X	–	–	–	3	10
	JP c,mn	11 ccc 010 nn nnn nnn mm mmm mmm	C2+c×8 n m	PC←mn (Only when condition is met)	–	–	X	–	X	–	–	–	3	10
	JR $+e	00 011 000 aa aaa aaa	18 a	PC←$+e	–	–	X	–	X	–	–	–	3	12
	JR C,$+e	00 111 000 aa aaa aaa	38 a	If C=0, continue If C=1, PC←$+e	–	–	X	–	X	–	–	–	2 3	7 12
	JR NC,$+e	00 110 000 aa aaa aaa	30 a	If C=0, PC←$+e If C=1, continue	–	–	X	–	X	–	–	–	3 2	12 7
	JR Z,$+e	00 101 000 aa aaa aaa	28 a	If Z=0, continue If Z=1, PC←$+e	–	–	X	–	X	–	–	–	2 3	7 12
	JR NZ,$+e	00 100 000 aa aaa aaa	20 a	If Z=0, PC←$+e If Z=1, continue	–	–	X	–	X	–	–	–	3 2	12 7
	DJNZ $+e	00 010 000 aa aaa aaa	10 a	B←B−1, If B=0, continue B←B−1, If B<>0, continue	–	–	X	–	X	–	–	–	2 3	8 13
	JP (HL)	11 101 001	E9	PC←HL	–	–	X	–	X	–	–	–	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

e represents the extension in the relative addressing mode, a = e − 2. e is a signed two's complement number in the range of $-126 \leq e \leq 129$

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign Negative

Note :
- a = e − 2 in the Opcode provides an effective address of PC + e as PC is incremented by 2 before the addition of e.
- $ indicates the reference to the location counter value of the current segment.
- The notation $(HL)_b$, $(IX + d)_b$ indicates bit $_b$ (0 to 7) within the contents of the register pair.
- The notation r_b indicates bit $_b$ (0 to 7) within the r register.
- a = e-2 in the op-code provides effective address of PC + e as PC is incremented by 2 prior to the addition of e.

270489

TMPZ84C00A Instruction Set (9/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
JUMP	JP (IX)	11 011 101 11 101 001	DD E9	PC←(IX)	–	–	X	–	X	–	–	–	2	8
	JP (IY)	11 111 101 11 101 001	FD E9	PC←(IY)	–	–	X	–	X	–	–	–	2	8
CALL AND RETURN	CALL mn	11 001 101 nn nnn nnn mm mmm mmm	CD n m	(SP−1)←PC_H,(SP−2)←PC_L PC←mn SP←SP−2	–	–	X	–	X	–	–	–	5	17
	CALL c,mn	11 ccc 100 nn nnn nnn mm mmm mmm	C4+cX8 n m	If condition c is met, same as CALL mn. If condition c is not met,continue	– –	– –	X X	– –	X X	– –	– –	– –	5 3	17 10
	RET	11 001 001	C9	PC_L←(SP), PC_H←(SP+1) SP←SP+2	–	–	X	–	X	–	–	–	3	10
	RET c	11 ccc 000	C0+cx8	If condition c is met, same as RET. If condition c is not met,continue	– –	– –	X X	– –	X X	– –	– –	– –	3 1	11 5
	RETI	11 101 101 01 001 101	ED 4D	Return from interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RETN	11 101 101 01 000 101	ED 45	Return from non-maskable interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RST j	11 kkk 111	C7+kx8	(SP−1)←PC_H,(SP−2)←PC_L PC_H←0,PC_L←j,SP←SP−2	–	–	X	–	X	–	–	–	3	11
INPUT AND OUTPUT	IN A,(n)	11 011 011 nn nnn nnn	DB n	A←(n) n→A0~A7,A→A8~A15	–	–	X	–	X	–	–	–	3	11
	IN r,(C)	11 101 101 01 rrr 000	ED 40+rx8	r←(C) If r=110, only the flags will be affected.	*	*	X	*	X	P	0	–	3	12
	INI	11 101 101 10 100 010	ED A2	(HL)←(C),B←B−1,HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	INIR	11 101 101 10 110 010	ED B2	(HL)←(C),B←B−1,HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	IND	11 101 101 10 101 010	ED AA	(HL)←(C),B←B−1,HL←HL−1	X	*M	X	X	X	X	1	X	4	16
	INDR	11 101 101 10 111 010	ED BA	(HL)←(C),B←B−1,HL←HL−1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUT (n),A	11 010 011 nn nnn nnn	D3 n	(n)←A n→A0~A7,A→A8~A15	–	–	X	–	X	–	–	–	3	11
	OUT (C),r	11 101 101 01 rrr 001	ED 41+rX8	(C)←r	–	–	X	–	X	–	–	–	3	12
	OUTI	11 101 101 10 100 011	ED A3	(C)←(HL),B←B−1,HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	OTIR	11 101 101 10 110 011	ED B3	(C)←(HL),B←B−1,HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUTD	11 101 101 10 101 011	ED AB	(C)←(HL),B←B−1,HL←HL−1	X	*M	X	X	X	X	1	X	4	16
	OTDR	11 101 101 10 111 010	ED BB	(C)←(HL),B←B−1,HL←HL−1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16

j	kkk
00H	000
08H	001
10H	010
18H	011
20H	100
28H	101
30H	110
38H	111

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

IN r,(C): ←[B<>0] / ←[B=0] *1

INIR, INDR: ←[B<>0] / ←[B=0]

OTIR: ←B[<>0] / ←[B=0] *1

OTDR: ←[B<>0] / ←[B=0]

Note :
- *M If the result of B − 1 is zero, the Z flag is set, otherwise it is reset.
- A0 through A15 indicate the address bus.
- [ ] indicates the total condition of the number of cycles and states indicated by arrow.

*1 C→A0~A7
B→A8~A15

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-Carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign negative

120489

3.5 USAGE

Basic TMPZ84C00A configurations using memory and peripheral LSIs are described below.

3.5.1 Memory Address Assignment

When the memory is being accessed, the MPU outputs address and control signals. These signals are used as the memory chip enable signals.

The MPU uses 16-bit address signals to specify the addresses for 64K (0-FFFF). With systems having only one memory, memory addresses can be specified with these signals alone. When there are several memories, however, the memories must be arranged so that access is possible using 64K of space. Normally, several address buses are decoded to create this arrangement, several address buses are developed for use as one memory chip enable signal for all memories.

Example: The addresses for an 8K×8-bit ROM and 8K×8-bit RAM are arranged as shown in Figure 3.21. Figure 3.22 shows am example using the $\overline{MREQ}$ signal, $\overline{RD}$ signal and address signal A13 as the chip enable signals.

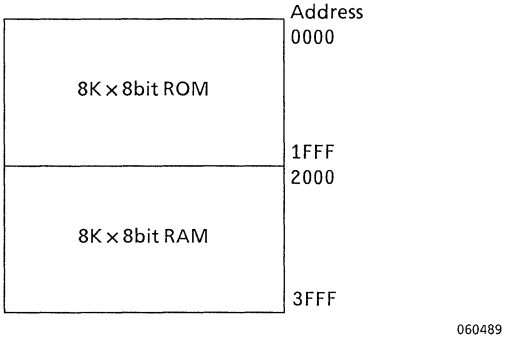


Figure 3.21 Address Assignment

3.5.2 Connection with TLCS-Z80 family peripheral LSI

TMPZ84C00A can connect with peripheral LSI directly. A simple connecting example of the TMPZ84C00A with peripheral LSI is shown in Figure 3.23.

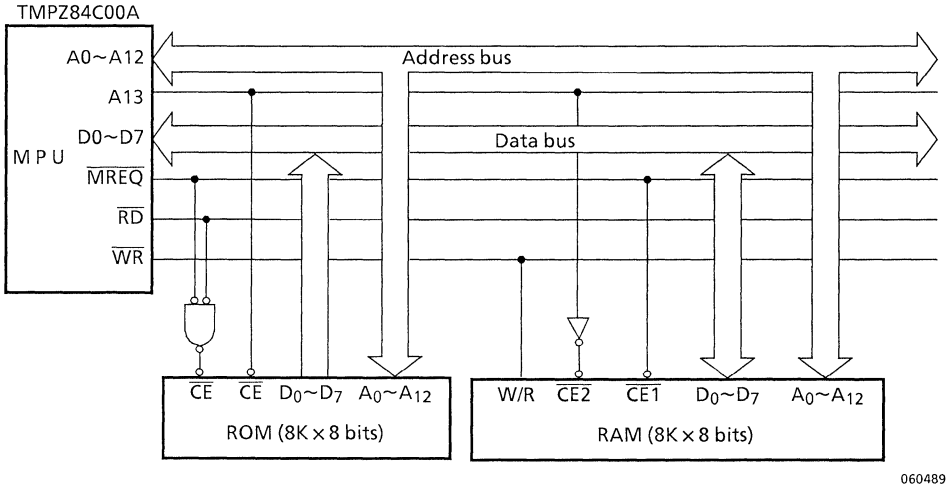
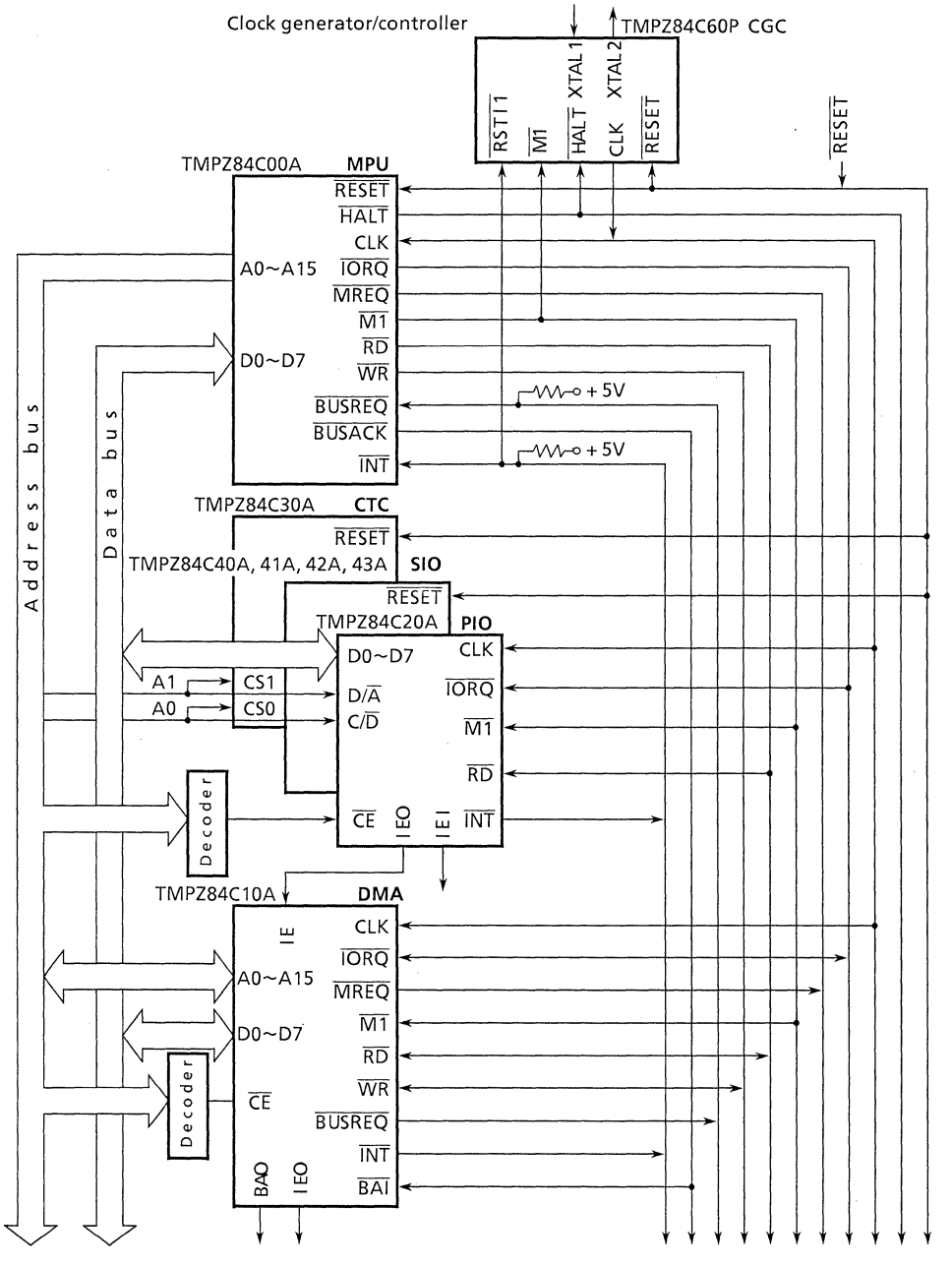


Figure 3.22 Example Connection with Memories



060489

Figure 3.23 Example Connection with TLCS-Z80 family peripheral LSIs

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING	UNIT
V_{CC}	Supply Voltage	− 0.5~ + 7	V
V_{IN}	Input Voltage	− 0.5~VCC + 0.5	V
PD	Power Dissipation (TA = 85°C)	250	mW
T_{SOLDER}	Soldering Temperature (10sec)	260	°C
TSTG	Storage Temperature	− 65~150	°C
TOPR	Operating Temprature	− 40~85	°C

060489

4.2 DC ELECTRICLAL CHARACTERISTICS

DC Characteristics (1/2)
T_{OPR} = − 40°C~85°C, V_{CC} = 5V ± 10%, VSS = 0V

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
VILC	Low Level Clock input Voltage		− 0.3	—	0.6	V
VIHC	High Level Clock input Voltage		V_{CC} − 0.6	—	V_{CC} + 0.3	V
VIL	Input Low Voltage (except CLK)		− 0.5	—	0.8	V
VIH	Input High Voltage (except CLK)		2.2	—	V_{CC}	V
VOL	Output Low Voltage	IOL = 2.0mA	—	—	0.4	V
VOH1	Output High Voltage (I)	IOH = -1.6mA	2.4	—	—	V
VOH2	Output High Voltage (II)	IOH = -250μA	V_{CC} − 0.8	—	—	V
ILI	Input Leak Current	$V_{SS} \leqq V_{IN} \leqq V_{CC}$	—	—	± 10	μA
ILO	3 state Output current in Floating	V_{SS} + 0.4 ≦ V_{OUT} ≦ V_{CC}	—	—	± 10	μA

060489

DC Characteristics (2/2)

SYMBOL	ITEM	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
$I_{CC}1$	Supply Current (Operating)	V_{CC} = 5V fCLK = (Note1) VIHC = VIH = V_{CC} − 0.2V VILC = VIL = 0.2V	AP-6 /AM-6 /AT-6	—	15	22	mA
			AP-8 /AM-8 /AT-8	—	20	25	
$I_{CC}2$ (Note2)	Supply Current (Stand by)	V_{CC} = 5V CLK = (Note2) VIHC = VIH = V_{CC}-0.2V VILC = VIL = 0.2V		—	0.5	10	μA

060489

Note 1 $f_{CLK} = 1/T_CC$ (MIN.)
Note 2 At T4 "LOW" state after the halt instruciton fetch cycle.

4.3 AC ELECTRICAL CHARACTERISTICS (1/3)

$T_{OPR} = -40°C \sim 85°C$, $V_{CC} = 5V \pm 10\%$, $VSS = 0V$

NO.	SYMBOL	ITEM	AP-6/AM-6 /AT-6 (6MHz)		AP-8/AM-8 /AT-8 (8MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	
1	TcC	Clock Cycle Time	165	DC	125	DC	ns
2	TwCh	Clock Pulse Width (High)	65	DC	55	DC	ns
3	TwC1	Clock Pulse Width (Low)	65	DC	55	DC	ns
4	TfC	Clock Fall Time	—	20	—	10	ns
5	TrC	Clock Rise Time	—	20	—	10	ns
6	TdCr (A)	Clock ↑ to Address Valid Delay	—	90	—	80	ns
7	TdA (MREQf)	Address Valid to $\overline{MREQ}$ ↓ Delay	35	—	20	—	ns
8	TdCf (MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay	—	70	—	60	ns
9	TdCr (MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay	—	70	—	60	ns
10	TwMREQh	$\overline{MREQ}$ pulse Width (High)	65	—	45	—	ns
11	TwMREQ1	$\overline{MREQ}$ pulse Width (Low)	135	—	100	—	ns
12	TdCf (MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay	—	70	—	60	ns
13	TdCf (RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay	—	80	—	70	ns
14	TdCr (RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay	—	70	—	60	ns
15	TsD (Cr)	Data Setup Time to Clock ↑	30	—	30	—	ns
16	ThD (RDr)	Data Hold Time to $\overline{RD}$ ↑	0	—	0	—	ns
17	TsWAIT (Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	60	—	50	—	ns

060489

AC Electrical Characteristics (2/3)

NO.	SYMBOL	ITEM	AP-6/AM-6 /AT-6 (6MHz)		AP-8/AM-8 /AT-8 (8MHz)		UNIT
			MIN	MAX.	MIN	MAX.	
18 *	ThWAIT (Cf)	$\overline{WAIT}$ Hold Time after Clock ↓	10	—	10	—	ns
19	TdCr (M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay	—	80	—	70	ns
20	TdCr (M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay	—	80	—	70	ns
21	TdCr (RFSHf)	Clock ↑ to $\overline{RFSH}$ ↓ Delay	—	110	—	95	ns
22	TdCr (RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay	—	100	—	85	ns
23	TdCf (RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay	—	70	—	60	ns
24	TdCr (RDf)	Clock ↑ to $\overline{RD}$ ↓ Delay	—	70	—	60	ns
25	TsD (Cf)	Data Setup to Clock ↓ during M2, M3, M4 or M5 Cycles	40	—	30	—	ns
26	TdA (IORQf)	Address Stable prior to $\overline{IORQ}$ ↓	110	—	75	—	ns
27	TdCr (IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay	—	65	—	55	ns
28	TdCf (IORQr)	Clock ↓ to $\overline{IORQ}$ ↑ Delay	—	70	—	60	ns
29	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	25	—	5	—	ns
30	TdCf (WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay	—	70	—	60	ns
31	TwWR	$\overline{WR}$ Pulse Width	135	—	100	—	ns
32	TdCf (WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay	—	70	—	60	ns
33	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	-55	—	-55	—	ns
34	TdCr (WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay	—	60	—	55	ns
35	TdWRr (D)	Data Stable from $\overline{WR}$ ↑	30	—	15	—	ns
36	TdCf (HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓	—	260	—	225	ns
37	TwNMI	$\overline{NMI}$ Pulse Width	70	—	60	—	ns
38	TsBUSREQ (Cr)	$\overline{BUSREQ}$ Setup Time to Clock ↑	50	—	40	—	ns
39 *	ThBUSREQ(Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	10	—	10	—	ns
40	TdCr(BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay	—	90	—	80	ns
41	TdCf(BUSACKr)	Clockt ↓ to $\overline{BUSACK}$ ↑ Delay	—	90	—	80	ns
42	TdCr(Dz)	Clock ↑ to Data Float Delay	—	80	—	70	ns
43	TdCr(CTz)	Clock ↑ to Control Out-puts Float Delay($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)	—	70	—	60	ns
44	TdCr(Az)	Clock ↑ to Address Float Delay	—	80	—	70	ns

060489

AC Electrical Characteristics (3/3)

NO.	SYMBOL	ITEM	AP-6/AM-6 /AT-6 (6MHz)		AP-8/AM-8 /AT-8 (8MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	
45	TdCr(A)	$\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ to Address Hold Time	35	—	20	—	ns
46	TsRESET(Cr)	$\overline{RESET}$ to Clock ↑ setup Time	60	—	45	—	ns
47 *	ThRESET(Cr)	$\overline{RESET}$ to Clock ↑ Hold Time	10	—	10	—	ns
48	TsINTf(Cr)	$\overline{INT}$ to Clock ↑ Setup Time	70	—	55	—	ns
49 *	TsINTr(Cr)	$\overline{INT}$ to Clock ↑ Hold Time	10	—	10	—	ns
50 *	TdM1f(IORQf)	$\overline{M1}$ ↓ to $\overline{IORQ}$ ↓ Delay	365	—	270	—	ns
51	TdCf(IORQf)	Clock ↓ to $\overline{IORQ}$ ↓ Delay	—	70	—	60	ns
52	TdCr(IORQr)	Clock ↑ to $\overline{IORQ}$ ↑ Delay	—	70	—	60	ns
53	TdCf(D)	Clock ↓ to Data Valid Delay	—	130	—	115	ns

060489

Note 1 AC Test Condition
$VIH = 2.4V$, $VIL = 0.4V$, $VIHC = V_{CC} - 0.6V$, $VILC = 0.6V$
$VOH = 2.2V$, $VOL = 0.8V$, $CL = 100PF$

Note 2 Items with an asterisk (*) are non-compatible with NMOS Z80.

4.4 CAPACITANCE

$TA = 25°C$

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	Clock input Capacitance	f = 1MHz All pins except measured are connected to GND.	—	—	8	pF
CIN	Input Capacitance		—	—	6	pF
COUT	Output Capacitance		—	—	10	pF

060489

4.5 TIMING DIAGRAM

Figure 4.1 to 4.8 show the basic timings of respective operations. Numbers shown in the Figures correspond with those in the AC Electrical Characteristics Table in 4.3.

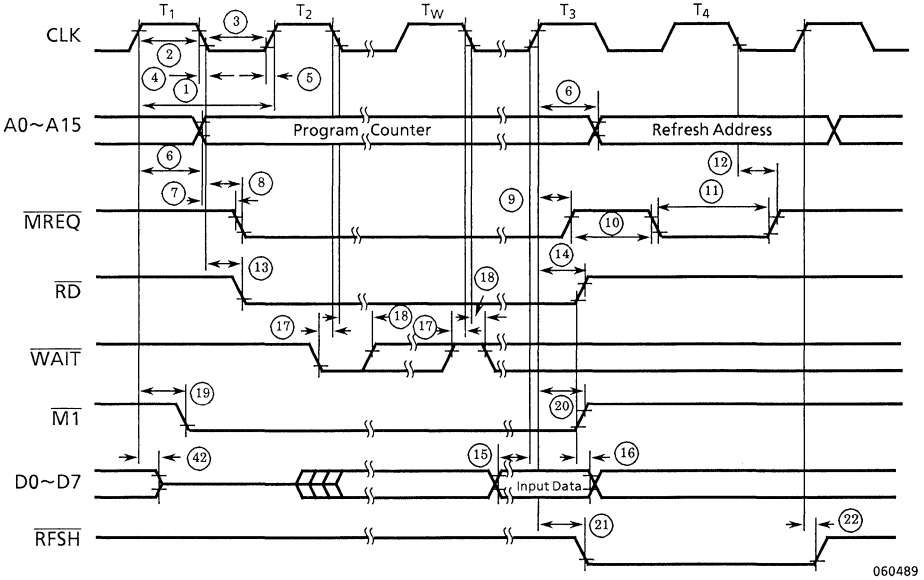


Figure 4.1 Operation Code Fetch Cycle

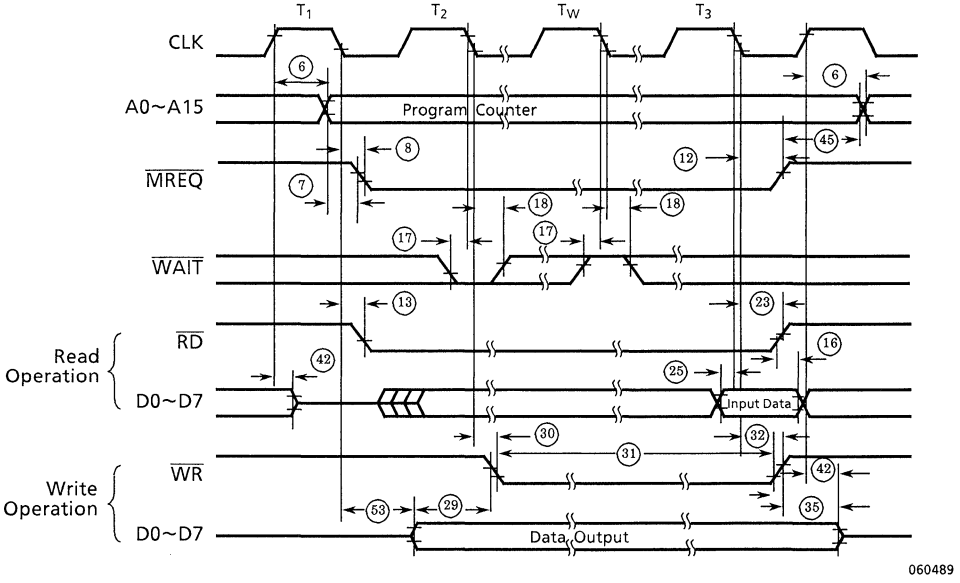
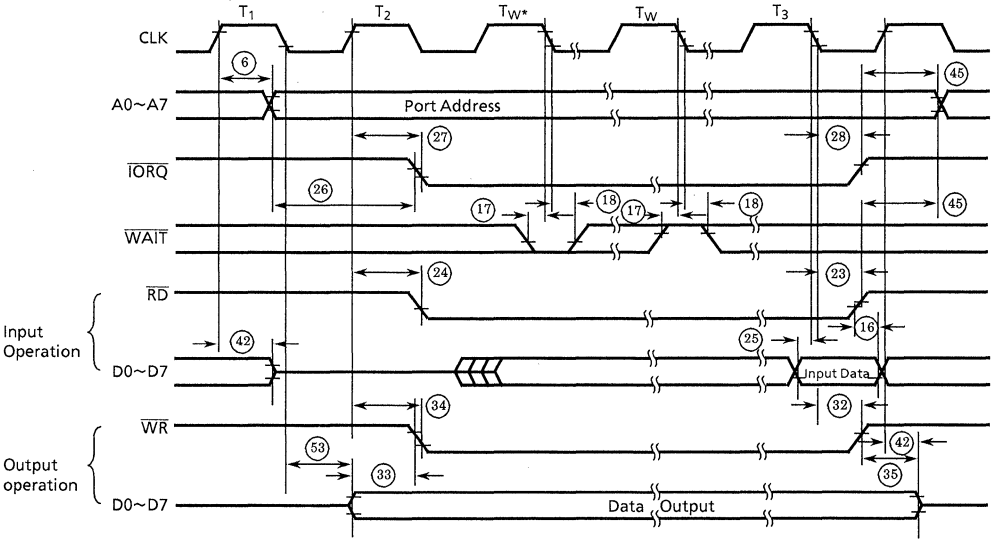
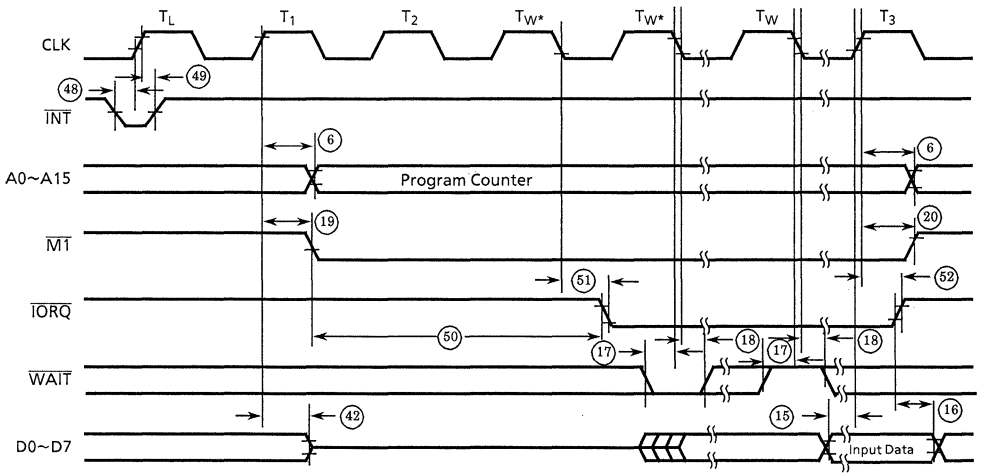


Figure 4.2 Memory Read/Write Cycle



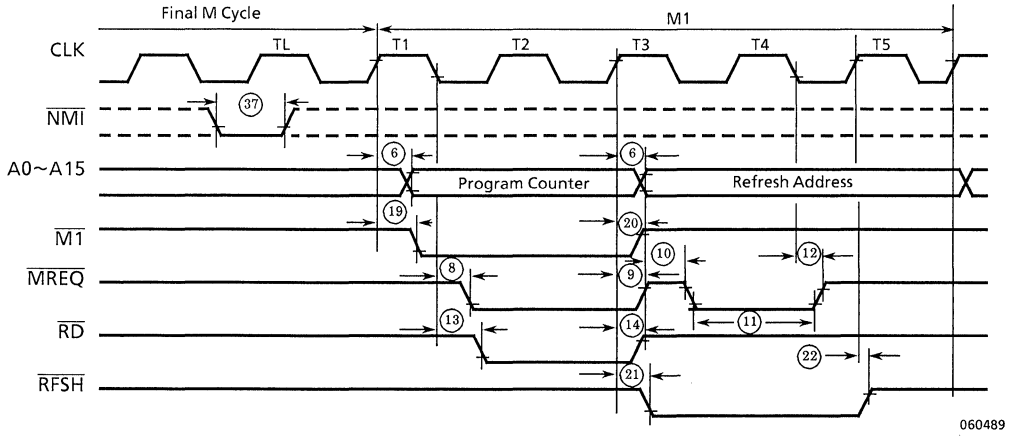
Note : 1 wait state (TW*) is inserted automatically by MPU. 060489

Figure 4.3 Input/Output Cycle



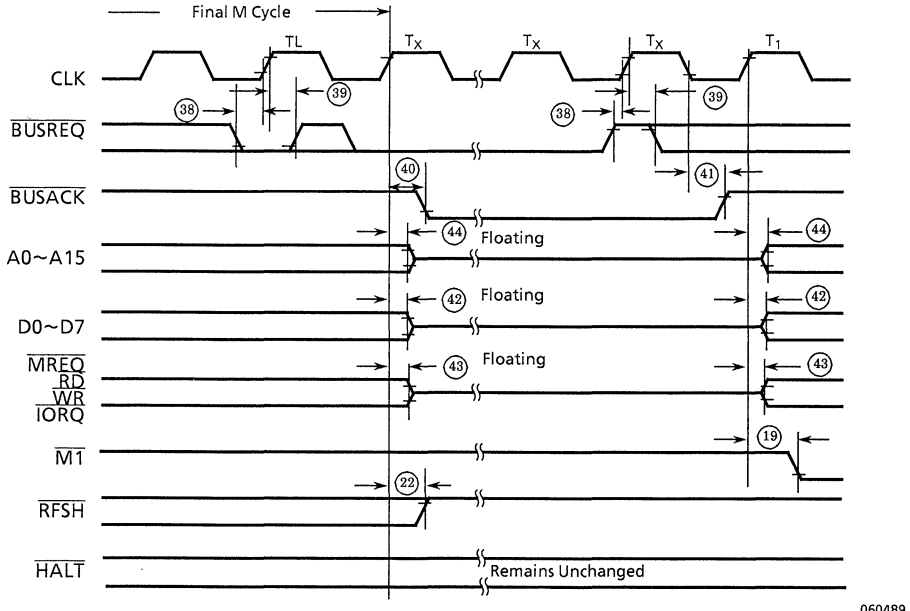
Note 1 TL is the final state of the preceding instruction.
Note 2 2 wait state (TW*) is inserted automatically by MPU. 060489

Figure 4.4 Interrupt Request/Acknowledge Cycle



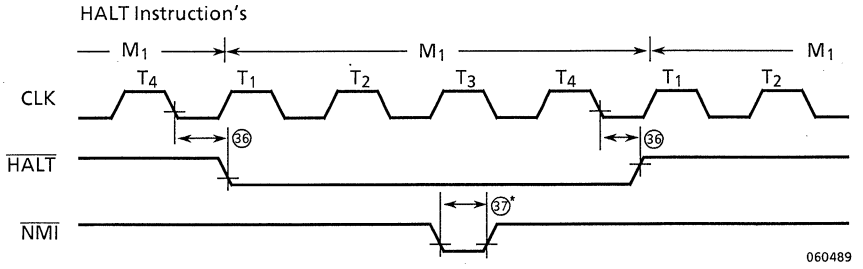
Note : $\overline{NMI}$ is asynchoronous input but in order to assure the positive response in the following cycle, $\overline{NMI}$ trailing edge signal must be generated keeping abreast of the leading edge of the preceding TL state.

Figure 4.5 Non-maskable Interrupt Request Cycle



Note 1 : TL is the final state of any machine cycle
Note 2 : TX is optional clock used by requested peripheral LSI.

Figure 4.6 Bus Request/Acknowledge Cycle



Note : $\overline{INT}$ signal is also used for releasing from the halt state

Figure 4.7 Halt Acknowledge Cycle

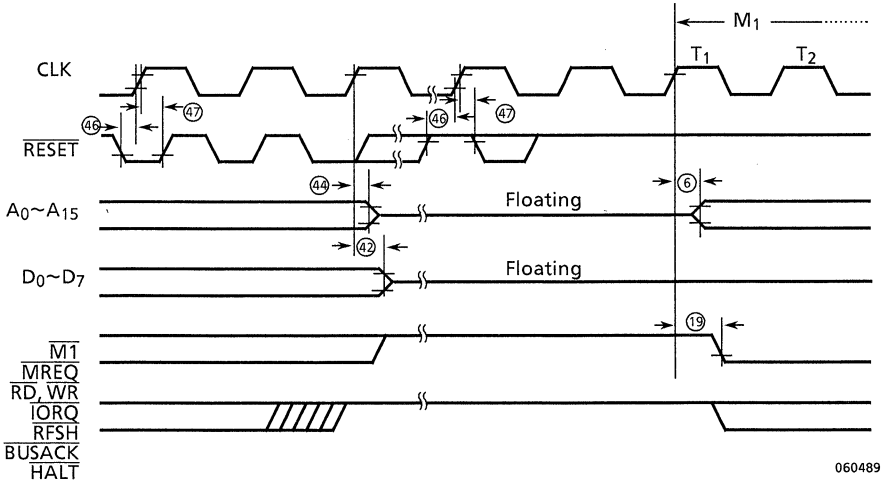


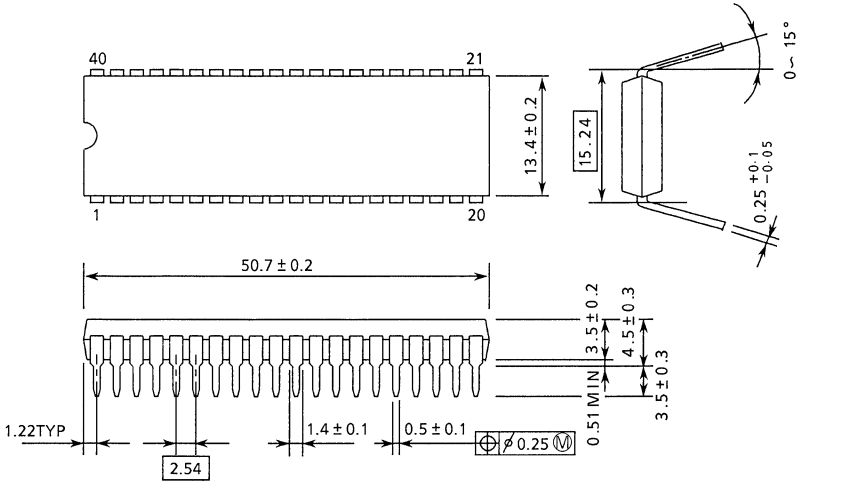
Figure 4.8 Reset Cycle

5. PACKAGE DIMENSION

5.1 DIP PACKAGE

DIP40-P-600

Unit : mm

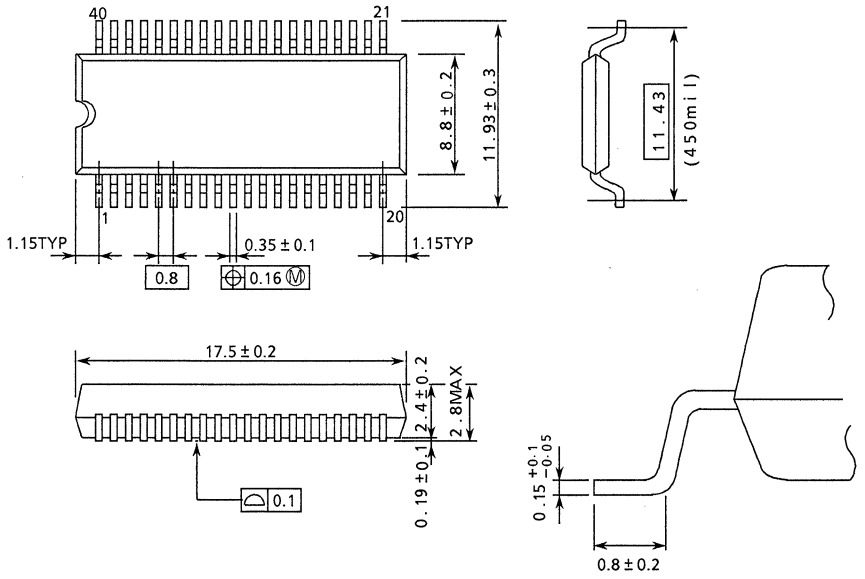


Note 1 : This dimension is measured at the center of bending points of leads.
Note 2 : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.40 leads.

5.2 SOP PACKAGE

SSOP40-P-450

Unit : mm



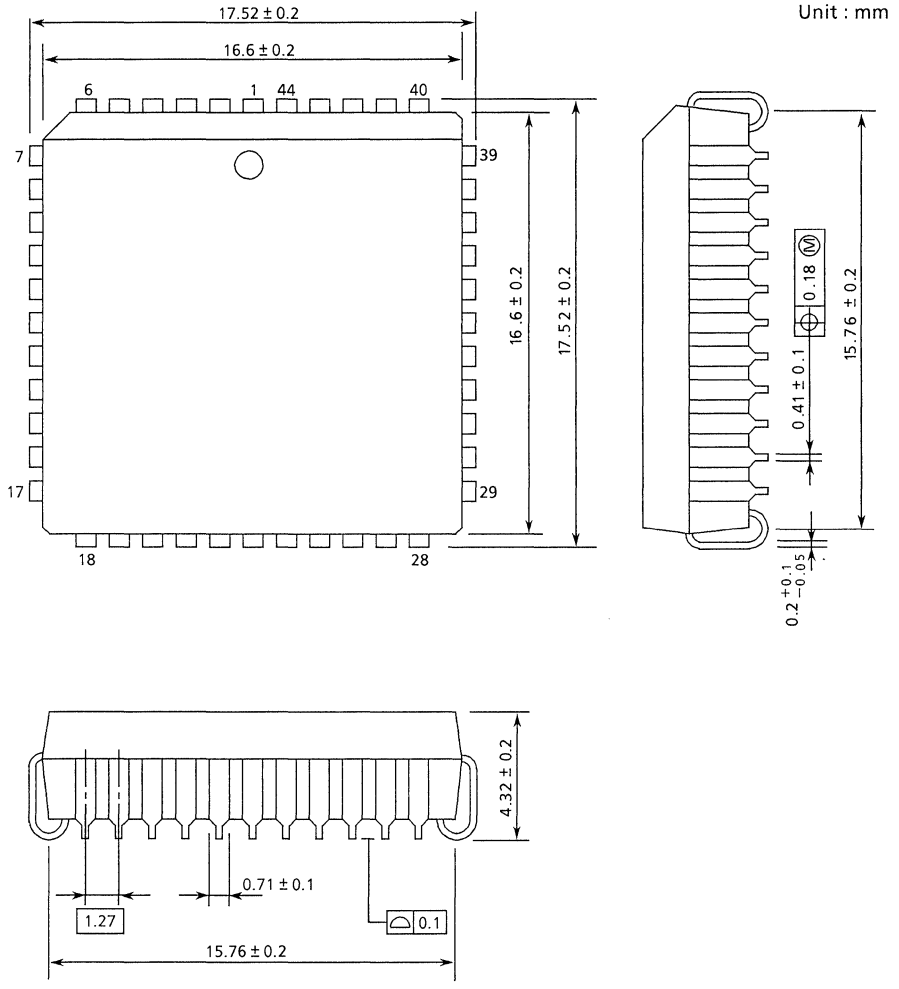
270289

Note : Package Width and length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

5.3 PLCC PACKAGE

QFJ44-P-S650

Unit : mm



270289

6. CAUTIONS

Please observe the following cautions when using the TMPZ8400A.

(1) The $\overline{RESET}$ signal input used for resetting must be held at "0" for at least 3 clocks.

(2) When the MPU is not the bus master ($\overline{BUSREQ}=0$), the memory is not refreshed because the $\overline{RFSH}$ signal is "1" and address signals are at high impedance. With systems using dynamic RAM, an external circuit is required for memory refresh if this condition persists for any length of time.

Also, interrupts cannot be received when the MPU is not the bus master.

(3) When exiting a power down operation with the MPU in hold status, supply the prescribed stabilized clock.

(4) Maskable interrupt mode 2 is only for use with Z80 family peripheral LSIs.

(5) Only the program counter, interrupt enable flip-flop, internal NMI flip-flop, I register and R register of the MPU are initialized. All other registers must be initialized by program when necessary. Also, set the interrupt mode to mode 0.

(6) The interrupt enable flip-flop is set to "1" by the instruction following the EI instruction to enable receipt of maskable interrupts.

(7) Only the program counter register is saved during interrupt processing. Save and restore interrupt processing routines as necessary.

(8) When using maskable interrupt mode 2, a data table for the vector addresses must be created in the memory.

(9) When periphral LSIs and memory are connected with the MPU on a PCB, use wiring as large as possible and the shortest routing for connecting Vss (GND) and Vcc.

Caution is necessary because of the large spike currents which can occur when signals change ($0\rightarrow1$, $1\rightarrow0$) with high-speed versions.

(10) As countermeasures for the above, connect a capacitor with good pulse response between Vcc and Vss (GND) of the MPU and other devices to absorb the pulse current.

TMPZ84C01F, TMPZ84C02AF-6
TLCS-Z80MPU : 8-BIT MICROPROCESSOR
CMOS Z80 8-BIT MICROPROCESSOR

1. OUTLINE AND FEATURES

The TMPZ84C01F/02AF-6 is an 8-bit microprocessor (hereinafter referred to as MPU) with a built-in clock generator/controller, which provides low power operation and high performance.

Built into the TMPZ84C01F/02AF-6 are a control function and clock generator for the standby function in addition to paired 6 general purpose registers, accumulator, flag registers, an arithmetic-and-logic unit, bus control, memory control and timing control circuits. TMPZ84C02AF-6 can insure duty 50% clock output easily with Internal Oscillation Divider.

The TMPZ84C01F/02AF-6 is fabricated using Toshiba's CMOS Silicon Gate Technology and molded in a 44-pin mini-flat package.

The principal functions and features of the TMPZ84C01F/02AF-6 are as follows.

(1) Instruction Set compatible with the Zilog's Z80 MPU.

(2) Low power consumption

Table 1.1 Operation modes and Supply Current (Vcc = 5v.TYPICAL)

Product name	Operating frequency	RUN MODE	IDLE1 MODE	IDLE2 MODE	STOP MODE
TMPZ84C01F	fosc = 4MHz fCLK = 4MHz	15mA	1mA	3mA	0.5μA
TMPZ84C02AF-6	fosc = 12MHz fCLK = 6MHz	20mA	1.5mA	4.5mA	0.5μA

170489

(3) DC to 6MHz operation (at 5V ± 10%)

(4) Single 5V power supply (at 5V ± 10%)

(5) Operating temperature (− 40°C to 85°C)

(6) On-chip clock generator

(7) In the HALT state, the following 4 modes are selectable:

- Run mode
- IDLE 1 mode
- IDLE 2 mode
- STOP mode

In the following explanation for the same content, IDLE1 Mode and IDLE2 Mode are referred to as IDLE1/2.

(8) Powerful set of 158 instructions available

(9) Powerful interrupt function

(a) Non-maskable interrupt terminal ($\overline{NMI}$)

(b) Maskable interrupt terminal ($\overline{INT}$)

The following 3 modes are selectable ;

- 8080 compatible interrupt mode (interrupt by Non-Z80 family peripharal LSI) (Mode 0)
- Restart interrupt (Mode 1)
- Daisy chain structure interrupt using Z80 family peripheral LSI (Mode 2)

(10) An auxiliary register provided to each of general purpose registers.

(11) Two index registers

(12) 10 addressing modes

(13) Built-in refresh circuit for dynamic memory.

(14) Molded in 44-pin mini flat package

(15) Built-in clock divider for insuring duty 50% easily (TMPZ84C/02AF-6)

(16) Low voltage oparation Vcc = 2.7V ~ 5.5V (TMPZ84C01F)

Further, in the following text and explanations for charts and tables, hexadecimal numbers are directly used without giving an identification to explanation of address, etc. to the extent not to cause confusions.

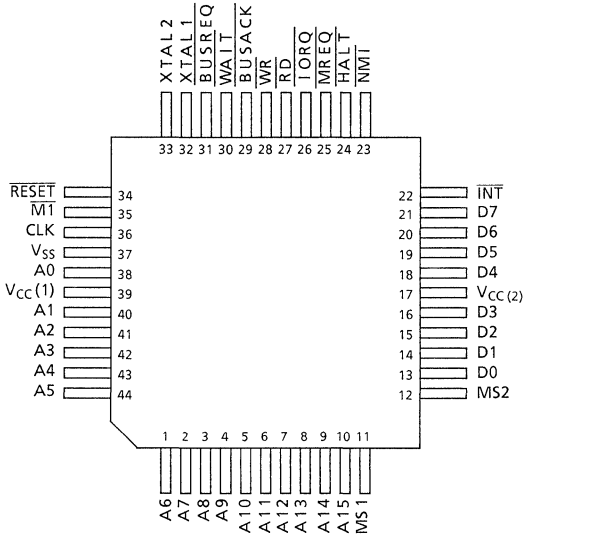
Note : Z80 is a trademark of Zilog Inc., U.S.A.

2. PIN CONNECTIONS AND PIN FUNCTIONS

The pin connections and I/O pin names and brief functions of the TMPZ84C01/02A are shown below.

2.1 PIN CONNECTIONS

The pin connections of the TMPZ84C01/02A are as shown in Figure 2.1.



Note : Connect Pin 39 and Pin 17 externally.

Figure 2.1 Pin Connections (Top View)

2.2 PIN NAMES AND FUNCTIONS

I/O pin names and functions are as shown in Table 2.1.

Table 2.1 Pin Names and Functions (1/3)

Pin Name	Number of Pin	Input/Output 3-state	Function
A0~A15	16	Output 3-state	16-bit address bus. Specify addresses of memories and I/O to be accessed. During the refresh period, addresses for refreshing are output.
MS1, MS2	2	Input	Mode selection input. One of 4 modes (RUN, IDLE1/2, STOP) is selected according to the state of these 2 pins.

060489

Table 2.1 Pin Names and Functions (2/3)

Pin Name	Number of Pin	Input/Output 3-state	Function
D0-D7	8	I/O 3-state	8-bit bidirectional data bus.
$\overline{INT}$	1	Input	Maskable interrupt request signal. Interrupt is generated by peripheral LSI. This signal is accepted if the interrupt enable flip-flop (IFF) is set at "1". $\overline{INT}$ is normally used on Wired-OR. In this case, a pull-up resistor is externally connected.
$\overline{NMI}$	1	Input	Non-maskable interrupt request signal. This interrupt request has the higher priority than the maskable interrupt request and does not rely upon the state of the interrupt enable flip-flop (IFF).
$\overline{HALT}$	1	Output	Halt signal. MPU execute HALT instruction and when the halt state is resulted. "0" is output.
$\overline{MREQ}$	1	Output 3-state	Memory request signal. When an effective address for memory access is on the address bus, "0" is output.
$\overline{IORQ}$	1	Output 3-state	I/O request signal. When addresses for I/O are on the lower 8 bits (A0-A7) of the address bus in the I/O operation, "0" is output. In addition, $\overline{IORQ}$ signal is output together with $\overline{M1}$ signal at time of interrupt acknowledge cycle to inform peripheral LSI of the state that the interrupt response vector may be put on the data bus.
$\overline{RD}$	1	Output 3-state	Read signal "0" signal is output for a period when MPU can receive data from a memory or peripheral LSI. It is possible to put data from a specified peripheral LSI or memory on the MPU data bus after gating by this signal.
$\overline{WR}$	1	Output 3-state	Write signal. This signal is output when data to be stored in a specified memory or peripheral LSI is on the MPU data bus.
$\overline{BUSACK}$	1	Output	Bus acknowledge signal. In response to $\overline{BUSREQ}$ signal, this signal informs a peripheral LSI of the fact that the address bus, data bus, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$ signals have been placed in the high impedance state.
$\overline{WAIT}$	1	Input	Wait signal. $\overline{WAIT}$ signal is a signal to inform MPU of specified memory or peripheral LSI which is not ready for data transfer. As long as $\overline{WAIT}$ signal as at "0" level, MPU is continuously kept int the wait state.

060489

Table 2.1 Pin Names and Functions (3/3)

Pin Name	Number of Pin	Input/Output 3-state	Function
$\overline{BUSREQ}$	1	Input	Bus request signal. $\overline{BUSREQ}$ signal is a signal requesting placement of the address bus , data bus, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$ signals in the high impedance state. $\overline{BUSREQ}$ signal is normally used on wired-OR. In this case, a pull-up resistor is externally connected.
$\overline{RESET}$	1	Input	Reset signal. $\overline{RESET}$ signal is used for initialization MPU and must be kept in active state ("0") for a period of at least 3 clocks.
$\overline{MI}$	1	Output	Signal showing machine cycle 1. "0" is output together with $\overline{MREQ}$ signal in the operation code fetch cycle. This signal is output for every opcode fetch when 2byte opcode is executed. In the maskable interrupt acknowledge cycle, this signal is output together with $\overline{IORQ}$ signal.
XTAL 1 (XIN) XTAL 2 (XOUT)	2	Input output	Crystal oscillator connecting terminal. For 01F, connects an oscillator having the oscillation frequency as high as the system clock (CLK) frequency. For 02AF/02AF-6, connects an oscillator having the oscillation frequency 2 times as high as the system clock (CLK) frequency.
CLK	1	Output	Single-phase clock output. When the HALT instruction in STOP Mode is executed, MPU stops its operation and holds clock output at "0" level.
Vcc (1), (2)	2	Power supply	+5 Connent 39 pin and 17 pin externally.
Vss	1	Power supply	0V

060489

3. FUNCTIONAL DESCRIPTION

The system configuration, functions and basic operation of the TMPZ84C01F/02AF-6 are described here.

3.1 BLOCK DIAGRAM

The block diagram of the interval configuration is shown in Figure 3.1.

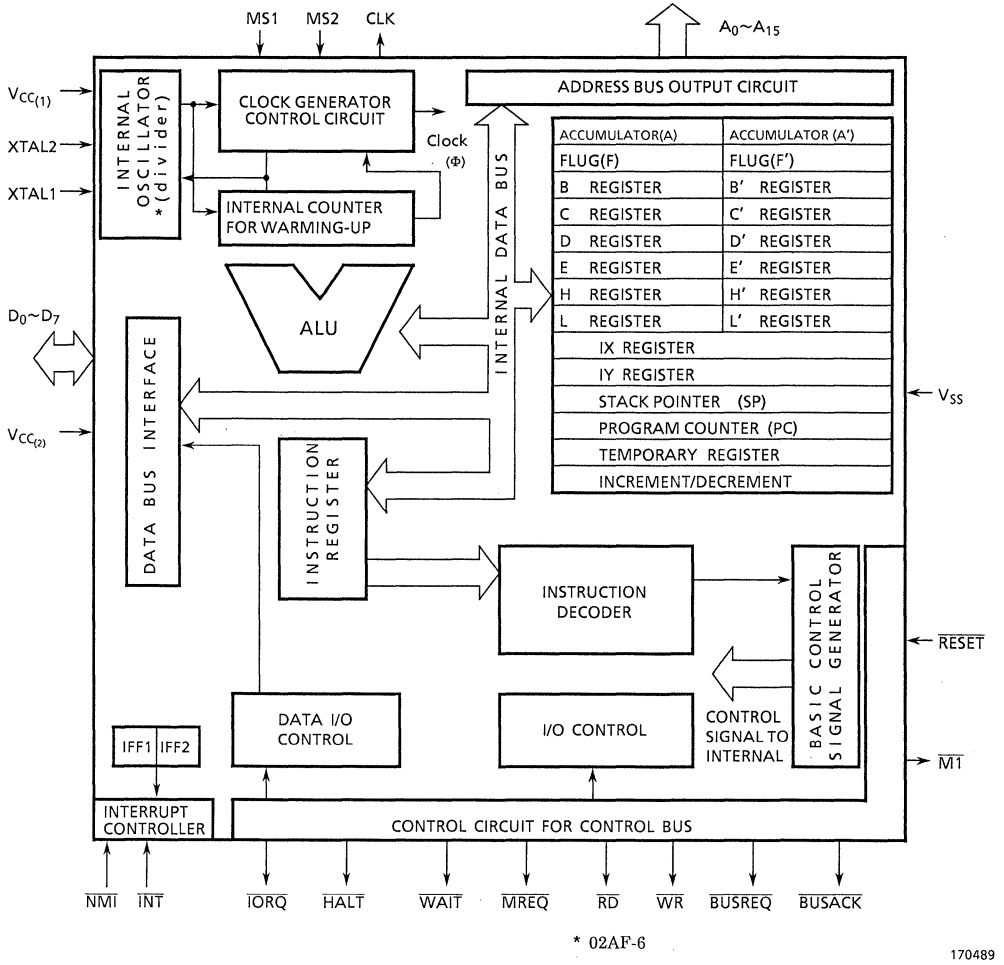


Figure 3.1 Block Diagram

3.2 SYSTEM CONFIGURATION

The TMPZ84C01F/02AF-6 has a built-in system clock generator for CMOS Z80 in addition to the standard functions of the TMPZ84C00A CMOS Z80 MPU.

The explanation is provided here with emphases placed on the halt function relative to the clock generator, which is an additional function. The internal register group, reset and interrupt function are identical to those of the TMP84C00A. For details please refer to the data sheet for the TMPZ84C00A.

In this section, the following principal components and functions will be described,

(1) Generation of clock

(2) Operation mode

(3) Warming-up time at time of restart

3.2.1 Generating the system clock

The TMPZ84C01F/02AF-6 has a built-in oscillation circuit and required clock can be easily generated by connecting an oscillator to the external terminals (XTAL1, XTAL2). For the TMPZ84C01F, Clock in the same frequency as input oscillation frequency is generated. As the TMPZ84C02AF-6 has a built-in divider, Clock in the half frequency as input oscillation frequency is generated.

Examples of oscillator connection are shown in Figure 3.2.

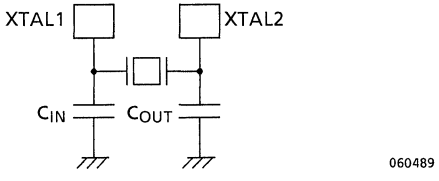


Figure 3.2 Example of Oscillator Connection

Table 3.1 External connecting capacitance

Oscillation Frequency	CIN	COUT	Application Product
4MHz	22PF	33PF	TMPZ84C01F
8MHz	22PF	33PF	TMPZ84C02AF-6
12MHz	33PF	33PF	TMPZ84C02AF-6

170489

In case of using Crystal oscillator, please use products with the following characteristics or use the following maker's products.

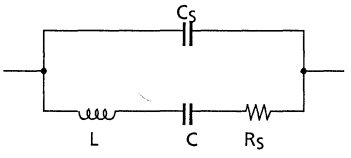


Figure 3.2 Crystal oscillator equivalent circuit

Table 3.2 Crystal oscillator necessary characteristics

Oscillation Frequency	CS	Rs	Application Product
4MHz	under4PF	under50Ω	TMPZ84C01F
8MHz	under4PF	under30Ω	TMPZ84C02AF-6
12MHz	under4PF	under25Ω	TMPZ84C02AF-6

Table 3.3 Crystal oscillator of recommendation

Oscillation Frequency	Product name	Maker
4MHz	MR4000-C20	Tokyo Denpa Inc.
8MHz	MR8000-C20	
12MHz	MR12000-C20	

170489

3.2.2 Operation modes

There are 4 kinds of operations modes available for the TMPZ84C01F/02AF-6 in connection with generation of clock ; RUN Mode, IDLE1/2 Modes and STOP Mode. One of these modes is selected by the mode select inputs (MS1, MS2).

The operation mode is effective when the halt instruction is executed and when the halt instruction is not executed, clock is supplied continuously. Restart of MPU from the stopped state under IDLE1/2 Mode or STOP Mode is effected by inputting either $\overline{\text{RESET}}$ signal or interrupt signal ($\overline{\text{INT}}$ or $\overline{\text{NMI}}$).

Operations of these modes in the halt state are shown in Table 3.4.

Table 3.4 Clock Generating Operation Modes

Operation Mode	MS1	MS2	Description at HALT State
RUN Mode	1	1	MPU continues the operation and supplies clock to the outside continuously.
IDLE1 Mode	0	0	The internal oscillator's operation only is continued and clock (CLK) output as well as internal operation are stopped at "0" level of T4 state in the halt instruction operation code fetch cycle.
IDLE2 Mode	0	1	The internal oscillator's operation and clock (CLK) output are continued but the internal operation are stopped at "0" level of T4 state in the halt instruction operation code fetch cycle.
STOP Mode	1	0	All operations of the internal oscillator, clock (CLK) output, and internal operation are stopped at "0" level of T4 state in the halt instruction operation code fetch cycle.

060489

3.2.3 Warming-up time at time of restart (STOP Mode)

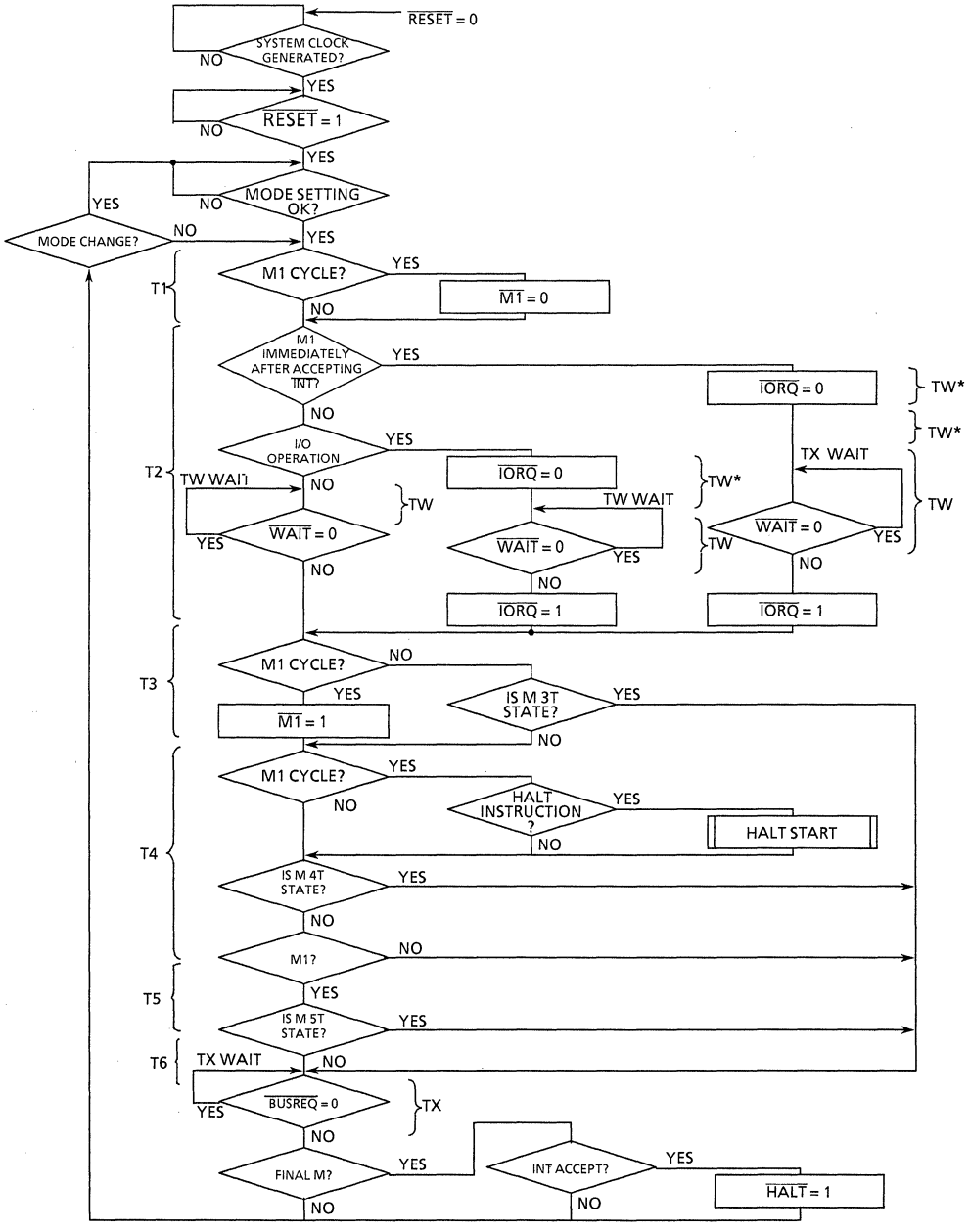
When MPU is released from the halt state by accepting an interrupt request,MPU, then, will execute an interrupt service routine. Therefore, when an interrupt request is accepted, MPU starts generation of internal system clock and clock output after a warming-up time by the internal counter ($2^{14}+2.5$) TcC (TcC ; Clock Cycle) to obtain a stabilized oscillation for MPU operation.

Further, in case of the restart by $\overline{\text{RESET}}$ signal, the internal counter does not operate for a quick operation at time of power ON.

3.3 STATUS CHANGE FLOWCHART AND BASIC TIMING

In this section, the status change and basic timing when the TMPZ84C01F/02AF-6 is operating are explained.

3.3.1 Status change flowchart



060489

Figure 3.4 (a) Status Change Flowchart

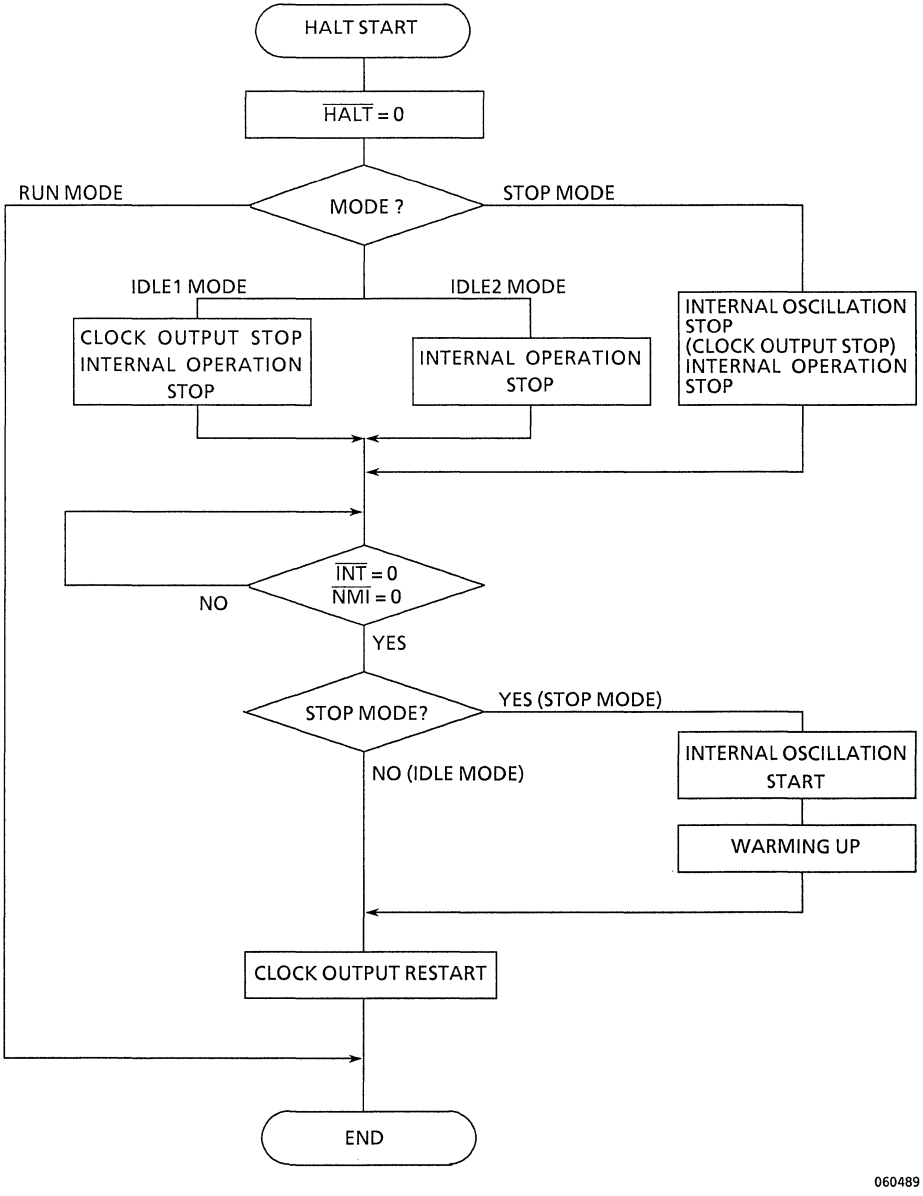


Figure 3.4 (b) Status Change Flowchart

3.3.2 Basic timing

The basic timing is explained here with emphases placed on the halt function relative to the clock generator. Except $\overline{RFSH}$ signal output, the following items are identical to those for the TMPZ84C00A. Refer to the data sheet for the TMPZ84C00A.

- Operation code fetch cycle
- Memory read/write operation
- Input/output operation
- Bus request/acknowledge operation
- Maskable interrupt request accepting operation
- Non-maskable interrupt request accepting operation
- Reset operation

Note that the TMPZ84C01F/02AF-6 dose not have the refresh terminal $\overline{RFSH}$ but refresh address is output on the address bus in the operation code fetch cycle ($\overline{M1}$) as in the TMPZ84C00A since the on-chip refresh control circuit is available.

(1) Operation when HALT instruction is execution

When MPU fetches a halt instruction in the operation code fetch cycle, $\overline{HALT}$ signal goes active (low level) in synchronous with falling edge of T4 state for the peripheral LSI and MPU stops the operation. The system clock generating operation after this differs depending upon the operation mode (RUN Mode, IDLE1/2 Mode or STOP Mode). If the internal system clock is running, MPU continues to execute NOP instruction even in the halt state.

(a) RUN Mode (MS1 = 1, MS2 = 1)

Shown in Figure 3.6 is the basic timing when the halt instruction is executed in RUN Mode.

In RUN Mode, system clock (ø) in MPU and clock output (CLK) to the outside of MPU are not stopped even after the halt instruction is executed. Therefore, until the halt state is released by the interrupt signal ($\overline{MNI}$ OR $\overline{INT}$) or $\overline{RESET}$ signal, MPU continues to execute NOP instruction.

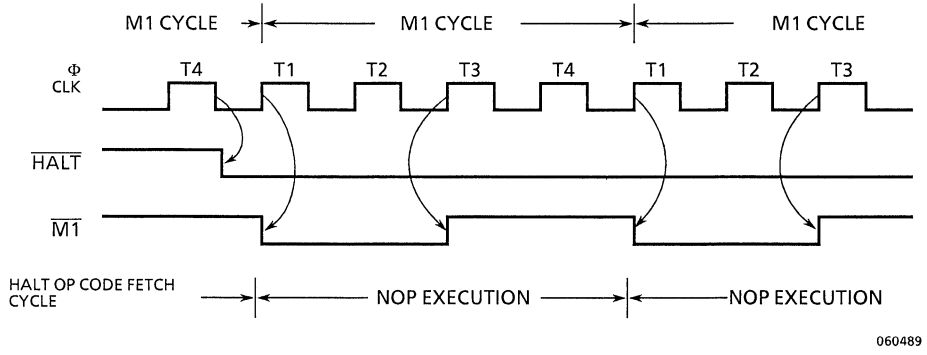


Figure 3.6 Timing of RUN Mode (at Halt Instruction Execution)

(b) IDLE1 Mode (ML1 = 0, MS2 = 0)

Shown in Figure 3.7 is the basic timing when the halt instruction is executed in IDLE1 Mode.

In IDLE1 Mode, system clock (ø) in MPU and clock output (CLK) to the outside of MPU are stopped and MPU stops its operation after the halt instruction is executed.

However, the internal oscillator continues to operate.

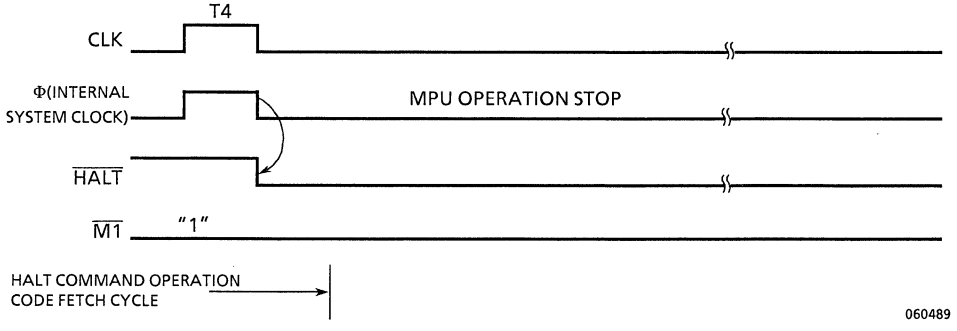


Figure 3.7 IDLE1 Mode Timing (at Halt Instruction Execution)

(c) IDLE2 Mode (MS1 = 0, MS2 = 1)

Shown in Figure 3.8 is the basic timing when the halt instruction is executed in IDLE2 Mode.

In IDLE2 Mode, system clock (ø) in MPU is stopped and MPU stops its operation after the halt instruction is executed.

However,the internal oscillator and clock output (CLK) to the outside of MPU continues to operate.

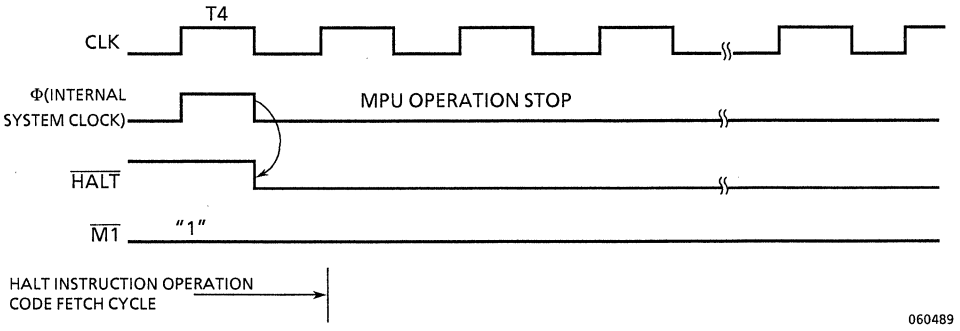


Figure 3.8 IDLE2 Mode timing (at Halt Instruction Execution)

(d) STOP Mode (MS1 = 1,MS2 = 0)

Shown in Figure 3.9 is the basic timing when the halt instruction is executed in STOP Mode.

In STOP Mode, internal operation and internal oscillator are stopped after the halt instruction is executed. Therefore, system clock (ø) in MPU and clock output (CLK) to the outside of MPU are stopped.

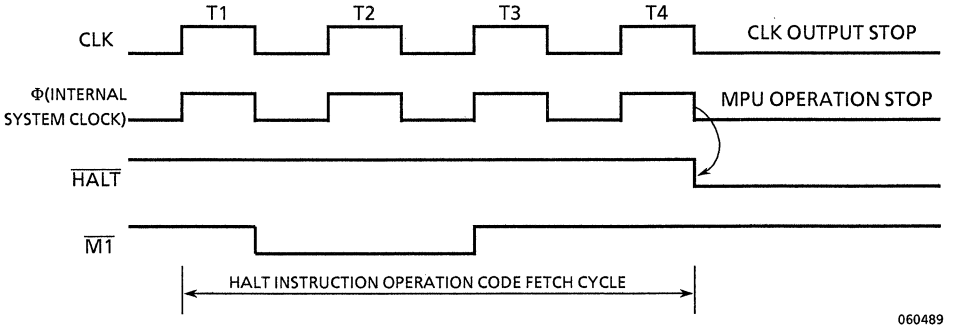


Figure 3.9 STOP Mode Timing (at Halt Instruction Execution)

(2) Release from halt state

The halt state of MPU is released when "0" is input to $\overline{\text{RESET}}$ signal and MPU is reset or an interrupt request is accepted. An interrupt request signal is sampled at the leading edge of the last clock cycle (T4 state) of NOP instruction. In case of the maskable interrupt, interrupt will be accepted by an active $\overline{\text{INT}}$ signal ("0" level). In case of the non-maskable interrupt, if the internal $\overline{\text{NMI}}$ F/F which is set at the leading edge of $\overline{\text{MNI}}$ signal is set to "1", the interrupt is accepted.

However, in case of the maskable interrupt, the interrupt enable flip-flop must have been set to "1". The accepted interrupt process is started from next cycle.

Further, when the internal system clock is stopped (IDLE1/2 Mode, STOP Mode), it is necessary first to restart the internal system clock. The internal system clock is restarted when $\overline{\text{RESET}}$ or interrupt signal ($\overline{\text{NMI}}$ or $\overline{\text{INT}}$) is input.

(a) RUN Mode (MS1, MS2 = 1)

The halt release operation by acceptance of interrupt request in RUN Mode is shown in Figure 3.10.

In RUN Mode the internal system clock is not stopped and therefore, if the interruption signal is recognized at the rise of T4 state of the continued NOP instruction, MPU will executes the interrupt process from next cycle.

The halt release operation by resetting MPU in RUN Mode is shown in Figure 3.11. After reset, MPU will execute an instruction starting from address 0000H. However, in order to reset MPU it is necessary to keep $\overline{\text{RESET}}$ signal at "0" for at least 3 clocks. In addition, if $\overline{\text{RESET}}$ signal becomes "1", after the dummy cycle for at least 2T states, MPU executes an instruction from address 0000H.

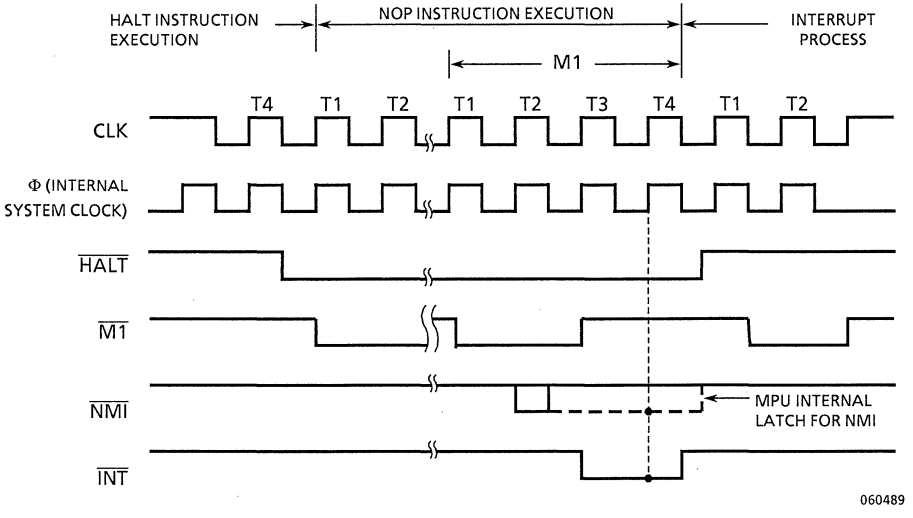


Figure 3.10 Halt Release Operation Timing by Interrupt Request Signal in RUN Mode

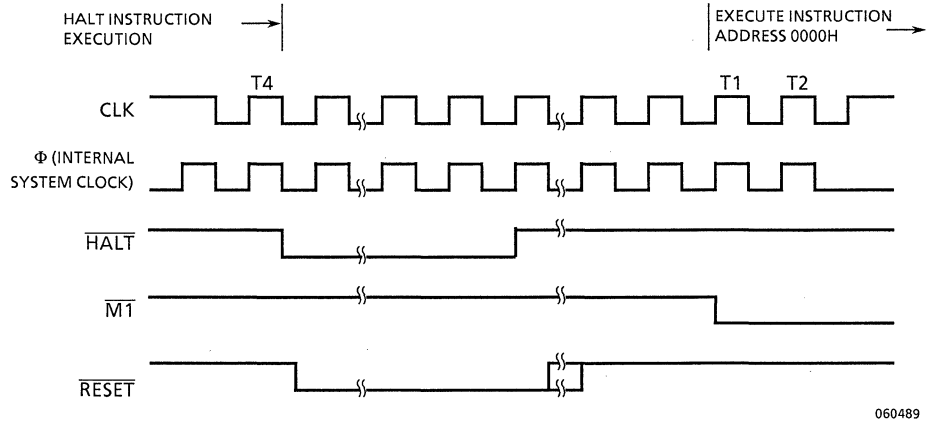


Figure 3.11 Halt Release Operation Timing by Reset in RUN Mode

(b) IDLE1 Mode (MS1 = 0, MS2 = 0), IDLE2 Mode (MS1 = 0, MS2 = 1)

The halt release operation by interrupt signal in IDLE1 Mode is shown in Figure 3.12 (a) and in IDLE2 Mode in Figure 3.12 (b).

When receiving $\overline{NMI}$ or $\overline{INT}$ signal, MPU starts the internal system clock operation . In IDLE1 Mode, MPU starts clock output to the outside at the same time. The operation stop of MPU in IDLE1/2 Mode is taken place at "0" level during T4 state in the halt instruction operation code fetch cycle. Therefore, after restarted by the interruption signal, MPU executes one NOP instruction and samples an interrupt signal at the rise of T4 state during the execution of this NOP instruction, and executes the interrupt process from next cycle.

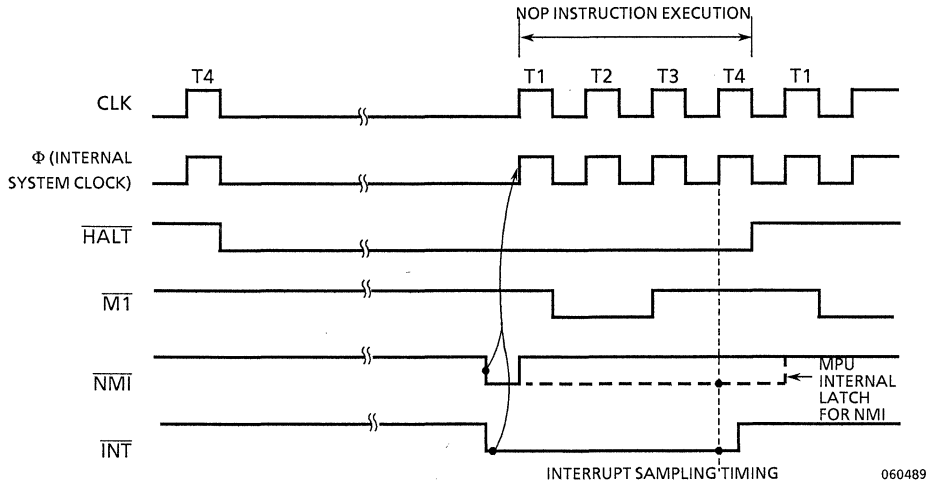


Fig 3.12 (a) IDLE1 Mode

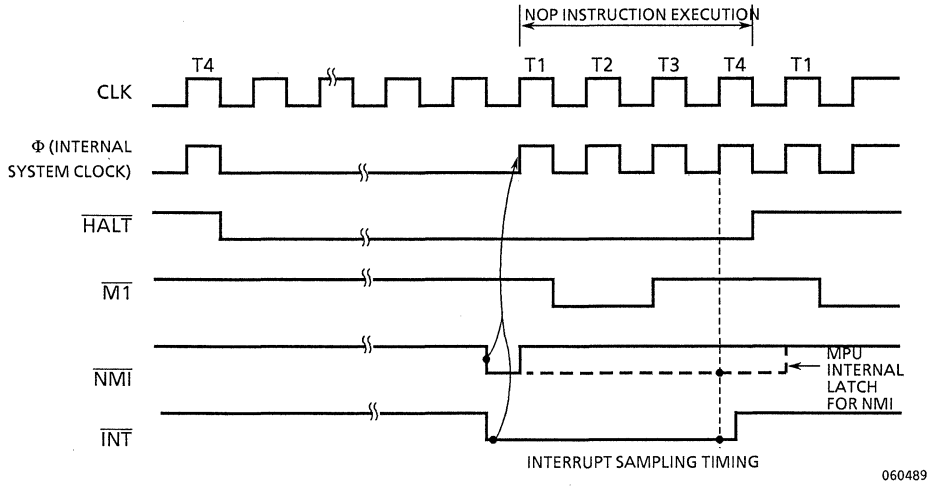


Fig 3.12(b) IDLE2 Mode

Fig.3.12Halt Release Operation Timing by Interrupt Request Signal in IDLE1/2 Mode

If no interrupt signal is accepted during the execution of the first NOP instruction after the internal system clock is restarted, MPU is not released from the halt state and is placed in IDLE1/2 Mode again at "0" level during T4 state of the NOP instruction, stopping the internal system clock. If $\overline{INT}$ signal is not at "0" level at the rise of T4 state, no interrupt request is accepted.

The halt release operation by resetting MPU in IDLE1 Mode is shown in Figure 3.13 (a) and that in IDLE2 Mode in Figure 3.13 (b).

When $\overline{\text{RESET}}$ signal at "0" level is input into MPU, the internal system clock is restarted and MPU will execute an instruction stored in address 0000H.

At time of $\overline{\text{RESET}}$ signal input, it is necessary to take the same care as that in resetting MPU in RUN Mode.

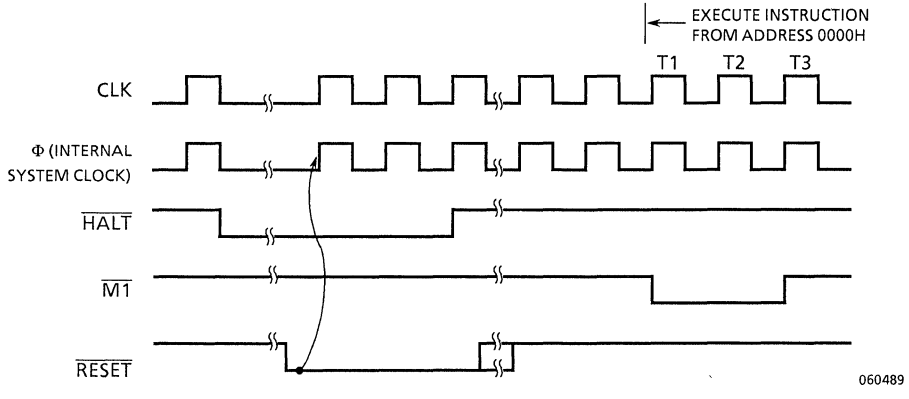


Figure 3.13 (a) IDLE1 Mode

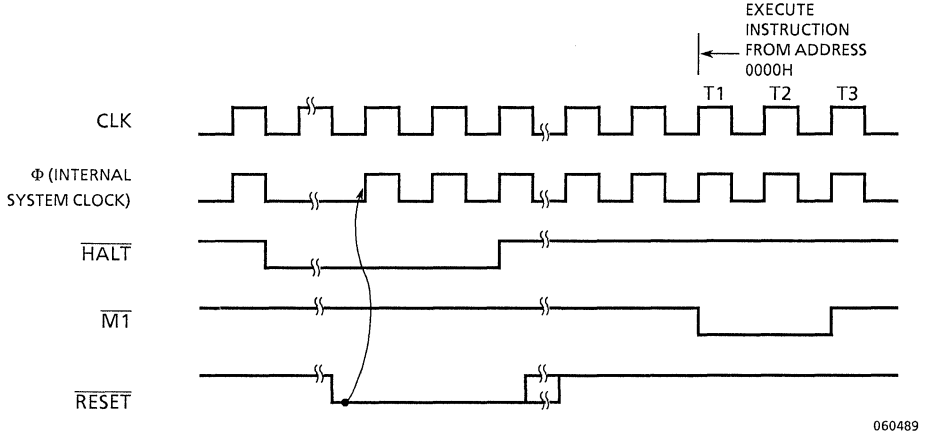


Figure 3.13 (b) IDLE2 Mode

(c) STOP Mode (MS1 = 1, MS2 = 0)

The halt release operation by interrupt signal in STOP Mode is shown in Figure 3.14.

When MPU received an interrupt signal, the internal oscillator is restarted. In order to obtain stabillized oscillation, the internal system clock and clock output to the outside are started after a warming-up time of $(2^{14}+2.5)$ TcC (TcC : Clock Cycle) by the internal counter passed.

MPU executes one NOP instruction after the internal system clock is restarted and at the same time, sampling an interrupt signal at the rise of T4 state during the execution of this NOP instruction. If the interrupt signal is accepted, MPU executes the interrupt process operation from next cycle.

At time of interrupt signal input, it is necessary to take the same care as that in the interrupt signal input in IDLE1/2 Mode. The halt release operation by MPU resetting in STOP Mode is shown in Figure 3.15.

When $\overline{\text{RESET}}$ signal at "0" level is input into MPU, the internal oscillator is restarted. However, since it performs a quick operation at time of power ON, the internal counter does not operate. Therefore, the operation may not be carried out properly due to unstable clock immediately after the internal oscillator is restarted. To restart the clock by $\overline{\text{RESET}}$ signal in STOP Mode, it is necessary to hold $\overline{\text{RESET}}$ signal at "0" level for sufficient time. When $\overline{\text{RESET}}$ signal becomes "1", after the dummy cycle for at least 2T states, MPU starts to execute an execution from address 0000H.

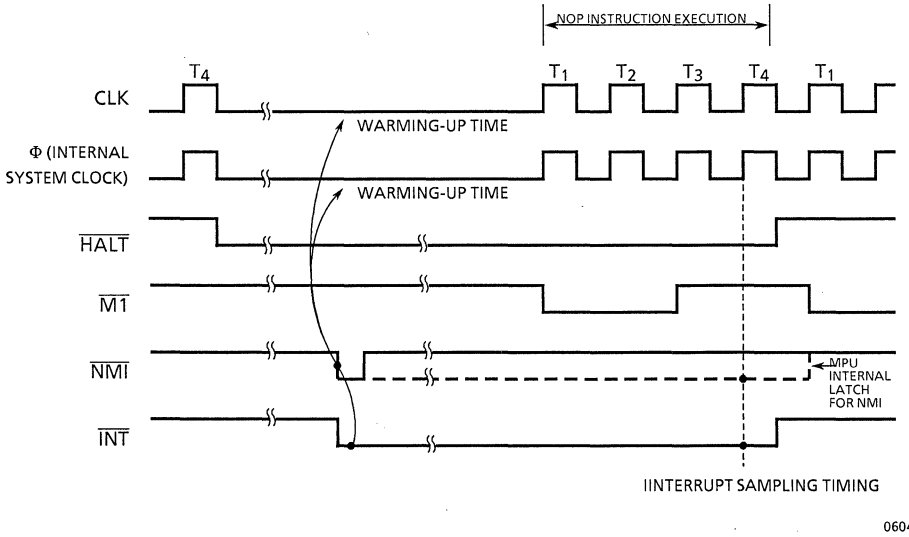


Figure 3.14 Halt Release Operation Timing by Interrupt Request Signal in STOP Mode

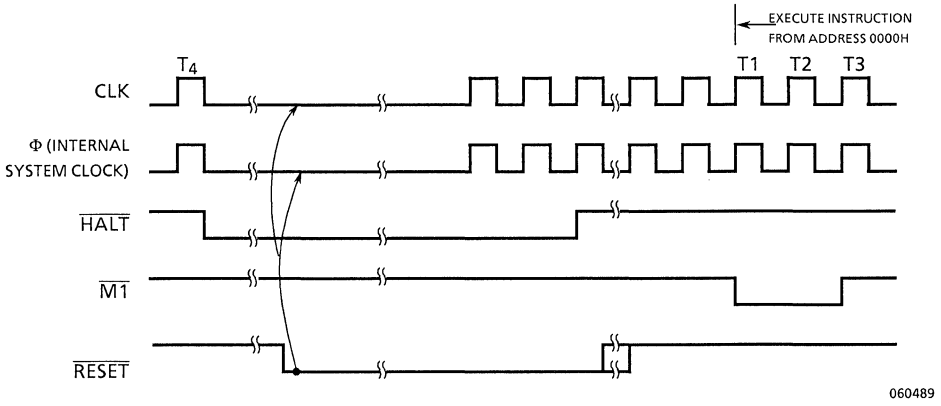


Figure 3.15 Halt Release Operation Timing by Reset in STOP Mode

3.4 INSTRUCTION SET

Instruction set of the TMPZ84C01F/02AF-6 are the same as those for the TMPZ84C00A. For details refer to the data sheet for the TMPZ84C00A.

3.5 METHOD OF USE

A connecting example of the TMPZ84C01F/02AF-6 with the TLCS-Z80 family peripheral LSI's is shown in Figure 3.16. For the explanation and precautions for connection, refer to Section 3.5 Method of use of the data sheet for the TMPZ84C00A.

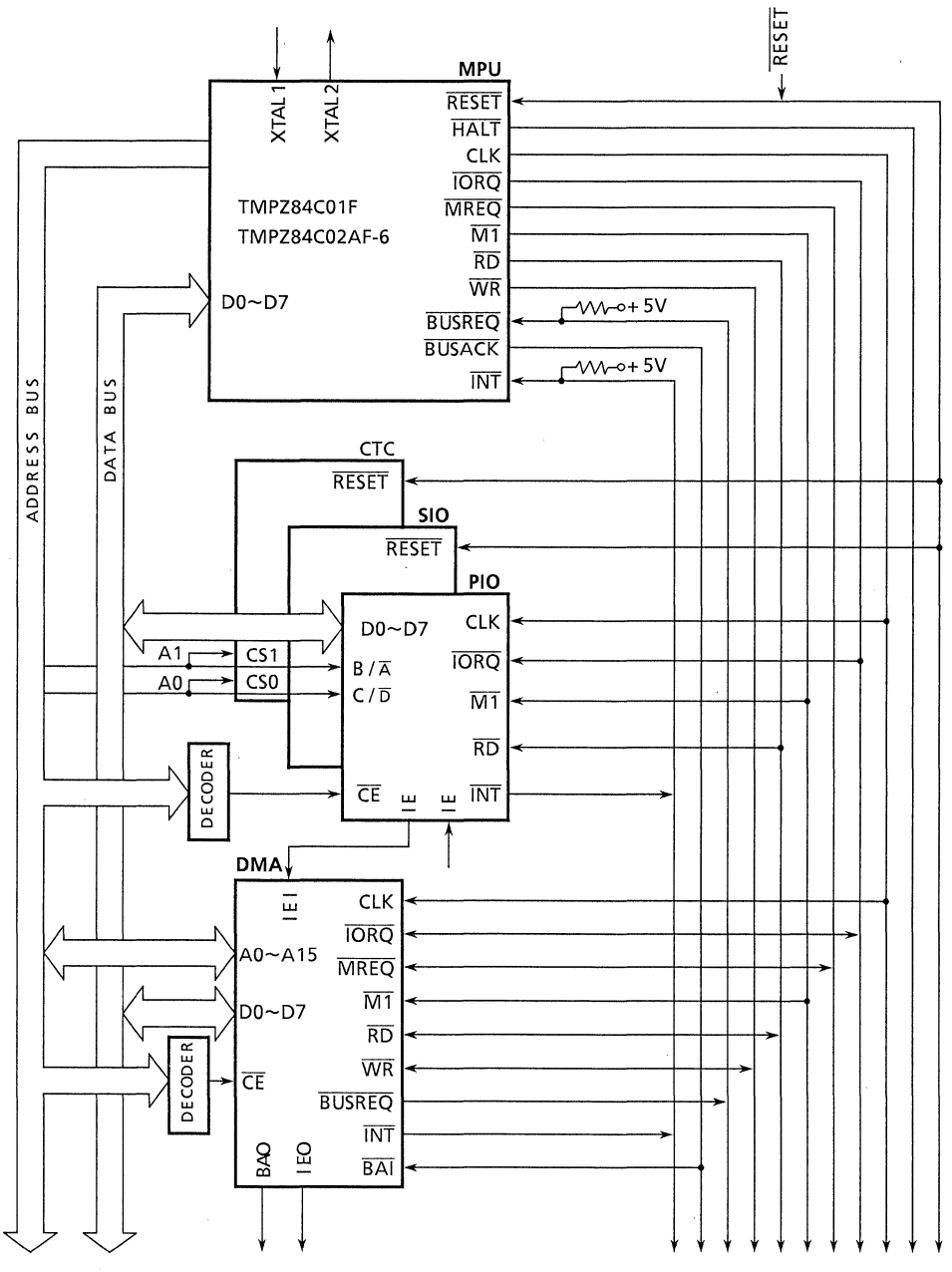


Figure 3.16 Example Connection with Z80 family peripheral LSI

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING	UNIT
VCC	Supply Voltage	− 0.5 to + 7	V
VIN	Input Voltabe	− 0.5 to V_{CC} + 0.5	V
PD	Power Dissipation (TA = 85°C)	250	mW
TSOLDER	Soldering Temperature (10sec)	260	°C
TSTG	Storage Temperature	− 65 to 150	°C
TOPR	Operating Temperature	− 40 to 85	°C

060489

4.2 DC ELECTRICAL CHARACTERISTICS

4.2.1 DC Chracteristics

$T_A = 10°C$ to $+60°C$ (TMPZ84C01F @ Low voltage operation)

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
VCC	Supply Voltage		2.7	—	5.5	V
VIHC	Input High Voltage (XTAL1)		VCC-0.2	—	VCC + 0.3	V
VIH	Input High Voltage (Except XTAL1)		VCC-0.2	—	VCC + 0.3	V
VILC	Input Low Voltage (XTAL1)		-0.3	—	0.2	V
VIL	Input Low Voltage (Except XTAL1)		-0.5	—	0.2	V

060489

4.2.2 DC Chracteristics (I)

$VCC = 2.7V$ to $5.5V$, $T_A = -40°C$ to $+85°C$ (TMPZ84C01F @ Low voltage operation)

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
ILI	Input Leak Current	VSS≦VIN≦VCC	—	—	±10	μA
ILO	3 State Output Current in Floating	VSS≦VIN≦VCC	—	—	±10	μA
IOH	Output High Current (Except XTAL2)	VOH = VCC-0.2V	0.1	—	—	mA
IOL	Output Low Current (Except XTAL2)	VOL = 0.2V	0.4	—	—	mA
ICC1	Supply Current (@Run Mode)	VCC = 3V, f = 1MHz VIH = Vcc, VIL = 0V	—	2.5	5.0	mA
ICC2	Supply Current (@Stop Mode)	VCC = 3V, VIH = Vcc, VIL = 0V	—	0.5	10	μA
ICC3	Supply Current (@IDLE1 Mode)	VCC = 3V, f = 1MHz VIH = Vcc, VIL = 0V	—	0.5	1.0	mA
ICC4	Supply Current (@IDLE2 Mode)	VCC = 3V, f = 1MHz VIH = Vcc, VIL = 0V	—	1.0	2.0	mA

060489

4.2.3 DC Chracteristics (II)

$T_A = -40°C$ to $85°C$, $VCC = 5V \pm 10\%$, $VSS = 0V$

SYMBOL	ITEM	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
VOLC	Low Level Clock Output Voltabe	IOL = 2.0mA		—	—	0.4	V
VOHC	High Level Clock Output Voltage	IOH = −2.0mA		V_{CC} − 0.6	—	—	V
VIL	Input Low Voltage			−0.5	—	0.8	V
VIH	Input High Voltage			2.2	—	VCC	V
VIHR	Input High Voltage ($\overline{RESET}$)			V_{CC} − 0.6	—	VCC	V
VILR	Input Low Voltage ($\overline{RESET}$)			−0.5	—	0.45	V
VOL	Output Low Voltage (Except Clock)	IOL = 2.0mA		—	—	0.4	V
VOH1	Output High Voltage (I) (Except Clock)	IOH = −1.6mA		2.4	—	—	V
VOH2	Output High Voltage (II) (Except Clock)	IOH = −250uA		VCC − 0.8	—	—	V
ILI	Input Leak Current	VSS ≦ VIN ≦ VCC		—	—	± 10	μA
ILO	3 State Output Current in Floating	VSS + 0.4 ≦ VOUT ≦ VCC		—	—	± 10	μA
ICC1	Supply Current (@ RUN Mode)	VCC = 5V, fCLK = (NOTE1) VIHC = VIH = VCC − 0.2, VILC = VIL = 0.2V	01F	—	15	20	mA
			02AF-6	—	20	24	

170489

SYMBOL	ITEM	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
ICC2	Supply Current (@ STOP Mode)	VCC = 5V, fCLK = (Note2) VIHC = VIH = VCC – 0.2V VILC = VIL = 0.2V		–	0.5	10	μA
ICC3	Supply Current (@ IDLE1 Mode)	VCC = 5V, fCLK = (Note1) VIHC = VIH = VCC – 0.2V VILC = VIL = 0.2V	01F	–	1.0	2.0	mA
			02AF-6	–	1.5	3.0	
ICC4	Supply Current (@ IDLE2 Mode)	VCC = 5V, fCLK = (Note1) VIHC = VIH = VCC – 0.2V VILC = VIL = 0.2V	01F	–	3.0	6.0	mA
			02AF-6	–	4.5	7.5	

170489

Note 1 : $fCLK = 1/T_{CC}$ (MIN)
Note 2 : At T4 "LOW" state of the halt instruction fetch cycle.

4.3 AC ELECTRICAL CHARACTERISTICS

TOPR = −40℃ to 85℃, VCC = 5V ± 10%, VSS = 0V

(1/4)

No.	SYMBOL	ITEM	01F (V_{CC} = 2.7V~5.5V fCLK = 1MHz)		01F (fCLK = 4MHz)		02AF-6 (fCLK = 6MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
1	T_{cC}	Clock frequency	1000	DC	250	DC	165	DC	ns
2	T_{WCh}	High clock pulse width	400	DC	110	DC	65	DC	ns
3	T_{WCl}	Low clock pulse width	400	DC	110	DC	65	DC	ns
4	TfC	Clock falling time	—	200	—	30	—	20	ns
5	TrC	Clock rising time	—	200	—	30	—	20	ns
6	TdCr (A)	Effective address output delay from clock rise	—	400	—	110	—	90	ns
7	TdA (MREQf)	Address output definite time prior to $\overline{MREQ}$	200	—	65	—	35	—	ns
8	TdCf (MREQf)	Delay from clock fall to $\overline{MREQ}$ = "L"	—	300	—	85	—	70	ns
9	TdCr (MREQr)	Delay from clock rise to $\overline{MREQ}$ = "H"	—	300	—	85	—	70	ns
10	TwMREQh	$\overline{MREQ}$ high level pulse width	400	—	110	—	65	—	ns
11	TwMREQl	$\overline{MREQ}$ low level pulse width	800	—	220	—	135	—	ns
12	TdCf (MREQr)	Delay from clock fall to $\overline{MREQ}$ = "H"	—	300	—	85	—	70	ns
13	TdCf (RDf)	Delay from clock fall to $\overline{RD}$ = "L"	—	350	—	95	—	80	ns
14	TdCr (RDr)	Delay from clock rise to $\overline{RD}$ = "H"	—	300	—	85	—	70	ns
15	TsD (Cr)	Data set-up time for clock rise	180	—	35	—	30	—	ns
16	ThD (RDr)	Data hold time for $\overline{RD}$ rise	0	—	0	—	0	—	ns
17	TsWAIT (Cf)	$\overline{WAIT}$ signal set-up time for clock fall	350	—	70	—	60	—	ns
18 *	ThWAIT (Cf)	$\overline{WAIT}$ hold time after clock fall	10	—	10	—	10	—	ns
19	TdCr (M1f)	Delay from clock rise to $\overline{M1}$ = "L"	—	400	—	100	—	80	ns

170489

(2/4)

No.	SYMBOL	ITEM	01F (V_{CC} = 2.7V~5.5V fCLK = 1MHz)		01F (fCLK = 4MHz)		02AF-6 (fCLK = 6MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
20	TdC r(M1r)	Delay from clock rise to $\overline{M1}$ = "H"	—	400	—	100	—	80	ns
21	TdCf (RDr)	Delay from clock fall to $\overline{RD}$ = "H"	—	300	—	85	—	70	ns
22	TdCr (RDf)	Delay from clock rise to $\overline{RD}$ = "L"	—	300	—	85	—	70	ns
23	TsD (Cf)	Data set-up time for clock fall (at time of M2, M3, M4, M5 cycle)	250	—	50	—	40	—	ns
24	TdA (IORQf)	Address definite time prior to $\overline{IORQ}$ fall	550	—	180	—	110	—	ns
25	TdCr (IORQf)	Delay from clock rise to $\overline{IORQ}$ = "H"	—	300	—	75	—	65	ns
26	TdCf (IORQr)	Delay from clock fall to $\overline{IORQ}$ = "H"	—	300	—	85	—	70	ns
27	TdD (WRf)	Data definit time prior to $\overline{WR}$ fall	200	—	80	—	25	—	ns
28	TdCf (WRf)	Delay from clock fall to $\overline{WR}$ = "L"	—	300	—	80	—	70	ns
29	TwWR	$\overline{WR}$ pulse width	750	—	220	—	135	—	ns
30	TdCf (WRr)	Delay from clock fall to $\overline{WR}$ = "H"	—	300	—	80	—	70	ns
31	TdD (WRf)	Data definit time prior to $\overline{WR}$ fall	10	—	−10	—	−55	—	ns
32	TdCr (WRf)	Delay from clock rise to $\overline{WR}$ = "L"	—	250	—	65	—	60	ns
33	TdWRr (D)	Output data holding after $\overline{WR}$ = "H"	200	—	60	—	30	—	ns
34	TdCf (HALT)	Delay from clock fall to HALT = "L" or "H"	—	1000	—	300	—	260	ns
35	TwNMI	$\overline{NMI}$ pulse width	300	—	80	—	70	—	ns
36	TsBUSREQ (Cr)	Set-up time for clock rise	250	—	50	—	50	—	ns
37 *	ThBUSREQ (Cr)	$\overline{BUSREQ}$ hold time after clock rise	10	—	10	—	10	—	ns

170489

(3/4)

No.	SYMBOL	ITEM	01F (V_{CC} = 2.7V~5.5V fCLK = 1MHz)		01F (fCLK = 4MHz)		02AF-6 (fCLK = 6MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
38	TdCr (BUSACKf)	Time from clock rise to $\overline{BUSACK}$ = "L"	—	400	—	100	—	90	ns
39	TdCf (BUSACKr)	Time from clock fall to $\overline{BUSACK}$ = "H"	—	400	—	100	—	90	ns
40	TdCr (Dz)	Delay from clock rise to data bus float state	—	350	—	90	—	80	ns
41	TdCr (CTz)	Delay from clock rise to control output float state ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$)	—	300	—	80	—	70	ns
42	TdCr (Az)	Delay from clock rise to address bus float state	—	350	—	90	—	80	ns
43	TdCr (A)	Address holding time from $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ or $\overline{WR}$	200	—	80	—	35	—	ns
44	TsRESET (Cr)	$\overline{RESET}$ set-up time for clock rise	300	—	60	—	60	—	ns
45 *	ThRESET (Cr)	$\overline{RESET}$ hold time for clock rise	10	—	10	—	10	—	ns
46	TsINTf (Cr)	$\overline{INT}$ set-up time for clock rise	300	—	80	—	70	—	ns
47 *	TsINTr (Cr)	$\overline{INT}$ hold time after clock rise	10	—	10	—	10	—	ns
48 *	TdM1f (IORQf)	$\overline{M1}$ output ("L") definite time prior to $\overline{IORQ}$ fall	2000	—	565	—	365	—	ns
49	TdCf (IORQf)	Delay from clock fall to $\overline{IORQ}$ = "L"	—	300	—	85	—	70	ns
50	TdCr (IORQr)	Delay from clock rise to $\overline{IORQ}$ = "H"	—	300	—	85	—	70	ns
51	TdCf (D)	Delay from clock fall to data output	—	550	—	150	—	150	ns

170489

(4/4)

No.	SYMBOL	ITEM	01F (V_{CC} = 2.7V~5.5V fCLK = 1MHz)		01F (fCLK = 4MHz)		02AF-6 (fCLK = 6MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
52	TRST1S	Clock (CLK) restart time by $\overline{INT}$ (STOP mode)	—	(**) (2^{14} + 2.5) × TcC	—	(**) (2^{14} + 2.5) × TcC	—	(**) (2^{14} + 2.5) × TcC	ns
53	TRST2S	Clock (CLK) restart time by $\overline{NMI}$ (STOP mode)	—	(**) (2^{14} + 2.5) × TcC	—	(**) (2^{14} + 2.5) × TcC	—	(**) (2^{14} + 2.5) × TcC	ns
54	TRST1I	Clock (CLK) restart time by $\overline{INT}$ (IDLE1/2 mode)	—	(**) 2.5 TcC	—	(**) 2.5 TcC	—	(**) 2.5 TcC	ns
55	TRST2I	Clock (CLK) restart time by $\overline{NMI}$ (IDLE1/2 mode)	—	(**) 2.5 TcC	—	(**) 2.5 TcC	—	(**) 2.5 TcC	ns

170489

Note 1 : Test conditions (1) @VCC = 5V ± 10%
VIH = 2.4V, VIL = 0.4V, VIHC = VCC − 0.6V, VILC = 0.6V, VOH = 2.2V, VOL = 0.8V
Test conditions (2) TMPZ84C01F @Low Voltage Operation
VIH = VCC-0.2V, VIL = 0.2V, VIHC = VCC-0.2V, VILC = 0.2V, VOH = VCC/2V, VOL = VCC/2V

Note 2 : Items with an asterisk (*) are non-compatible with NMOS Z80.

Note 3 : **TYPICAL

4.4 CAPACITANCE

TA = 25°C

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	CLOCK input capacitance	f = 1MHz	—	—	8	pF
CIN	input capacitance	All pins except measured	—	—	6	pF
COUT	output capacitance	pin are connected to GND	—	—	10	pF

070489

4.5 TIMING DIAGRAM

Figures 4.1 to 4.10 show the basic timings of respective operations. Numbers shown in the figures correspond with those in the AC Electrical Characteristics Table in 4.3.

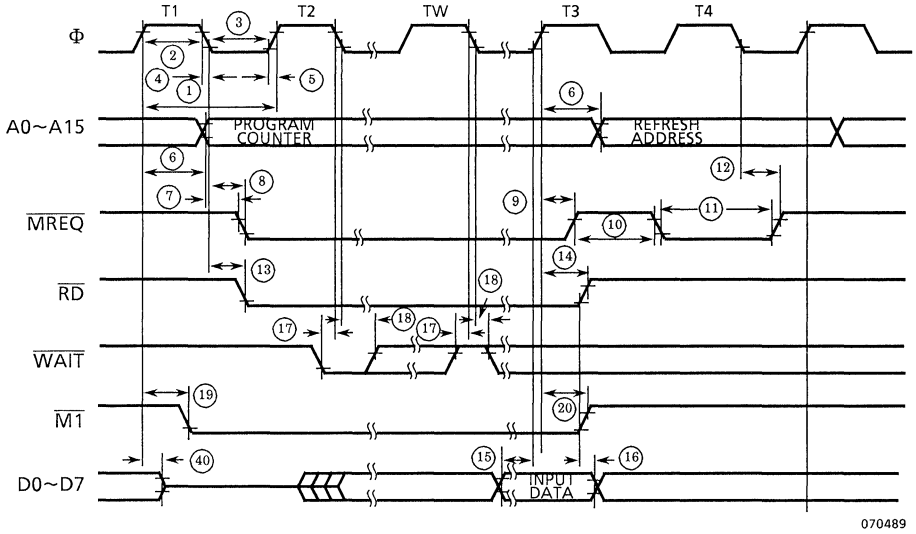


Figure 4.1 Operation Code Fetch Cycle

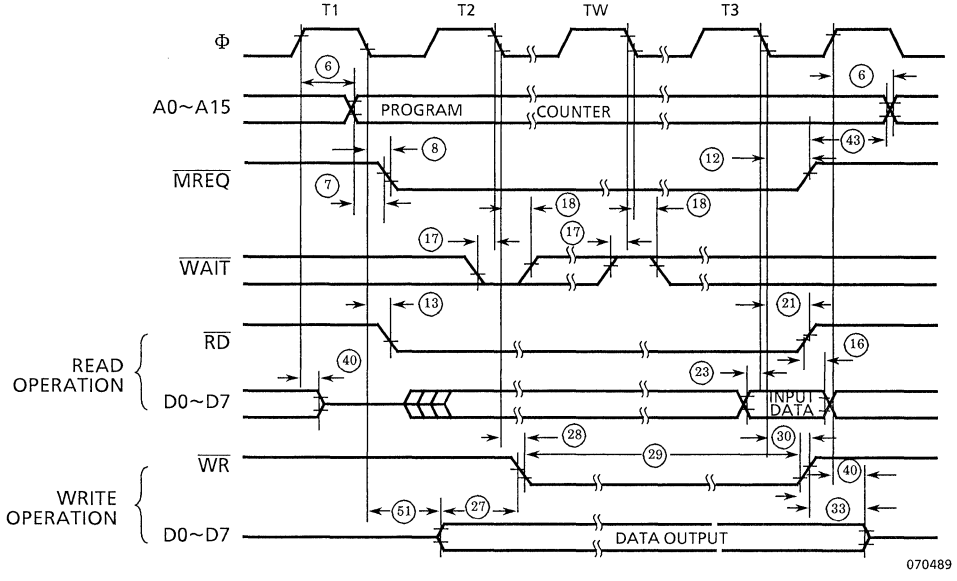
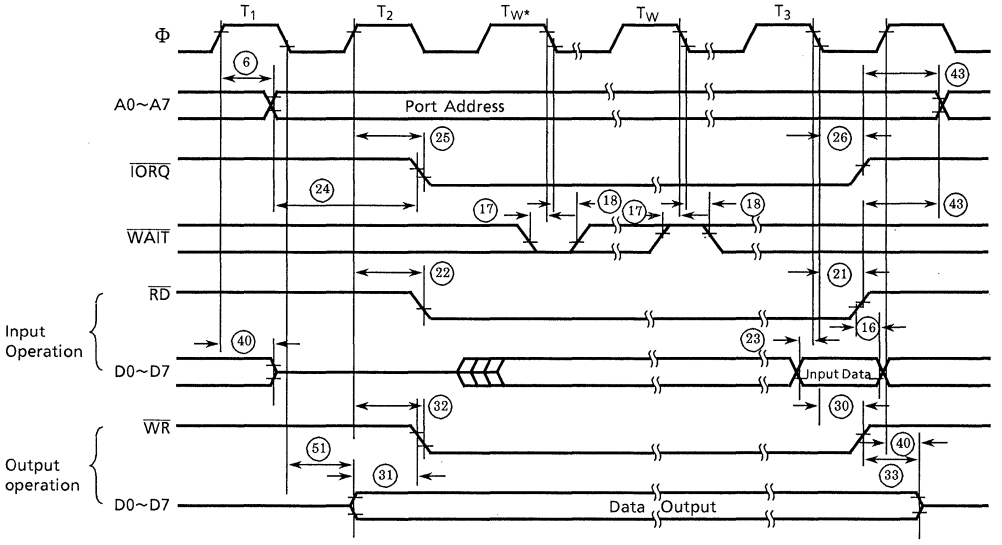


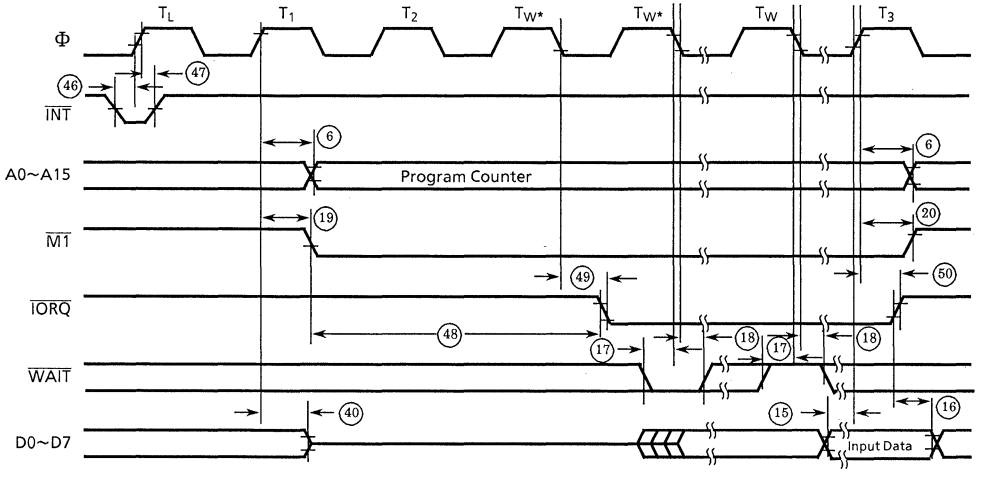
Figure 4.2 Memory Read/Write Cycle



Note : 1 wait state (TW*) is inserted automatically by MPU.

070489

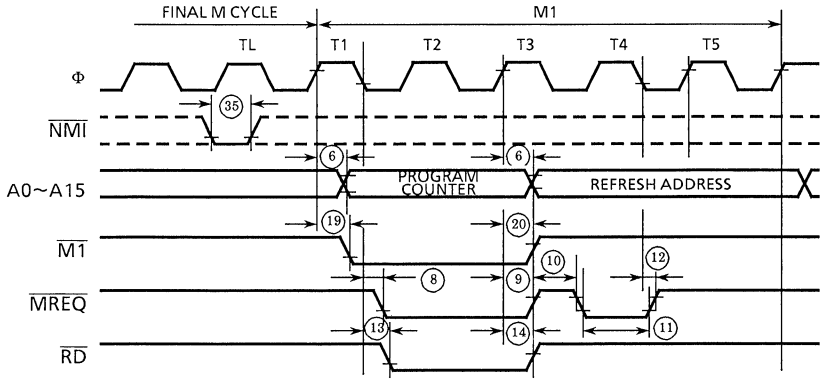
Figure 4.3 Input/Output Cycle



Note 1 TL is the final state of the preceding instruction.
Note 2 2 wait state (TW*) is inserted automatically by MPU.

070489

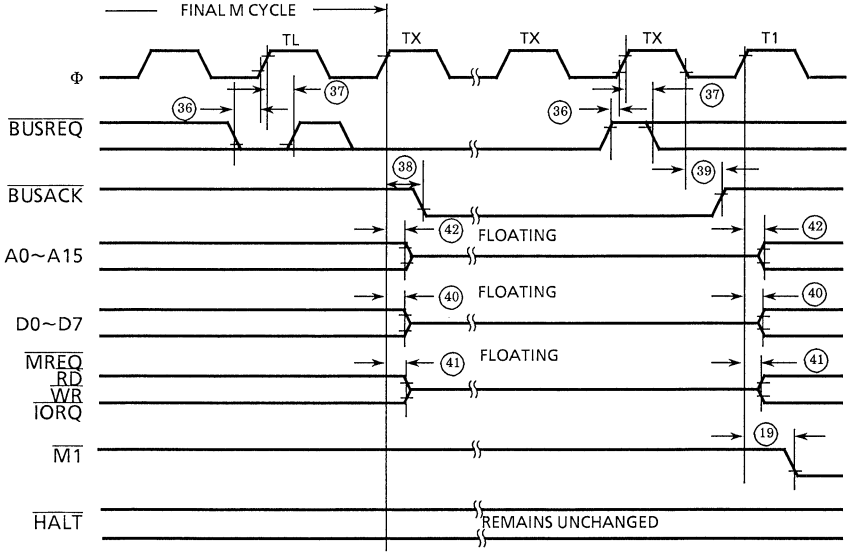
Figure 4.4 Interrupt Request/Acknowledge Cycle



Note : NMI is asynchronous input but in order to assure the positive response in the following cycle, NMI triling edge signal cust be generated keeping abreast of the leading edge of the preceding TL state.

070489

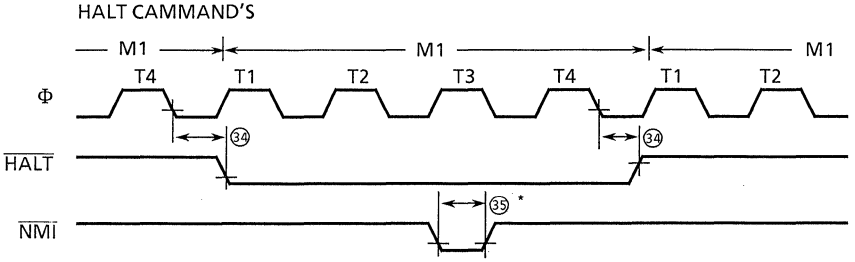
Figure 4.5 Non-maskable Interrupt Request Cycle



Note :
1. TL is the final state of any machine cycle.
2. TX is optional clock used bv requested peripheral LSI.

070489

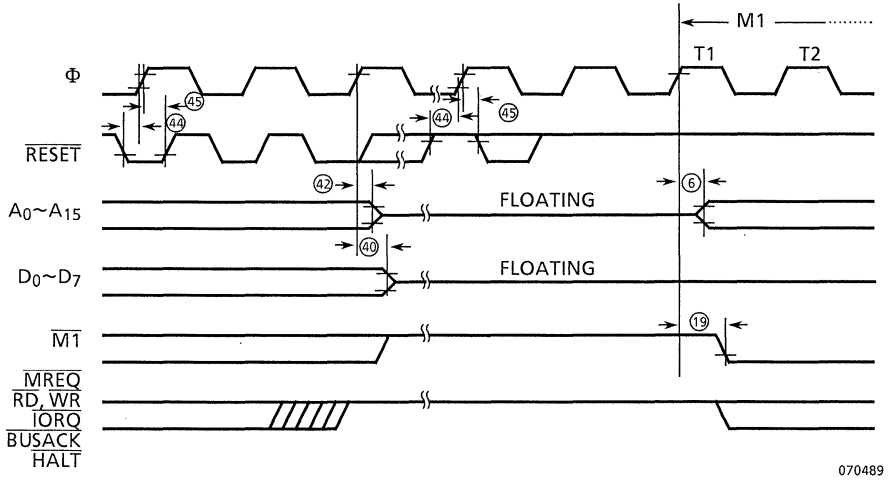
Figure 4.6 Bus Repuest/Acknowledge Cycle



Note : INT signal is also used for releasing from the halt state.

070489

Figure 4.7 Halt Acknowledge Cycle



070489

Figure 4.8 Reset Cycle

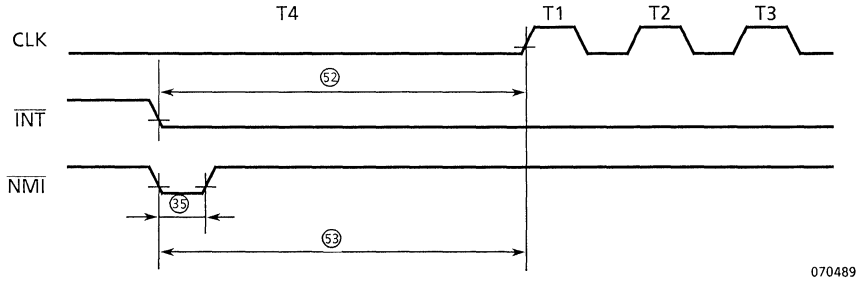


Figure 4.9 Clock Restart Timing (STOP Mode)

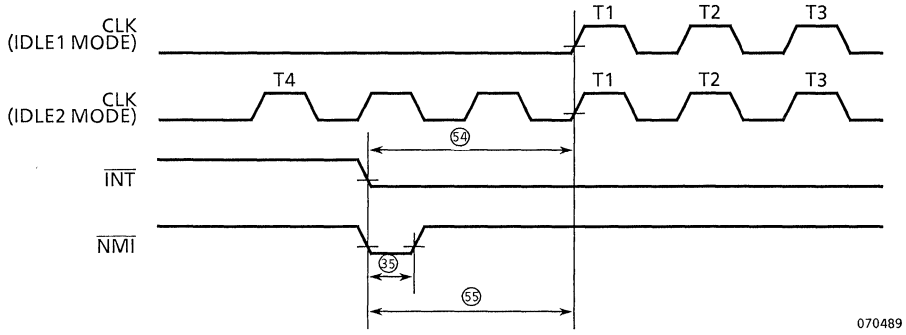


Figure 4.10 Clock Restart Timing (IDLE1/2 Mode)

5. PRECAUTIONS

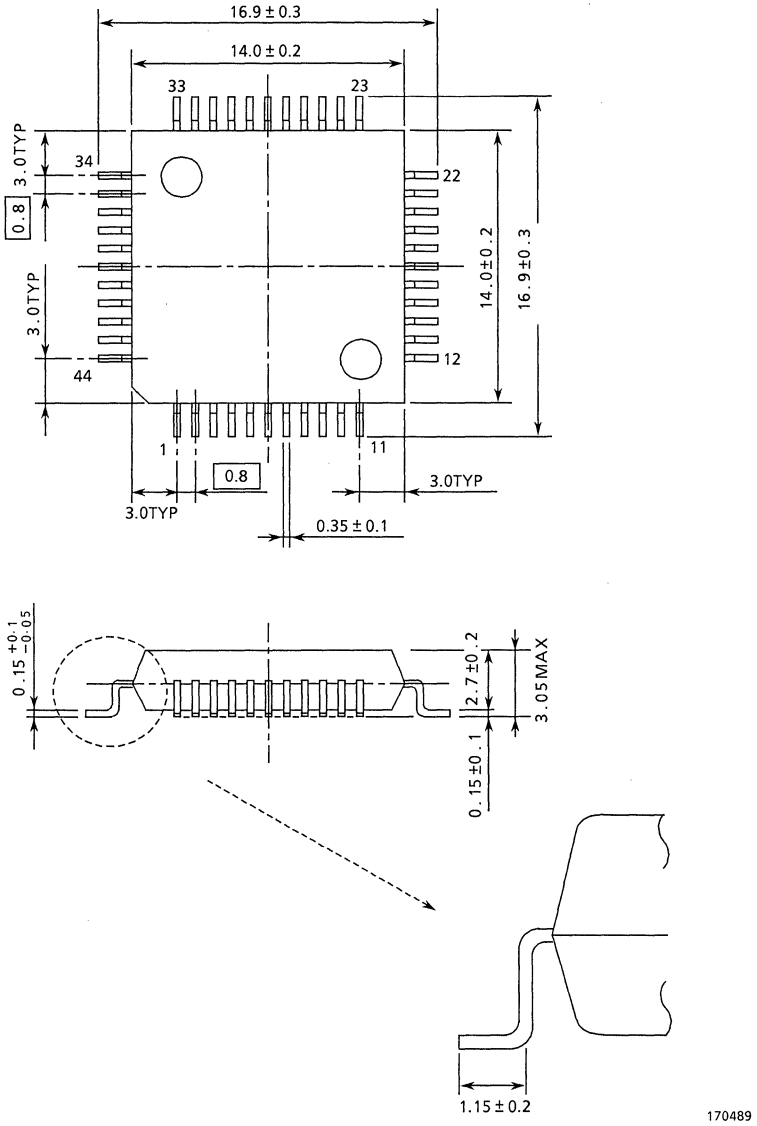
(1) To reset MPU, it is necessary to hold $\overline{\text{RESET}}$ signal input at "0" level for ; least 3 clocks.
In particular, to release the HALT state by $\overline{\text{RESET}}$ signal in STOP Mode, hold $\overline{\text{RESET}}$ signal at "0"level for sufficient time in order to stabilize output from the internal oscillator.

(2) In releasing MPU from the HALT state by interrupt signal in IDLE1/2 Mode and STOP Mode, MPU will not be released from the HALT state and the internal system clock will stop again unless an interrupt signal is accepted during the execution of NOP instruction even when the internal system clock is restarted by the interrupt signal input. In particular, care must be taken when $\overline{\text{INT}}$ is used.
Other precautions are identical to those for the TMPZ84C00A except those for $\overline{\text{RFSH}}$ terminal. Refer to the data sheet for the TMPZ84C00A.

6. OUTLINE DRAWING

QFP44-P-1414F

Unit : mm



Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

TMPZ84C10AP-6 / TMPZ84C10AM-6 / TMPZ84C10AT-6
CMOS-Z80 DMA : DIRECT MEMORY ACCESS CONTROLLER

1. GENERAL DESCRIPTION AND FEATURES

TMPZ84C10A (hereinafter referred to as DMA) is the CMOS Z80 DMA (Direct Memory Access Controller) which provides low power consuming but powerful and versatile operations.

This DMA is designed to improve system performance by allowing the system memory and peripheral LSI's to directly transfer data between them.
Memory-to-memory and I/O-to-I/O (I/O devices as peripheral LSI or I/O devices such as printer, etc.) data transfer capability is also provided.

The TMPZ84C10A is fablicated using Toshiba's CMOS Silicon Gate Technology. The principal functions and features of the TMPZ84C10A are as follows.

(1) Compatible with the Zilog Z80 DMA.

(2) DC to 6MHz operatio

(3) Single 5V power supply (at 5V ± 10%)

(4) Data transfer rate 3M bytes/sec (at 6MHz)

(5) Data transfer in max. 64K byte block length.

(6) Address generation with incrementing, decrementing, or fixed address by source and destination.

(7) Built-in daisy chain structure interrupt circuit.

(8) Low power consumption
10μA MAX (5V, stand-by)

(9) Extended operating temperarure
−40℃ to 85℃

(10) Transfer, search, or transfer/search operations can be specified.

(11) Byte, burst or continuous modes can be specified.

(12) Bit maskable byte searching function.

(13) Built-in Reset logic that is synchronized external signal, Software and when powered on.

Further, in the following text and explanations for charts and tables, hexadecimal numbers are directly used without giving an identification to explanation of address, etc. to the extent not to cause confunctions.

Note : Z80 is a trademark of Zilog Inc., U.S.A.

2. PIN ASSIGNMENTS AND PIN FUNCTIONS

The pin assignments and I/O pin names and brief functions of the TMPZ84C10A are shown below.

2.1 PIN ASSIGNMENTS

The pin assignments of the TMPZ80C10A are as shown in Figure 2.1, Figure 2.2.

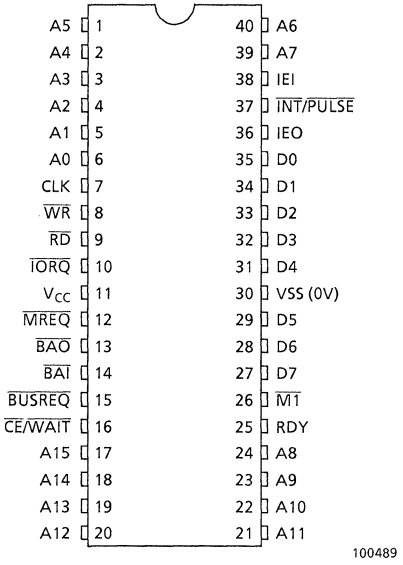


Figure 2.1 DIP, SOP Pin Assignments

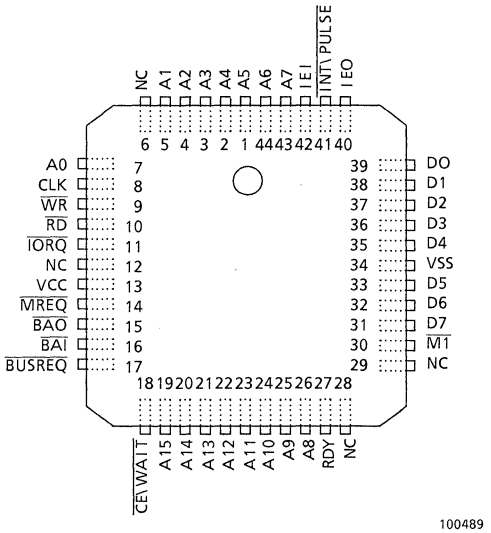


Figure 2.2 PLCC Pin Assignments

2.2 PIN NAMES AND FUNCTIONS

Table 2.1 Pin Names and Functions (1/2)

Pin Name	Number of Pin	Input/Output 3-state	Function
A0~A15	16	Output 3-state	16-bit address bus. DMA output address bus to source port and destination port.
CLK	1	Input	Single phase clock signal. Clock input to DMA. The same clock as that for MPU can be used.
$\overline{WR}$	1	I/O 3-state	Write signal. When used as input, MPU writes to DMA control register. When used as output, DMA controls write to the memory or I/O port address.
$\overline{RD}$	1	I/O 3-state	Read signal. When used as input, MPU reads out of DMA status register. When used as output, DMA controls read from the memory or I/O port address.

00489

Table 2.1 Pin Names and Functions (2/2)

Pin Name	Number of Pin	Input/Output 3-state	Functions
$\overline{IORQ}$	1	I/O 3-state	I/O request signal. When I/O data is read or written, DMA controls read/write.
$\overline{MREQ}$	1	Output 3-state	Memory request signal. When memory data is read or written, DMA controls read/write.
$\overline{BAO}$	1	Output	Bus line enable output signal. In the several DMA configuration, controls priority for the bus using right.
$\overline{BAI}$	1	Input	Bus line enable input signal. Indicates that the system bus using right is released for DMA control.
$\overline{BUSREQ}$	1	I/O	Bus line enable input signal. Indicates that the system bus control are sent to MPU. Open drain.
$\overline{CE}/\overline{WAIT}$	1	Input	Chip enable/wait signal. Normally, operates as $\overline{CE}$ but it is possible to program to operate as $\overline{WAIT}$ at time of data transfer.
RDY	1	Input	Ready signal. Monitored by DMA to determine effective polarity. Effective polarity is programmable.
$\overline{M1}$	1	Input	Signal showing machine cycle 1. Indicates that MPU is in the operation code fetch cycle or interruption acknowledge.
D0-D7	8	I/O 3-state	8-bit bidirectional data bus. Control byte from MPU, status byte from DMA and data from the memory or I/O are transferred through these terminals.
IEO	1	Output	Interrupt enable output signal. Using jointly with IEI, forms the daisy chain structure for interrupt priority when several peripheral LSI's are connected.
IEI	1	Input	Interrupt enable input signal. Using jointly with IEO, forms the daisy chain structure for interrupt priority when several peripheral LSI's are connected.
$\overline{INT}/\overline{PULSE}$	1	Output	Interrupt request signal. For interrupt request and pulse generation. Open drain.
VCC	1	Power supply	+ 5V
VSS	1	Power supply	0V

100489

3. DESCRIPTION OF OPERATION

3.1 BLOCK DIAGRAM

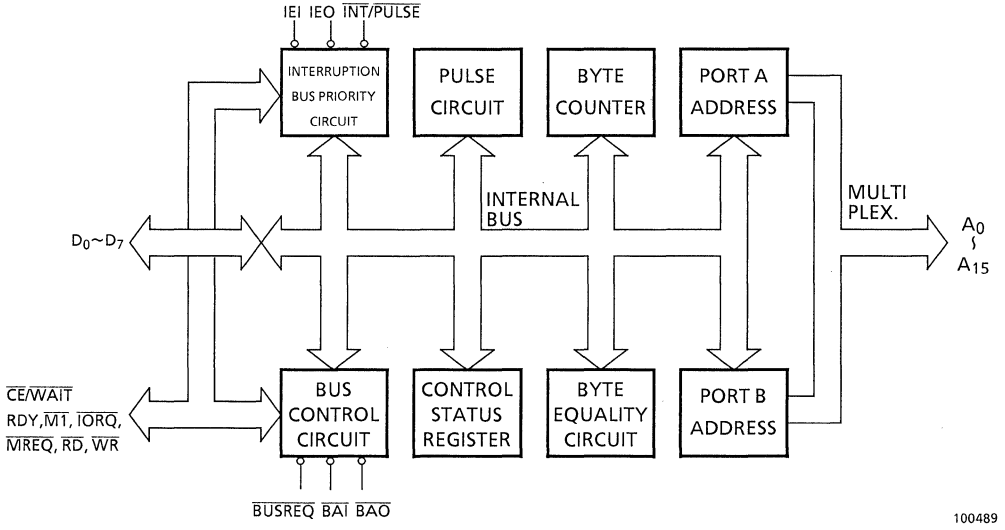


Figure 3.1 Block Diagram

3.2 SYSTEM CONFIGURATION

The architecture (system configuration and functions) which must be known in using DMA will be described here.

Components

(1) Reset circuit
(2) Control & status register group
(3) Bus control circuit
(4) Pulse circuit
(5) Byte counter
(6) Byte equality circuit
(7) Port A address, Port B address
(8) Interrupt priority circuit
(9) Basic function
(10) Interrupts

3.2.1 Reset logic

DMA has following three reset functions.

(1) Power on reset

DMA has reset circuit that is synchronized automaticaly when powered on.

(2) External signal reset

When $\overline{M1}$ signal is active more than two system clocks without an active $\overline{RD}$ or $\overline{IORQ}$ signal, the DMA is reset at rising edge of $\overline{M1}$ signal.

(3) Soft ware reset

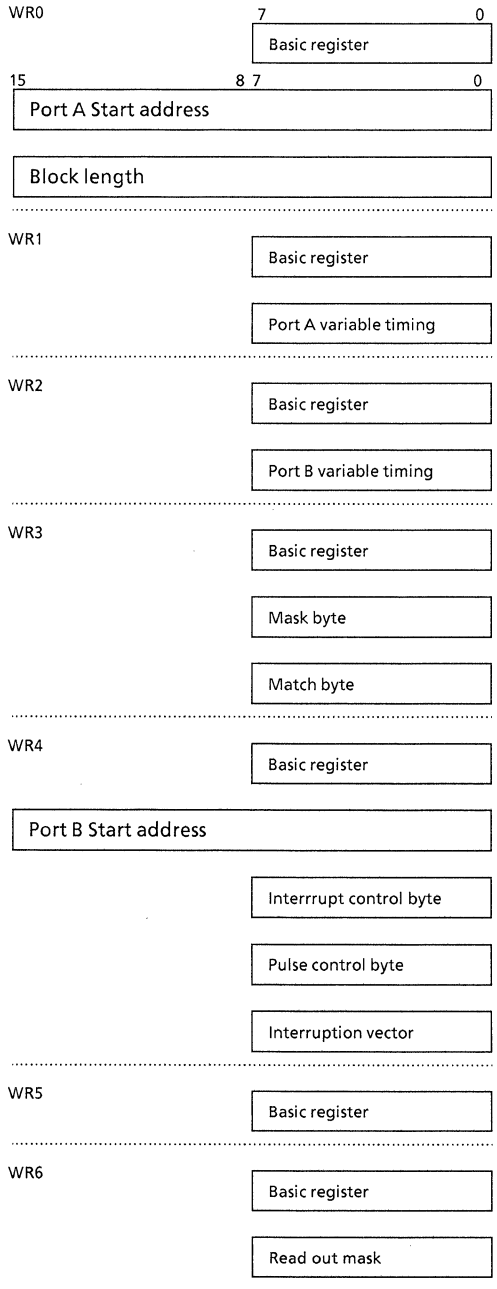
3.2.2 Control & status register group

The DMA is provided with 21 writable register control register group and 7 readable register status register group. Registers are all in 8 bits but 2 byte data is held in optional 2 continued registers. The Z80 Microprocessor (hereinafter referred to as MPU) is capable of setting and monitoring values in respective registers.

The control register group is classfied into 7 groups of WR0 to WR6 (Figure 3.2), each of which is consisting of the basic register and related registers. The operation of DMA is controlled by programming in the control register group.

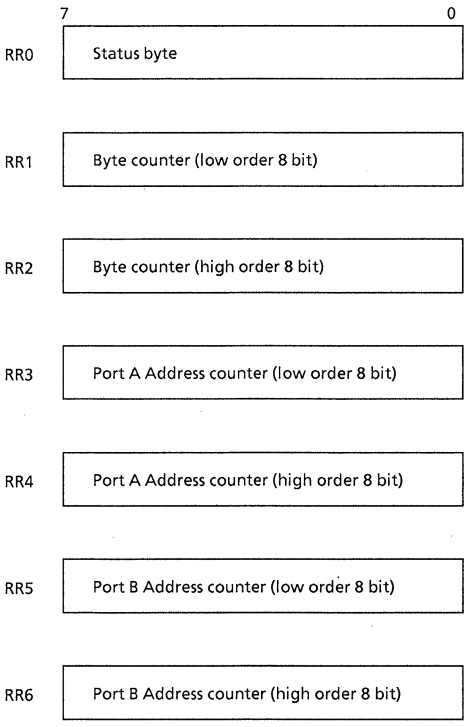
The status regiser group consists of PR0 to PR6 (Figure 3.3) and are used to know state of execution or end of DMA operation.

Further, these registers are described in detail in 3.4 Commands.



100489

Figure 3.2 Control Register



100489

Figure 3.3 Status Register

3.2.3 Bus control circuit

The bus control circuit controls bus direction between DMA and system bus at time of programming, while controls the control bus at time to data transfer according to the data transfer direction between the memory and I/O devices. In addition, it controls updating of required address counter and byte counter.

(1) Bus direction control

- At time of programming, the bus master is MPU and the control bus and data bus (when data is written into the control regiser) are in the direction from the system bus to DMA. Further, when data (status, etc.) is read from the control register, the direction will be from DMA to the system bus. At this time, the address bus buffer is disabled.
- At time data transfer, the bus master is DMA. The control bus buffer and address bus buffer are enabled, and the bus direction will become from DMA to the system bus.
- When data is read out of the memory or I/O device, the data bus direction is from the system bus to DMA but it becomes in the direction from DMA to the system bus at time of data write.

(2) Bus request

If DMA requested MPU to transfer the bus control right and received it, MPU cannot fetch commands from the memory and is placed in the completely idle state.

For DMA to request the bus control right to MPU, following 2 enable conditions are required:

(a) Enable command from MPU

(b) Active RDY condition

3.2.4 Pulse circuit

The pulse circuit generates pulse signals in the INT line for every 256 bytes of 0 to 255 when data transfer is started. The details are described in 3.3.2 (2) (k).

3.2.5 Byte counter

The byte counter is cleared when data is transferred and is incremented by one whenever data is transferred for every 1 byte. A value of this counter is always compared with block length of WR0 and when they agree with each other, the DMA operation ends.

3.2.6 Byte equality circuit

DMA always monitors data being transferred during the data teansfer and when equality is detected, generation of interruption becomes possible.

3.2.7 Port A address, Port B address

Data transfer is performed between Port A and Port B. Either port is specified by the source and destination specified by WR0.

3.2.8 Interrupt priority circuit

The Z80 system used the daisy chain structure to control interrupt among peripheral LSI's and the bus priority among multiple DMA's.
Further, for the interrupt timing, refer to 3.2.2 (2) (j).

(1) Interrupt daisy chain

When the interrupt priority is connected in the daisy chain structure, connect IEI and IEO. When the interrupt is acknowledged, the interrupt configuration of MPU is disabled. In order to allow other peripheral LSI to make the interrupt into MPU, it is necessary to enable the MPU's interrupt configuration by the interrupt enable command. The interrupt enable command is normally executed in the service routine. When the interrupt enable command is executed in the early part of the service routine, a peripheral LSI with higher priority can make the interrupt even when MPU is executing the service routine. (Interrupt in the nest structure is authorized.)

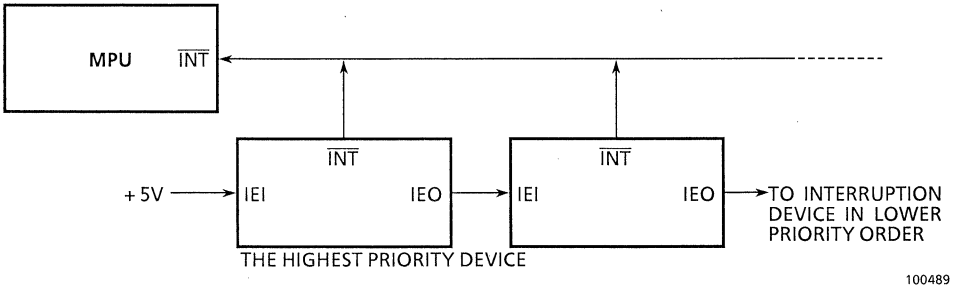


Figure 3.4 Interruption Daisay Chain

(2) Bus request daisy chain

When multiple DMA's are used, priority can be controlled by the daisy chain structure connection. Since $\overline{\text{BUSREQ}}$ signal of each DMA is bidirectional type, each DMA in the daisy chain is able to know bus requests as an input and until a DMA having the bus completes its operation, the bus requests of other DMA's are kept in wait state. Until completion of the operation, any DMA is not able to release the bus in operation by force. Further, the bus request daisy chain has no nesting function but is able to hold the bus until its process is completed. The priority among DMA 's in the daisy chain is in order from high order to low order corresponding to distances drom MPU. Priority is so decided that low order DMA will not receive $\overline{\text{BUSACK}}$ signal through $\overline{\text{BAI}}/\overline{\text{BAO}}$ chain of DMA when multilpe DMA's made the bus request in the same clock cycle period.

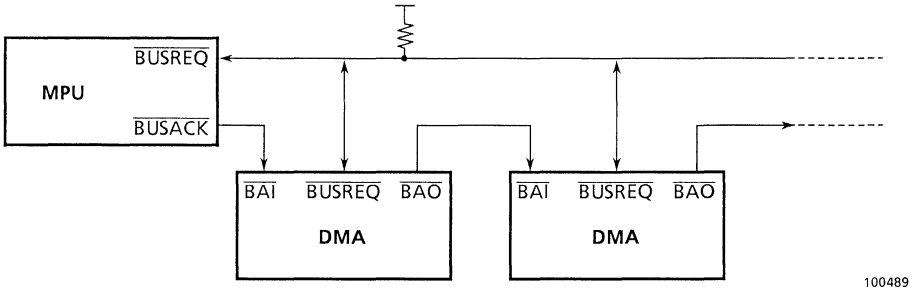


Figure 3.5 BUS REQUEST Daisy Chain

3.2.9 Basic functions

DMA is provided with the following basic functions:

(1) Data transfer paths

① Transfer between memory and I/O
② Transfer between memories
③ Transfer between I/O and I/O
④ Memory search
⑤ I/O search

(2) Operating classes

(3) Operation modes

① Byte mode
② Burst mode
③ Continuous mode

(4) Transfer speed

(5) Operating conditions

(6) Automatic restart

(7) Variable cycle

(8) Pulse generation

3.2.9.1 Data transfer paths

The data transfer paths of DMA are as shown in Figure 3.6.

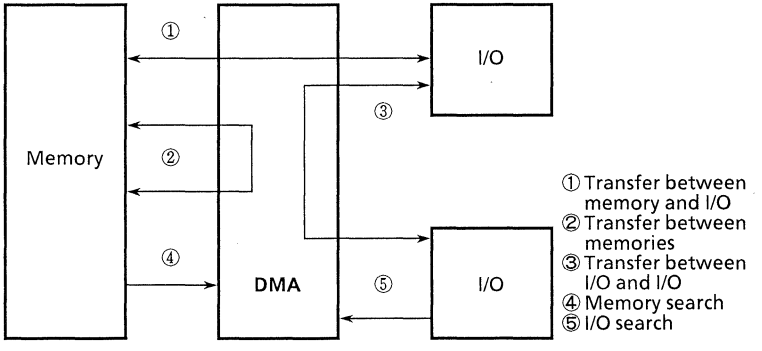


Figure 3.6 Transfer Paths of DMA

(1) Transfer between memory and I/O

This is the most ordinary method of data transfer and data transfer with the high speed serial interfaces (Z80SIO, etc.) is possible.

(2) Transfer between memories

This method of transfer is used for relocation of the memory content and high speed transfer of voluminous data between memories. In addition, this method is used to support memory mapped I/O. It is possible to program to make RDY conditions active for this type of data transfer.

The same function as that of LDIR command (block transfer command) of the Z80 MPU is provided. Number of clocks required by MPU for transfer of data of single byte is 21 clocks in case of the LDIR command while it can be processed in 4 clocks when DMA is used (in case of 2 cycle variable timing). Further, when DMA is used, approx. 420 clocks are required for initialization but in transfering data of 25 bytes or above, DMA becomes advantageous.

(3) Transfer between I/O and I/O

This method of transfer can be used in such applications as acquisition of real time data requiring temporary storage of input data. For instance, in transfering data from a diskette to a line printer, a program only starts the DMA operation and data is transferred from I/O to I/O. However, if I/O error occurred, its recovery becomes necessary.

Further, when there is a byte equality, it is possible to branch into various operations by the search function.

(4) Memory search

This memory search is used to search a large quantity of data at high spped. The same function as that of CPIR command (search command) of the Z80 MPU is provided. Number of clocks required by MPU for single byte memory search is 21 clocks when the CPTR command is used while the search is possible in 2 clocks (in case of 2 cycle variable timing) when DMA is used. Further, approx. 376 clocks are required for initialization when DMA is used and therefore, DMA is advantageous for memory search of more than 19 bytes. In addition, the search of special bytes in the end of block and character check block is also possible.

(5) I/O search

This is used for search of special bytes in the end of block and character check block. For instance, this is used for search of a file mark showing a file delimiter on a magnetic tape.

3.2.9.2 Operating classes

There are 3 kinds of basic operation classes for DMA. 2 out of these 3 kinds are further devided into 2 classes. In addition, the ports referred to here denote the data source and destination.

(1) Data transfer between 2 ports

① Transfer

Data transfer path in the flow of readout cycle followed by write cycle. This is executed without external logic circuit between DMA and MPU.

② Simultaneous transfer

Data transfer path for simultaneous read and write of data transferred between ports by generating required control signal through use of an external logic circuit. 2 times of effeciency of the transfer only class is obtained.

(2) Search of special bit pattern in byte at one port

① Search

This is a method to search special bit pattern by comparing data read from the source port with a matched byte. The matched byte is masked by another byte and can be compared with a special bit pattern (a certain bit in bytes).

(3) Data transfer between 2 ports and search

① Transfer/search

Data transfer is performed in the same transfer method as that of the only transfer class and at the same time, the same search as that for the search only class is performed.

② Simultaneous transfer/search

Data transfer is performed in the same transfer method as that of the simultaneous transfer class and at the same time, the same search as that for the search only class is performed.

In this case, an external logic circuit is required.

3.2.9.3 Operation modes

In the transfer methods for various operation classes, DMA can select the following modes:

- Byte mode
- Burst mode
- Continuous mode

The single byte transfer/search is shown in Figure 3.7 (commonly applicable to all modes).

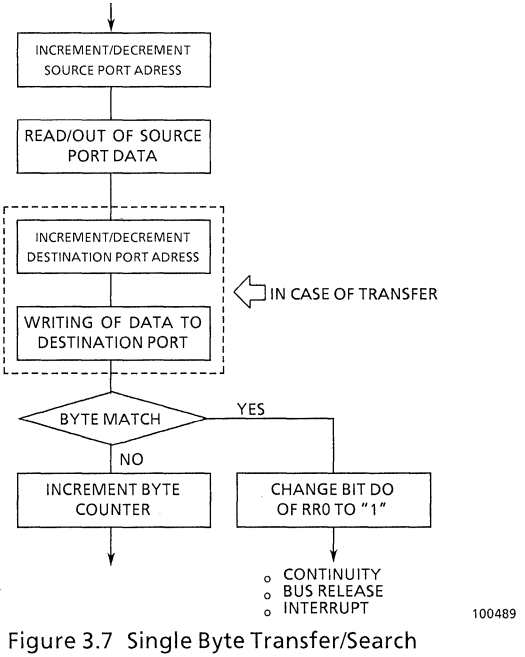


Figure 3.7 Single Byte Transfer/Search

The data transfer is started at the point of time when DMA is enabled. In the first single byte operation in any mode, RDY signal is first checked to determine if it is active.

Then, the bus request is made and single byte transfer/search is performed when DMA becomes the bus master. The same operation is continued until the end of block judgement is made. If it is not the end of block, however, a different operation is carried out after judgement of RDY signal. The operations in respective modes are as shown in Figure 3.8, 3.9 and 3.10.

(1) Byte mode

In the data transfer operation of DMA, the system bus control right is released whenever 1 byte is transferred at a time and the system bus control right is returned to MPU for at least one machine cycle period.

If RDY signal of DMA is active when one machine cycle passes after the system bus control right is returned to MPU, the bus request is made again to MPU and next one byte data transfer is performed. Further, when RDY signal is non-active, the system bus control right is retained by MPU. This operation is shown in Figure 3.8.

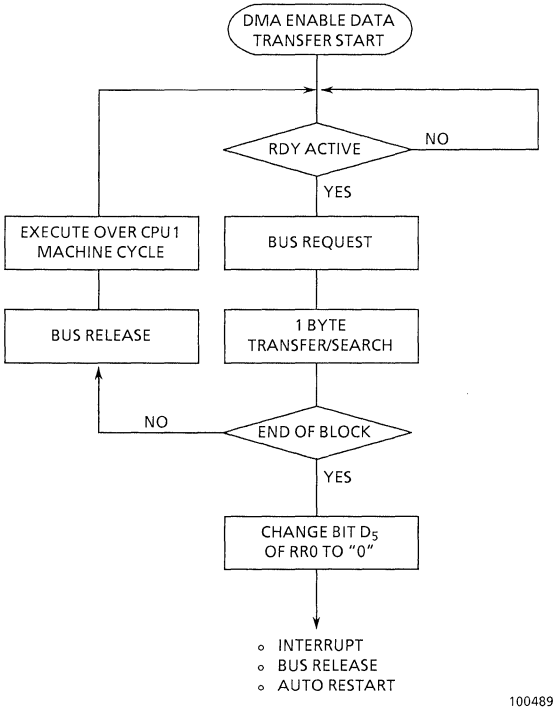


Figure 3.8 Byte Mode

(2) Burst mode

In the burst mode, after one byte data is transferred, RDY signal is checked to determine if it is active without releasing the system bus control right. If RDY signal is active, data transfer is continued until RDY signal becomes non-active and after the data transfer is completed, DMA stops to operate. Since MPU is ready to operate during the period in which I/O device does not transfer data (when RDY signal is non-active), data transfer rate and bus using efficiency are effective.

This operation is shown in Figure 3.9.

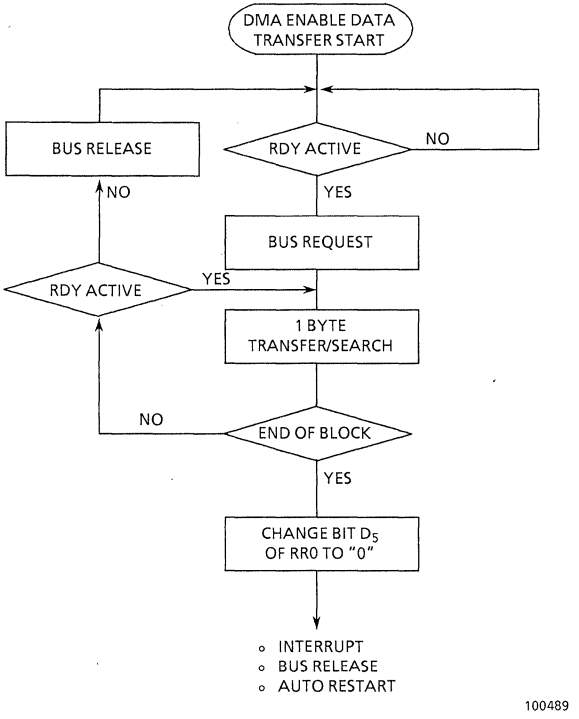


Figure 3.9 Burst Mode

(3) Continuous mode

When the data transfer is commenced, DMA retains the system bus control right until the transfer of last byte of a data block is completed or the stop condition of RDY signal becomes non-active during the operation, DMA is simply put in the idle state and still retains the system bus control right while waiting that RDY signal becomes active again. What must be taken care of is that if number of data bytes is smaller than that set in the byte counter, DMA cannot end the block transfer forever and the system is impeded to operate properly.

This operation is shown in Figure 3.10.

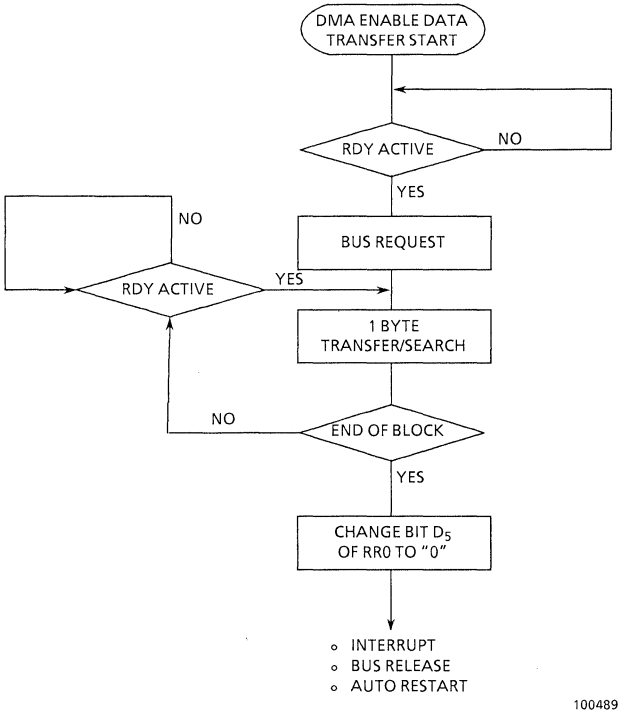


Figure 3.10 Continuous Mode

3.2.9.4 Transfer speed

Shown in Table 3.1 are the comparison of max. transfer rates in 5 transfer classes of DMA operation and that of max. transfer rates in block transfer command of MPU. The max. speed transfer rate is accomplished in the simultaneous transfer operation of DMA and at least one external logic circuit is required. DMA transfers shown in the table are based on the assumption that interruption is not involved in the burst or continuous mode, and that the read and write cycle is 2 cycles.

Table 3.1 Transfer and Search Max. Speed (Burst and continuous modes)

Operation	Z – 80 (6.0MHz)
Simultaneous transfer of DMA DMA search only Simultaneous transfer/search	3M byte/s
DMA transfer DMA transfer/search	1.5M byte/s
Block transfer command of MPU	0.300M byte/s

100489

Shown in Table 3.2 is the comparison of the Z80 throughput reduction rate (per transfer K baud) in the byte mode of data transfer by DMA with the throughput reduction rate in the byte transfer using the interrupt service routine by six commands (actual minimum) by MPU. The DMA transfer in this data is based on the assumption that read and write cycle timing is longer than 2 cycles (min.). Therefore, MPU throughput reduction rate in the 2 cycle simultaneous transfer is further reduced.

Table 3.2 Z-80 MPU Throughput Reduction Per DMA Transfer K Baud (Byte Mode)

Operation	Z – 80 (6.0MHz)
DMA transfer DMA transfer/search	0.027%
MPU transfer by interrupt	0.142%

100489

3.2.9.5 Operating conditions

Programmable conditions to get DMA perform certain operations and these operations are shown in Table 3.3. The conditions referred to here are those conditions for the internal registers of DMA, signals from peripheral LSI's and commands to DMA on the data bus. For details refer to Table 3.3.

Table 3.3 Operating Conditions

Conditions	Operations that can be caused under conditions at left hand
End of block	a. Bus release b. MPU interruption c. Auto restart
Coincidence of byte	a. Bus release b. MPU interruption c. Continuation
Pulse control byte coincided with low order byte of byte counter	a. Pulse generation
RDY signal is active	a. Bus request b. MPU interruption
RDY signal is non-active	a. Bus release b. Breaking (in case of continuous mode)
RETI command (interruption return command from MPU)	a. Bus request

100489

3.2.9.6 Auto restart

In DMA data transfer, it is possible to automatically clear the byte counter, load the content of the start address register on the address counter again and restart the data transfer at the end of block.

The automatic restart function can reduce a burden of software on MPU in the CRT refresh or repeating operation. In addition, it is possible to write different start address into the buffer register during the data transfer (when RDY signal is non-active and the bus is released during the data transfer in the byte mode or burst mode). At this time, it becomes possible to commence the automatic restart of data transfer from a new start address.

3.2.9.7 Variable Cycle

DMA is capable of changing readout and write cycle lengths through programming. This function is effective in increasing data transfer rate and reducing a burden on a software, and an external logic circuit may be omitted. Refer to 3.3.2 (2) (i) where this function is described.

3.2.9.8 Pulse generation

DMA generates pulse signal on the $\overline{INT}$ line for every 256 bytes for data transfer. This is described in detail in 3.3.2 (2) (k).

3.2.10 Interrupt

DMA is able to make an interrupt request to MPU under the following conditions:

- After DMA's RDY signal becomes active and before DMA makes a bus request ($\overline{BUSREQ}$="0")
- When the content of the byte counter coincides with that of the block length register and the end of block is detected.
- When the content of the coincided byte masked by the mask byte coincides with data in the transfer or search period when the byte coincidence is formed.

To make an interrupt request to MPU, it is necessary for DMA to release the bus. If DMA is the bus master, signal on the INT line generates periodic pulses to the peripheral LSI's, which are not sensed by MPU.
Therefore, at the end of block or after stop by byte coincidence, DMA releases the bus before interrupting MPU.

If interrupt at the enf of block and automatic restart at the end of block are set for DMA by programming, an interrupt is taken place at each end of block (at this point of time, it is acknowledged for the continuous operation). If the automatic restart is programmed in this case, the status flag at the end of block is not set. In this case, the interrupt vector cannot determine a factor for that interrupt.

On the Z80 system, interrupt is controlled through the daisy chain system. For the interrupt daisy chain, refer to 3.2.7 Interrupt/Priority Circuit. In addition, for the interrupt timing, refer to 3.3.2 (2) (j).

3.3 Status change flowchart and basic timing

The status change flowchart and the basic data transfer timing by DMA are shown here. The status change flowchart is shown in 3.3.1 and the basic timing in 3.3.2.

3.3.1 Status change flowchart

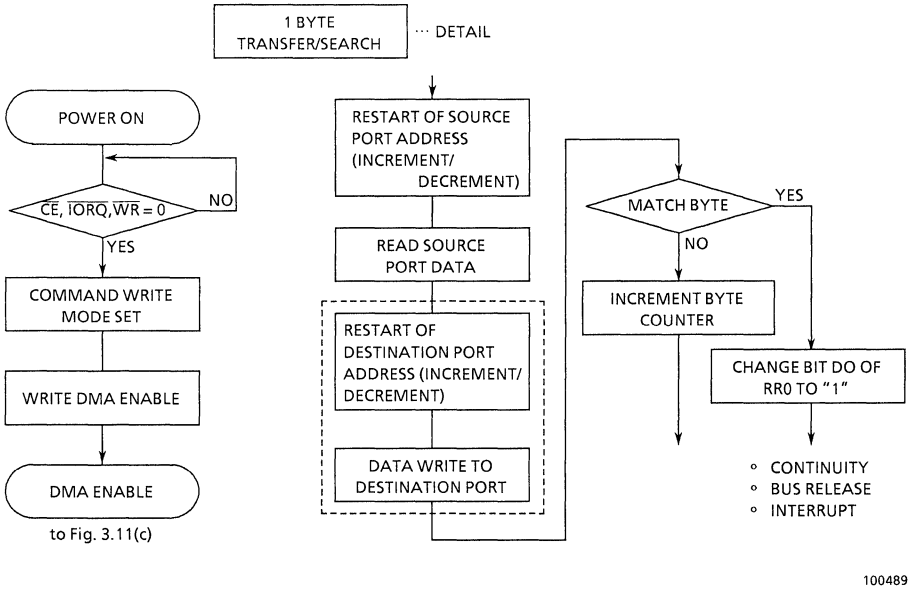
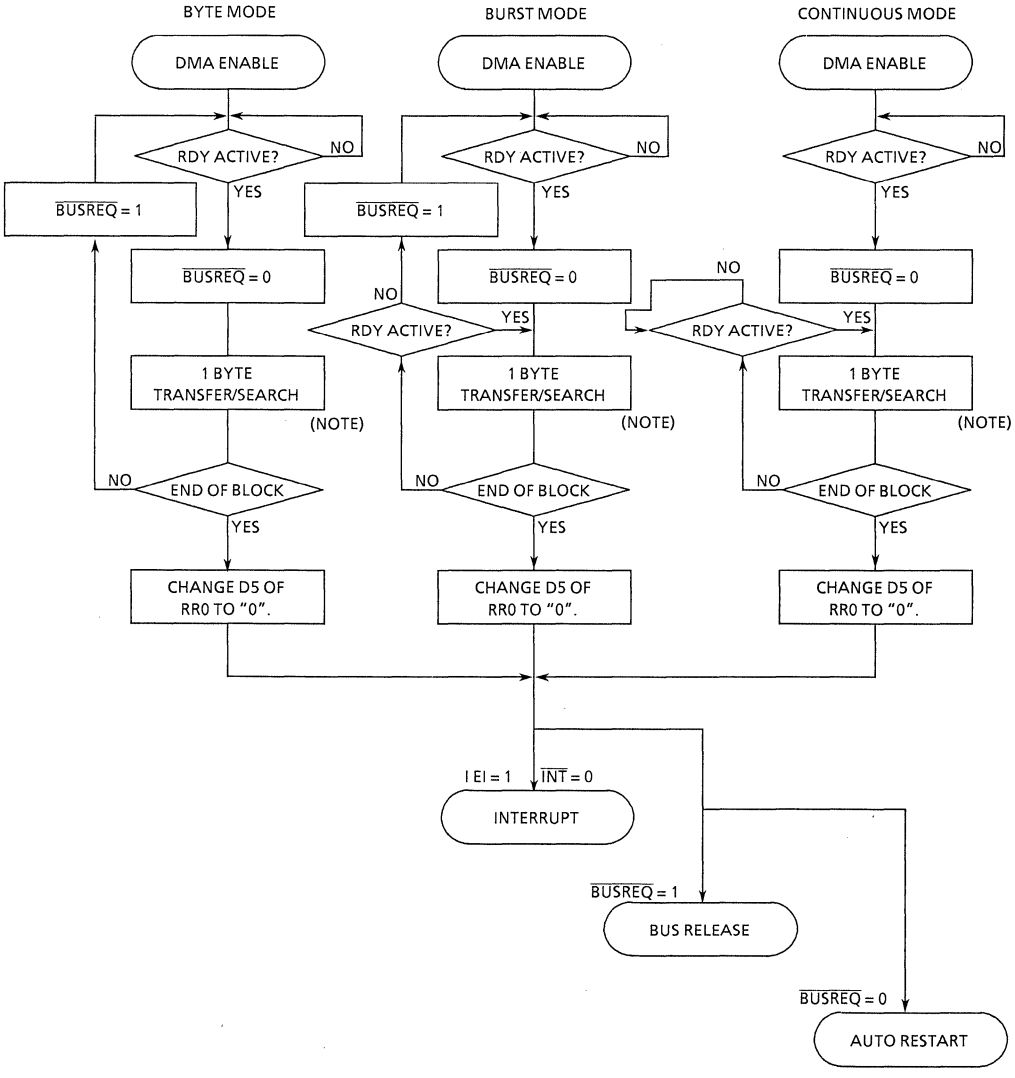


Figure 3.11 (a) Status Change Flowchart Figure 3.11 (b) Status Change Flowchart



Note : The details for signal byte transfer/search is shown in Figure 3.11 (b) Status Change Flowchart.

100489

Figure 3.11 (c) Status Change Flowchart

3.3.2 Basic timing

When DMA receives a command from MPU or reads the readout register, MPU has the system bus control right, $\overline{BUSACK}$ = "1", and MPU is called the bus master. When DMA operation is data transfer by DMA proper, $\overline{BUSACK}$ = "0", and DMA gets the system bus control right and becomes the bus master.

(1) When the bus master is MPU:

(a) Write timing into the write register

To write data into the write register, it is necessary for 3 signals of $\overline{CE}$, $\overline{IORQ}$ and $\overline{WR}$ to become "0" simultaneously at the rising edge of clock. At this leading edge, DMA latches these 3 signals. After latched, $\overline{CE}$, $\overline{IORQ}$ and $\overline{WR}$ signals may change to the invalid level after certain hold time. Further, DMA writes the status of the data bus (D0 to D7) into necessary write registers at the rising edge of next clock.

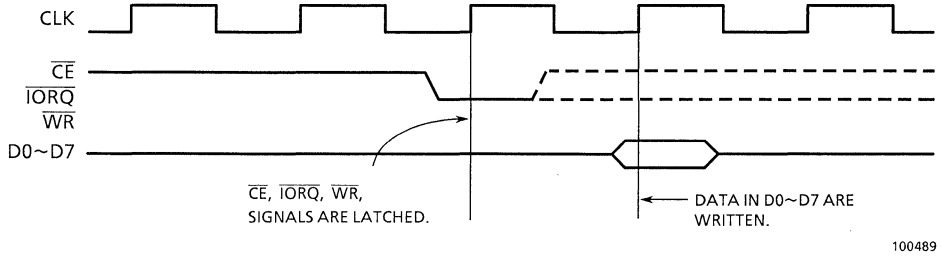


Figure 3.12 Write Timing into the Write Register of DMA

(b) Readout timing from the readout register

To readout the readout register it is necessary that 3 signals of $\overline{CE}$, $\overline{IORQ}$ and $\overline{RD}$ are at "0" and stable for more than 2 clocks. At the rising edges of 2 clocks, the status data is on the data bus and kept as long as $\overline{CE}$, $\overline{IORQ}$ and $\overline{WR}$ signals are active.

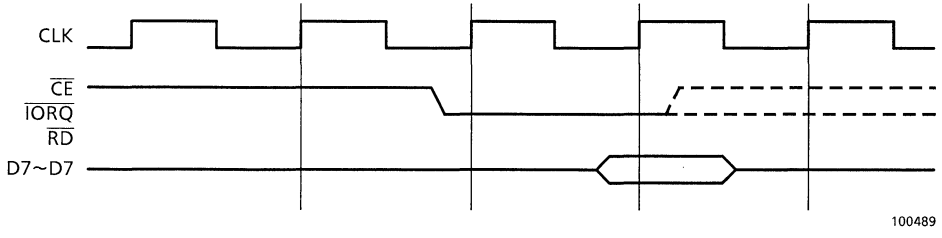


Figure 3.13 Readout Timing from the Readout Register of DMA

(2) When DMA is the bus master:

(a) Transfer

Transfer and transfer/search operations are performed at the same timing. Data is latched at the rising edge of $\overline{RD}$ signal (in case of the standard timing, the falling edge of T3 state) and held on the data bus during next write cycle. After $\overline{RD}$ signal becomes "1", the data bus buffer of DMA is enabled.

The standard timing is 3 clock cycles for the memory operation while it is 4 clock cycles for the I/O operation. In addition, in the I/O operation, the timing is 4 clock cycles including TW* which is automatically inserted between T2 and T3 state.

When $\overline{CE}/\overline{WAIT}$ signal is programmed as "Multiplex" in the write register WR5, DMA samples the status of this signal at the falling edge of T2 in case of the memory readout and at the falling edge of TW* in case of the I/O write. If $\overline{WAIT}$ signal is at "0" level at this time, DMA inserts one clock cycle (Tw) and if it is at "1" level, proceeds to next cycle. Further, when Tw is inserted, $\overline{WAIT}$ signal is sampled again during this period and the same processing is performed.

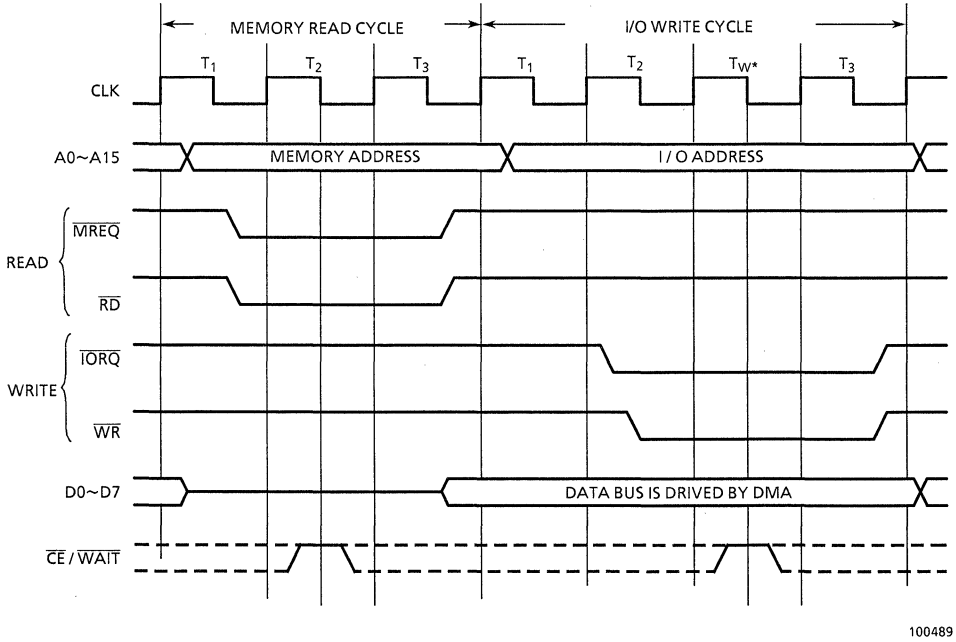


Figure 3.14 Transfer Standard Timing of Memory to I/O

- Memory to I/O

  In the memory readout, DMA put the memory address on the memory bus (A0 to A5) in the period of T1 rise and bring $\overline{MREQ}$ and $\overline{RD}$ signals to "0" level at the falling edge of T1 state. The memory data is read out at this point of time, put on the data bus (D0 to D7), latched by DMA at the falling edge of T3 immediately before the rise of $\overline{RD}$ signal, and when $\overline{RD}$ signal becomes "1" level, DMA data bus buffer is enabled and the latched data is output on the data bus.
  In the I/O write cycle, DMA put I/O address on the addresss bus in the T1 rise period, makes $\overline{IORQ}$ signal and $\overline{WR}$ signal to "0" level in the T2 rise period, and writes the data on the data bus (data readout from the memory) into I/O.

- I/O to memory

  In the I/O readout cycle, DMA put I/O address on the address bus in the T1 rise period and makes $\overline{IORQ}$ signal and $\overline{RD}$ signal to "0" in the T2 fall period. I/O data is read out and placed on the data bus at this time, and is latched by DMA at the trailing edge of T3 immediately before the rise of $\overline{RD}$ signal. When $\overline{RD}$ signal becomes "1" level, DMA data bus buffer is enabled and the latched data is output on the data bus.

  In the memory write cycle, DMA places memory address on the address bus in the T1 fall period, makes $\overline{MREQ}$ signal to "0" level at the falling edge of T1 and $\overline{WR}$ signal to "0" level in the T2 rise period, and write data on the data bus (data readout from I/O) into the memory.

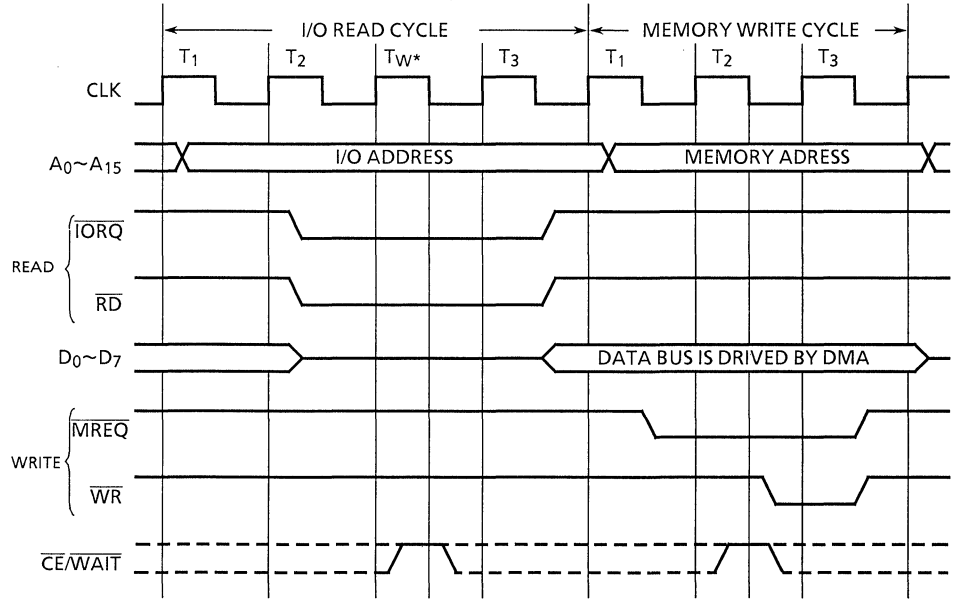


Figure 3.15 Transfer Standard Timing of I/O to Memory

- Memory to memory

  This operation is a combined operation of the memory read cycle and memory write cycle.

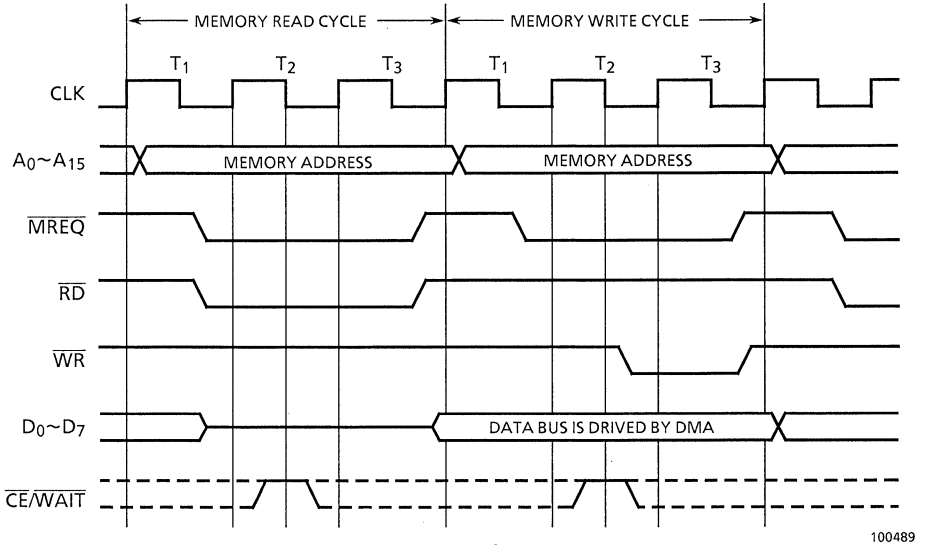


Figure 3.16 Transfer Timing of Memory to Memory

- I/O to I/O

  This operation is a combined operation of the I/O read cycle and I/O write cycle.

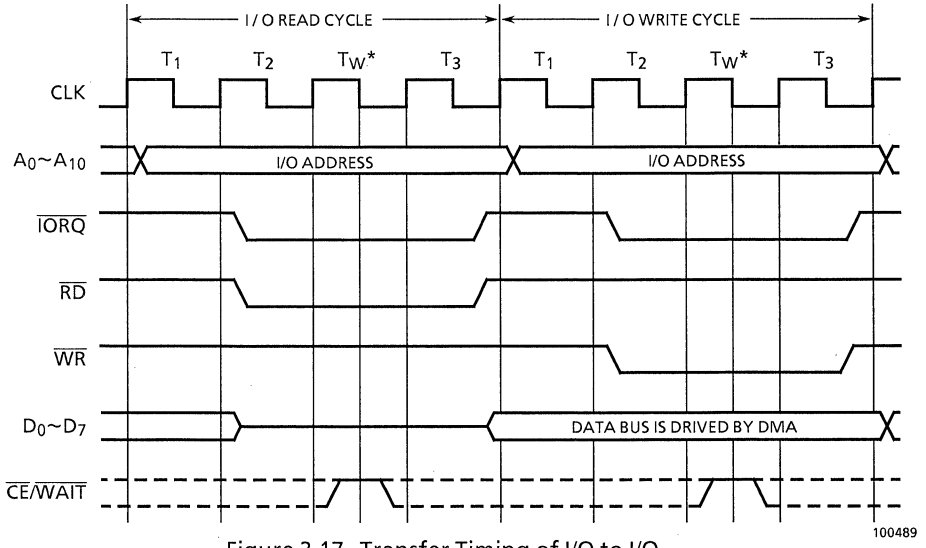


Figure 3.17 Transfer Timing of I/O to I/O

(b) Search timing

The search operation is identical to the readout only operation and data is only read into DMA register for comparison with coincided byte.

The timing of search operation is identical to that of memory to I/O transfer shown in Figure 3.14 and that of I/O to memory transfer in Figure 3.15.

(c) Simultaneous transfer

The simultaneous transfer operation and the simultaneous transfer/search operation are performed in the same timing.

When DMA is programmed in the search only mode, the read and write cycles are generated in one read cycle (source port readout period). Since only one address is generated on the address bus, the memory or I/O control signal is generated using an external logic circuit and DMA operation is performed according to this control signal. In addition, I/O ports are selected by hardware during the operation. Signals with (EXT) shown in Figure 3.18 through Figure 3.21 are those generated by an external logic circuit.

- Memory to I/O (Memory search cycle)

In this data transfer, the memory search mode is programmed and the memory readout and I/O write are performed in one read cycle by generating $\overline{\text{IORQ}}$ signal and $\overline{\text{WR}}$ signal in the memory readout cycle using an external logic circuit. The hardware performs the memory readout by $\overline{\text{MREQ}}$ signal and $\overline{\text{RD}}$ signal that are output by DMA and the I/O write by $\overline{\text{IORQ}}$ signal and $\overline{\text{WD}}$ signal that are generated using an external logic circuit.

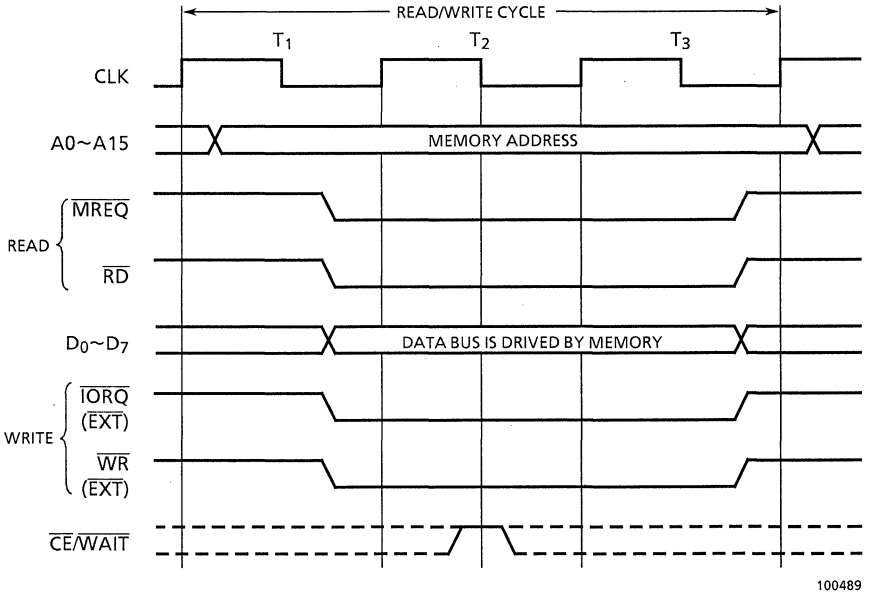


Figure 3.18 Simultaneous Transfer Timing of Memory to I/O (Memory search timing)

- I/O to memory (Memory search cycle)

  In this data transfer, the memory search mode is programmed and the I/O read and memory write operations are performed in one readout cycle by generating $\overline{IORQ}$ signal and $\overline{WR}$ signal in the memory readout cycle using an external logic circuit. The hardware performes the I/O readout using $\overline{RD}$ signal output by DMA and $\overline{IORQ}$ signal generated by an external logic circuit and the memory write using $\overline{MREQ}$ signal generated by DMA and WR signal produced by an external logic circuit.

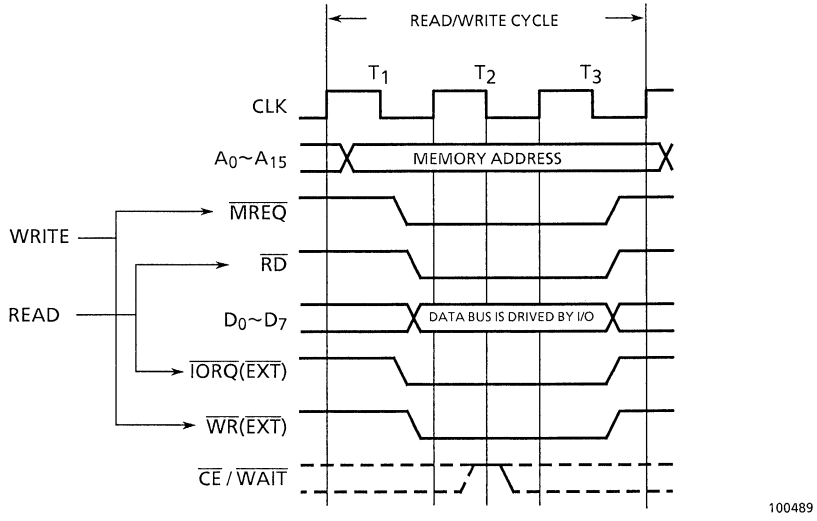


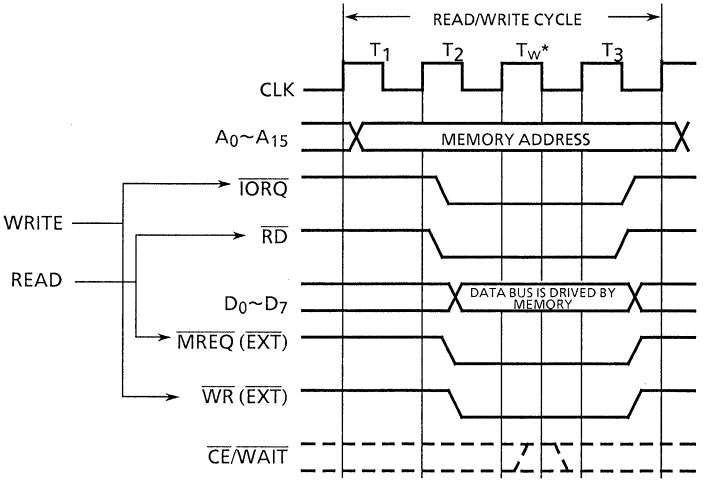
Figure 3.19 Simultaneous Transfer Timing of I/O to Memory (Memory search timing)

- Memory to I/O (I/O search cycle)

  In this data transfer, the I/O search mode is programmed and the memory read and I/O write operations are performed in one readout cycle by generating $\overline{\text{MREQ}}$ signal and $\overline{\text{WR}}$ signal in the I/O readout cycle using an external logic circuit. The hardware performs the memory readout using $\overline{\text{RD}}$ signal output by DMA and $\overline{\text{MREQ}}$ signal generated by an external logic circuit and the I/O write using $\overline{\text{IORQ}}$ signal generated by DMA and $\overline{\text{WR}}$ signal produced by an external logic circuit.

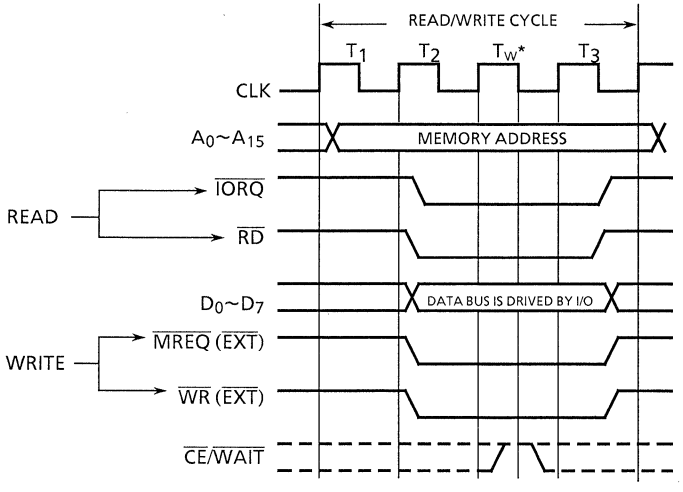
- I/O to memory (I/O search cycle)

  In this data transfer, the I/O search mode is programmed and the I/O read and memory write operations are performed in one readout cycle by generating $\overline{\text{MREQ}}$ signal and $\overline{\text{WR}}$ signal in the I/O readout cycle using an external logic circuit. The hardware performs the I/O readout using $\overline{\text{IORQ}}$ and $\overline{\text{RD}}$ signals output by DMA and the memory write using $\overline{\text{MREQ}}$ signal and $\overline{\text{WR}}$ signal produced by an external logic circuit.



Note : Althrough addresses on A0-A15 are originally I/O addresses, they are handled as memory addresses.

Figure 3.20 Simultaneous Transfer Timing of Memory to I/O (I/O search timing)



Note : Althrough addresses on A0-A15 are originally I/O addresses, they are handled as memory addresses.

Figure 3.21 Simultaneous Transfer Timing of I/O to Memory (I/O search timing)

(d) Bus request timing

When RDY signal becomes active, DMA samples RDY signal at the rising edge of the clock and if the bus is not full ($\overline{BUSREQ}$="1") DMA makes $\overline{BUSREQ}$ signal to "0" level at the rising edge of next clock and request MPU to hand over the system bus control right.

MPU samples $\overline{BUSREQ}$ signal at the rising edge of the last state clock of the machine cycle which MPU is executing at that point of time and if it is "0", makes $\overline{BUSACK}$ signal to "0" level at the rising edge of next clock.

Therefore, maximum value of a time required for MPU to hand over the bus control right to DMA ($\overline{BUSACK}$="0") after DMA detected that RDY signal becomes active is the sum of one machine cycle (valiable) and one clock period of MPU.

When detecting that $\overline{BAI}$ ($\overline{BUSACK}$) signal is at "0" level for 2 clock period, DMA start the DMA action. There is the delay time of max. One machine cycle +3 clock period after RDY signal becomes active till the DMA action is actually started.

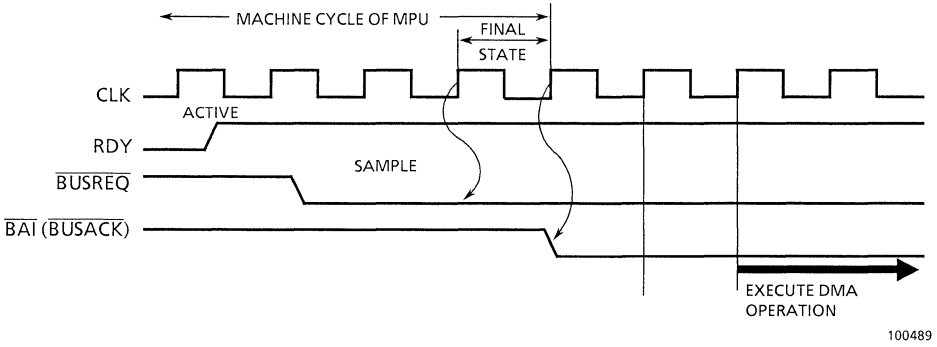


Figure 3.22 Bus Request Timing

(e) Bus release timing - byte mode

In the byte mode, DMA makes $\overline{BUSREQ}$ signal to "1" level at the rising edge of the clock immediately before end of each data transfer cycle (the end of readout cycle in the search operation and the end of write cycle in the transfer and trasfer/search operation.)

Although $\overline{BUSREQ}$ signal becomes "1" before the end of DMA cycle by one clock, MPU resumes the operation one clock after $\overline{BUSREQ}$ signal becomes "1" level and therefore, there will be no trouble.

After the bus is released, next bus request is made at the leading edge of the clock immediately after both $\overline{BUSREQ}$ signal and $\overline{BAI}$ signal becomes "1" level. RDY signal being active is the conditions for this.

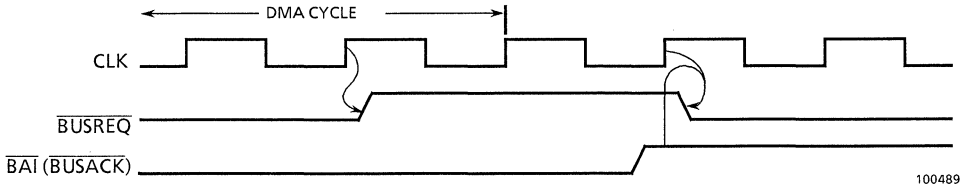


Figure 3.23 Bus Release Timing - Byte Mode

(f) Bus release at the end of block in the burst mode or continuous mode

When it is programmed to stop DMA at the end of block in the burst mode or continuous mode, $\overline{\text{BUSREQ}}$ signal is set to "1" level at the rising edge of the clock at the end of last data transfer. This last data is transferred even when RDY signal becomes non-active.

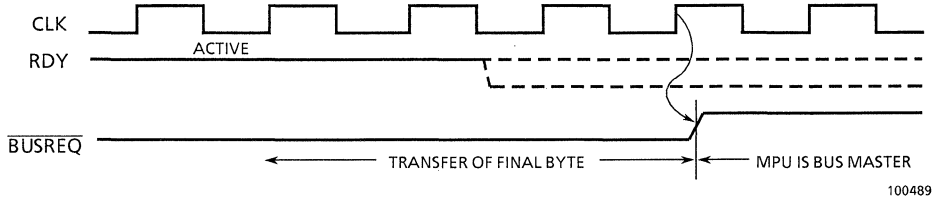


Figure 3.24 Bus Release Timing - at End of Block

(g) Bus release when coincidence is detected in the burst mode or continuous mode

When DMA is set in the burst mode or continuous mode and programmed to stop its operation at byte coincidence, DMA stops to operated when the byte coincidence is detected.

Since DMA operation is pipelined and the advance reading is performed, a check to determine if the n th data coincides with the coincidenced byte is carried out at the same time when the n + 1 st data is transferred. Therefore, data of N+1 byte is transferred and $\overline{\text{BUSREQ}}$ signal is set to "1" level at the leading edge of the clock when this transfer ended.

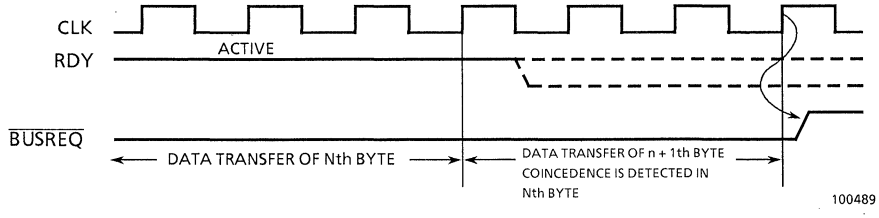


Figure 3.25 Bus Release Timing on Match - Byte Mode, Continuous Mode

(h) Bus release when RDY signal is non-active

If RDY signal becomes non-active in the burst mode, $\overline{\text{BUSREQ}}$ signal is set to "1" level at the rising edge of next clock after end of the byte operation that is under execution at the time. For instance, this is done when the read of the search only or simultaneous transfer/search operation ended or when the write of the trasfer/search operation ended. Therefore, the action for $\overline{\text{BUSREQ}}$ signal is slightly behind the action for RDY signal.

DMA always does not release the bus until the byte action at the time is completed.

In contrast with this, in the continuous mode $\overline{\text{BUSREQ}}$ signal is kept at "0" level even when RDY signal becomes non-active.

In addition, after the byte action at the time ended DMA is put in the idle state until RDY signal becomes active again.

This figure is shown in Figure 3.26.

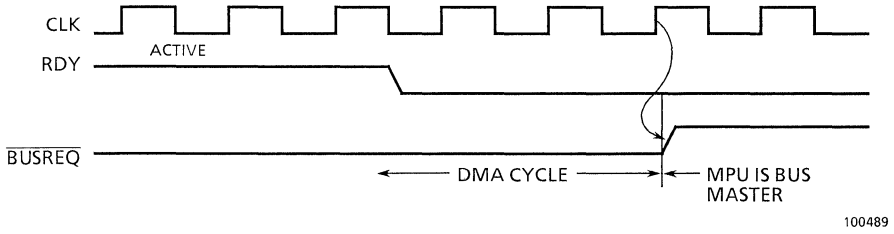


Figure 3.26 Bus Release Timing when RDY Signal becomes Non-active

Timing of RDY signal with other signals are shown in Figure 3.27, 3.28 and 3.29. In these figures the memory search only operation by the Z80 standard timing by mode is assumed. In each of the operation modes, RDY signal is sampled at the rising edge of the last clock of the read or write cycle to determine its level.

RDY signal can become non-active before completion of the last byte operation withour affecting its operation. In the byte or burst mode, $\overline{\text{BUSREQ}}$ signal and $\overline{\text{BAI}}$ signal are set to "1" at the end of byte operation of RDY signal. In the byte or burst mode, the bus control signals ($\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$) are also kept at "1" level as long as RDY signal is non-active. Further, the address bus and data bus are kept in 3 state.

The continuous mode differs from other modes in that the address bus holds an address which is incremented in advance against next byte during the period when RDY signal is non-active. This address can be used immediately after RDY signal becomes ative again.

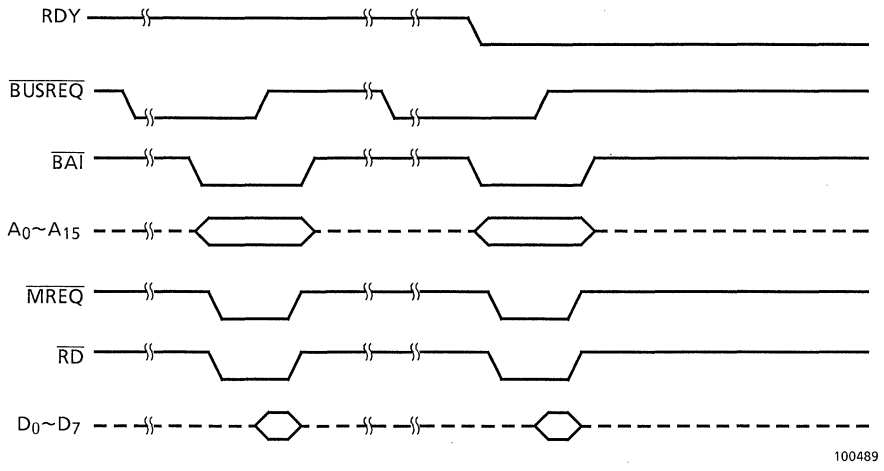


Figure 3.27 Timings of RDY Signal with Other Signals (Byte Mode)

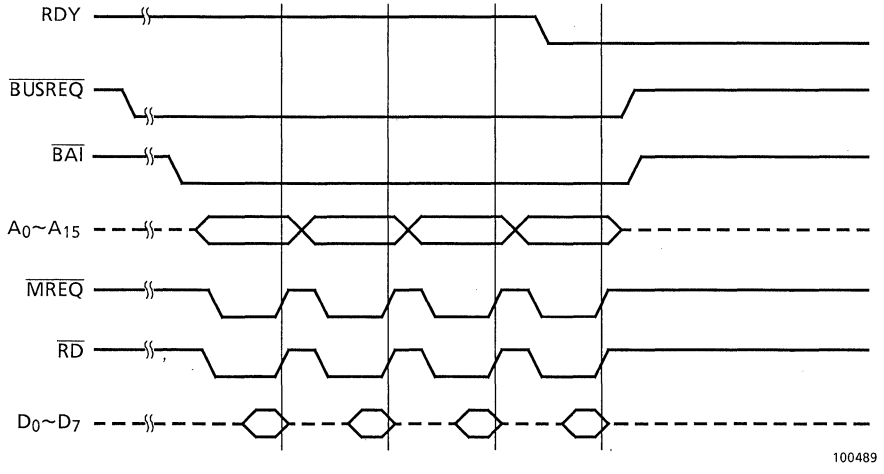


Figure 3.28 Timings of RDY Signal with Other Signals (Burst Mode)

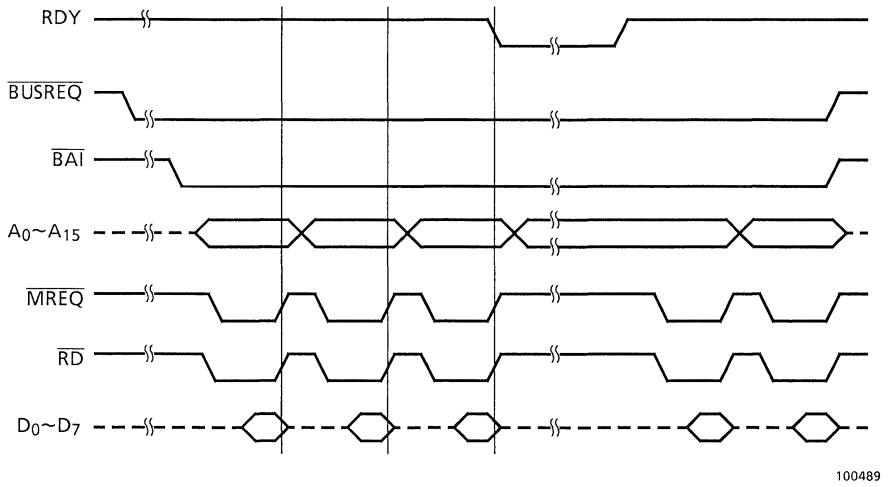


Figure 3.29 Timings of RDY Signal with Other Signals (Continuous mode)

(i) Variable cycle

When programmed, DMA is capable of changing read and write cycle lengths. Source and destination can be programmed independently by the write register WR1 (designation of Port A) and WR2 (designation of Port B). This variable cycle function allows the read or write in 2, 3 or 4 clock cycles (more clock cycles if Tw is inserted) and further, can increase or decrease pulse widths of all signals generated by DMA. Four signals relative to the data transfer; $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals have the function to end the rising edge timing earlier by 1/2 clock independently.

Differing from the standard timing, in the variable cycle mode $\overline{\text{IORQ}}$ signal becomes active earlier than $\overline{\text{MREQ}}$, $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals by 1/2 clock. Further, $\overline{\text{CE/WAIT}}$ signal can be used in the extension of 3 or 4 clock cycle variable memory cycle and 4 clock cycle variable I/O cycle only. In the 3 or 4 clock cycle memory operation, $\overline{\text{CE/WAIT}}$ signal is sampled at the T2 falling edge while it is sampled at the T3 falling edge in the 4 clock cycle I/O operation. In the 2 clock cycle operation it is not sampled. Use of this variable cycle effective in increasing data transfer rate and reducing software burden and further, can eliminate an external logic circuit. In addition, this function provides more faster memory read/write speed than normal speed.

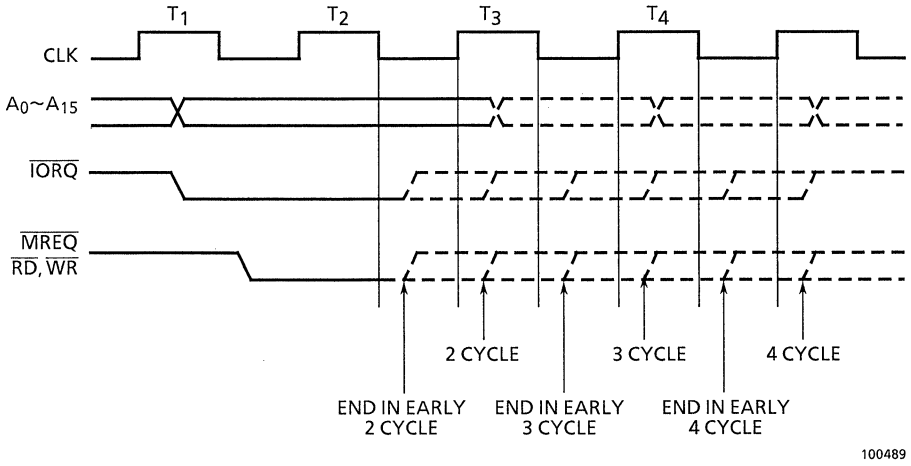


Figure 3.30 Variable Cycle

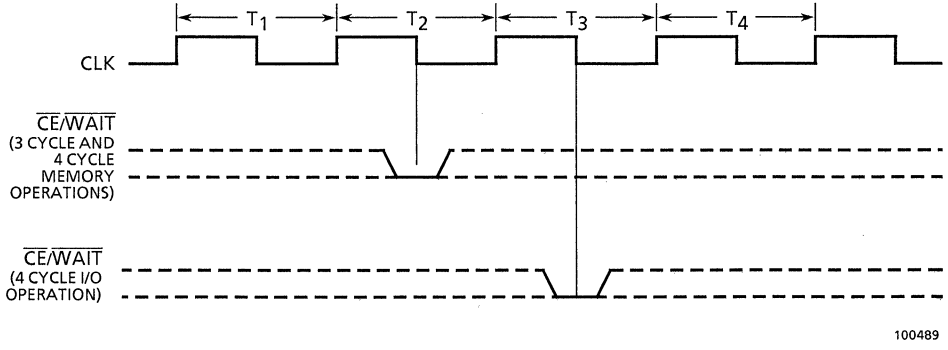


Figure 3.31 $\overline{\text{WAIT}}$ Sample in Variable Timing

(j) Interrupt

The timing for the interrupt acknowledge or return from interrupt is identical to that of other Z80 peripheral LSI's. INT signal is sampled by MPU at the rising edge of the last clock of all commands. If the interrupt enable is not set by the internal MPU software or when $\overline{BUSREQ}$ signal is active, this $\overline{INT}$ signal is not accepted. When $\overline{INT}$ signal is accepted, $\overline{IORQ}$ signal also becomes active at the same time (normally, $\overline{MREQ}$ signal) in the period of its M1 cycle, indicating that the interrupting LSI can load its 8-bit vector on the data bus. At the same time, two wait status are automatically inserted into this cycle. This is to facilitate execution of the priority interrupt mechanism and the wait status of 2T gives a stabilizing time to IE1 and IE0 signals and thus, it becomes possible to identify which peripheral LIS will react.

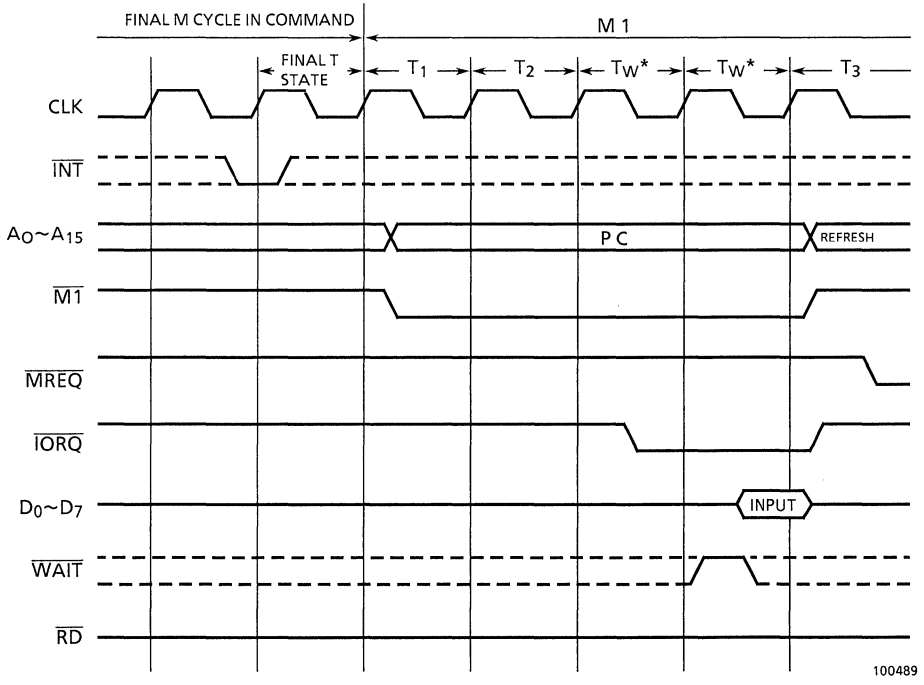


Figure 3.32 Interrupt Acknowledge

Interrupt on RDY signal (interrupt before the bus request) does not directly affect $\overline{BUSREQ}$ signal. The process in this case is carried out by giving following commands to the write register WR6 in the interrupt service routine.

- Enable after interrupt return (B7H)
- DMA enable (87H)

- Execution of REI instruction to reset IUS latch during the interrupt service in the Z80 DMA (ED4DH)

(k) Pulse generation

In the pulse generation, $\overline{INT}$ signal is set to "0" level (pulses are generated on the $\overline{INT}$ line) every 256 bytes after offset value is loaded to the write register WR4 by the program.

$\overline{INT}$ signal is put to "0" level during the DMA cycle in which pulse control bytes coincide with low order bytes of the byte counter and kept at "0" level in the full period of transfer cycle. Here, the transfer cycle means the read cycle (the search only or simultaneous transfer operation) or read/write cycle and lengths of the read and write cycles can be set independently by variable cycle.

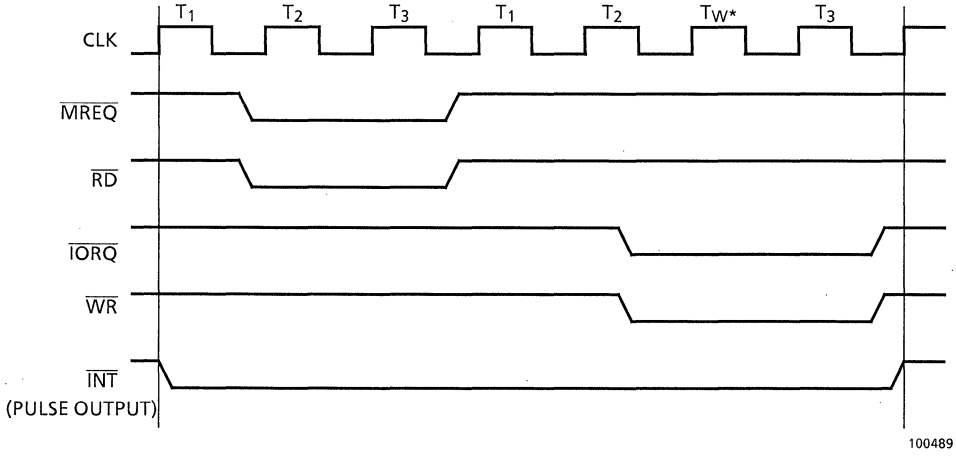


Figure 3.33 (a) Pulse Output (Standard timing at the time of transfer)

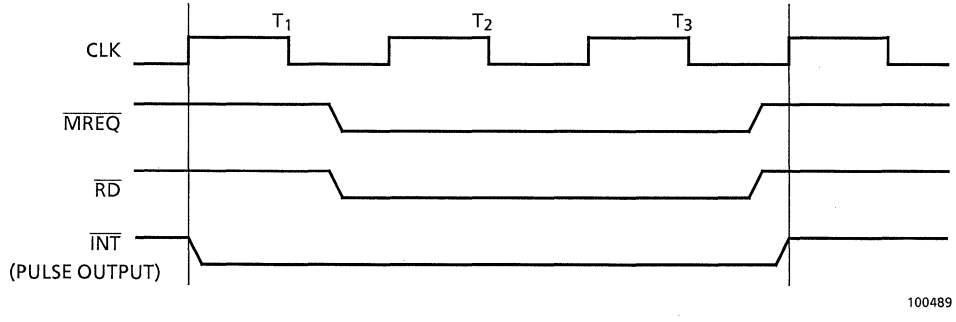


Figure 3.33 (b) Pulse Output (Standard timing at the time of memory search)

(l) Precautions

① Transfer timing

Although the DMA transfer timing is basically identical to the read/write timing of the Z80 MPU, care is required when variable cycle is used or in case of simultaneous transfer.

In the case of simultaneous transfer, all addresses which are output by DMA are interpreted to be memory addresses and I/O are selected by the hardware using an external logic circuit. It is normally programmed that I/O addresses are fixed and memory addresses are updated during the DMA operation. At this time, DMA controls memory addresses and outputs I/O select signal using an external logic circuit.

② Memory refresh

Since DMA has no refresh signal output function, the refresh of a dynamic RAM is performed normally using RFSH signal of MPU. If the transfer period becomes long in the DMA operation using the burst mode or continuous mode, another refresh method must be used.

③ Pulse generation

When the pulse generating function is used for transfer in the byte mode, pulse output is generated in two times. This is to avoid BAI signal from becoming non-active and MPU from being put in HALT state. Further, when offset value and low order 8 bits of the block length are equal other, pulse is once generated and after DMA operation is completed, pulse is generated during the read cycle of the 1st byte when the DMA operation is performed again without changing the offset value.

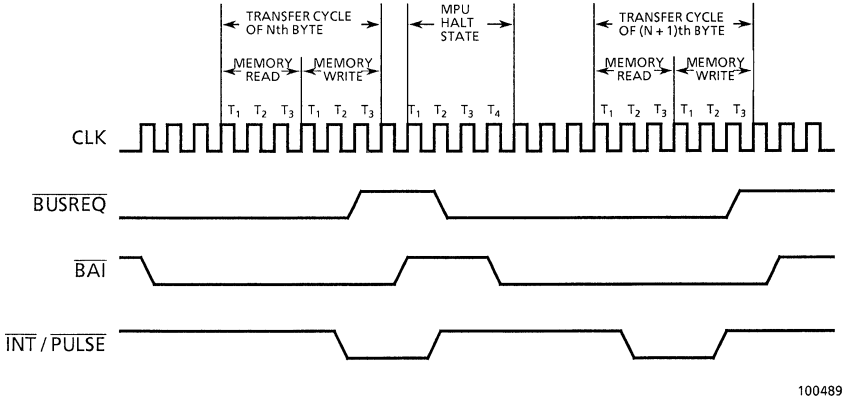


Figure 3.34 Pulse Generation Timing (Byte mode)

3.4 PERIPHERAL COMMANDS

To operate DMA, specify its operations by writing into the control register group through programming. In addition, the status of DMA can be known by reading the contents of the status register group.

To give effect to this on a program, write the operation into the write register by OTIR or OUT command to MPU and read out by INIR or IN command. In both cases, output of the I/O address decoder to DMA becomes "0" level. This output is connected to the $\overline{CE}/\overline{WAIT}$ pin.

The configurations of the control register group and status register group are as follows:

(1) Control register group

① Write register WR0
② Write register WR1
③ Write register WR2
④ Write register WR3
⑤ Write register WR4
⑥ Write register WR5
⑦ Write register WR6

(2) Status register group

① Readout register RR0
② Readout register RR1
③ Readout register RR2
④ Readout register RR3
⑤ Readout register RR4
⑥ Readout register RR5
⑦ Readout register RR6

3.4.1 Control register group

The control registers consist of 7 groups of WR0 to WR6, each of which consists of a basic register and related registers. If the pointer bit of the basic redister is "1", related redisters are accessed by turns.

The basic registers WR0 to WR6 are identified by the combination of bits 0, 1, 2, 6 and 7. There may be pointer bits for related registers. BBH (followed by the readout mask) command of WR6 has no pointer bit but data that follows this command is limited to the readout mask.

(1) Write register WR0

WR0 is identified by the condition that Bit 7 of the basic register is "0" and Bit 1, 0 are other than both "0".

WR0 has 4 pointer bits, each of which has related registers, respectively.

(a) Basic register bit 0, 1 (Designation of operating class)

Bit 0 and 1 designate the operating class; transfer, search only, and transfer/search operations. In addition, simultaneous transfer or transfer/search is obtained by selecting search and generating a proper bus control signal for complete transfer through external hardware.

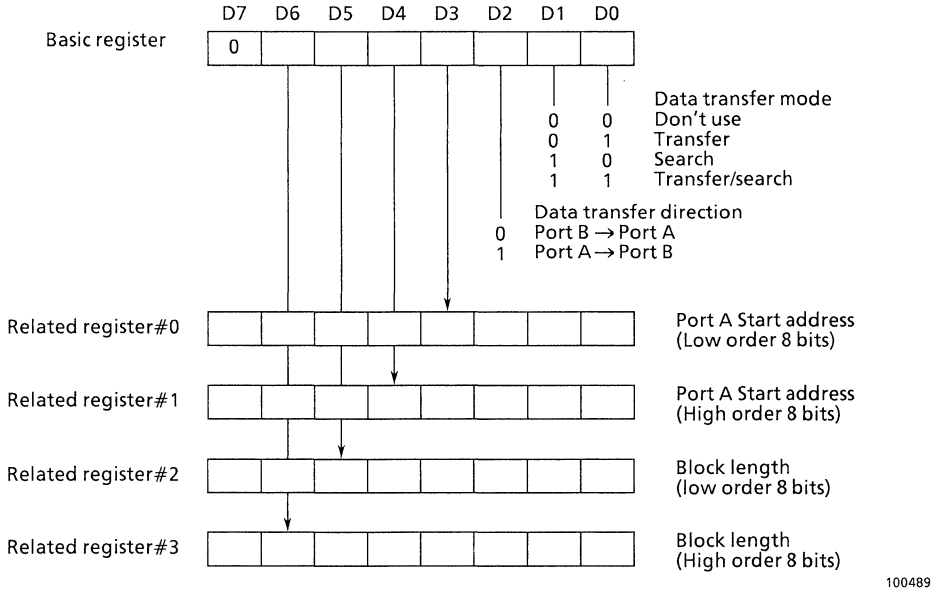


Figure 3.35 Write Register WR0

(b) Basic register bit 2 (Designation of data transfer direction)

Bit 2 declears the source port, and by implication, the destination port, if the operation is a sequential transfer. In the search only operation, the source port only is designated and in the simultaneous transfer or transfer/search operation, the destination port is decided by external wiring.

(c) Basic register bit 3-6 (Pointer bits)

Bit 3-6 are the pointer bits which are used to designate four related registers following respective bits.

(d) Related register #0, #1 (Port A start address)

These registers are accessed by Bit 3 and 4 of the basic register byte. When Port A is used as a source or destination, it is necessary to write the start address. Low order byts are written into #0 and high order bytes in #1.

(e) Related register #2, #3 (Block length)

These registers are designated by Bit 5 and 6 of the basic register.
Max. 64K bytes can be designated by writting low order bytes of block length into #2 and high order bytes into #3. However, as data read is pipe line type, number of bytes actually searched or transferred is more than that entered here by 1 or 2.

In addition, if "zero" is set for these registers, the transfer or search of $2^{16}+1$ bytes is carried out.

(2) Write register WR1

WR1 is identified by the condition that all of Bits 0, 1 and 7 of the basic register are "0" and Bit 2 is "1".

(a) Basic register bit 3 (Port A designation)

A memory is designated by Port A when "0" is written for Bit 3, while I/O is designated when "1" is written. This designation makes the control signal ($\overline{MREQ}$ or $\overline{IORQ}$) active against the cycle including this port.

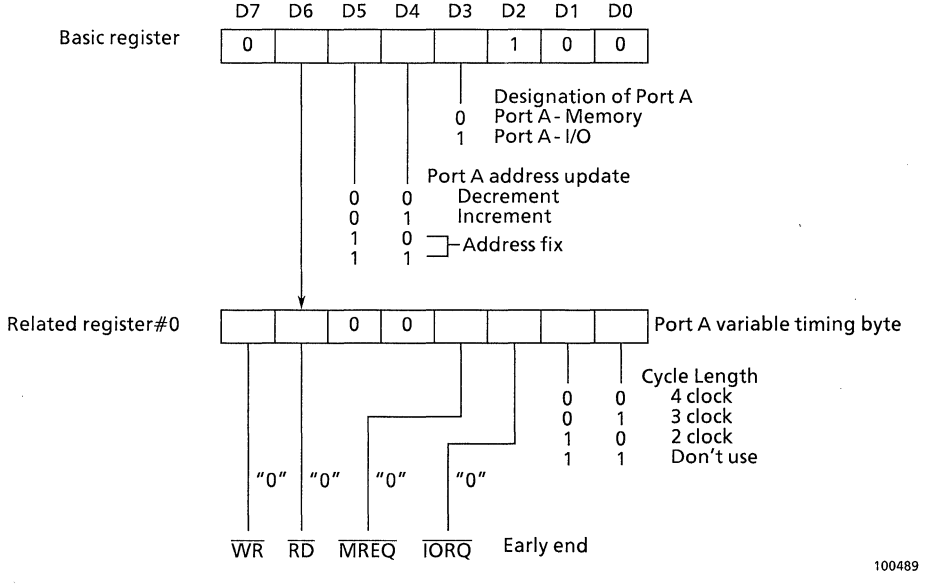


Figure 3.36 Write Register WR1

(b) Basic register Bit 4, 5 (Fixed or variable address designation)

Fixed or variable Port A address is designated by Bit 4 and Bit 5 for each transfer or search byte.

(c) Basic register Bit 6 (Pointer bit)

When Bit 6 is set to "1". next related register is accessed. In addition, when Bit 6 is set to "0", DMA's variable cycle is not used.

(d) Related register #0 (Port A variable timing byte)

By setting values for this register, Port A cycle length and control signal timing can be designated.

- Bit 0, 1 (Cycle length)

  Length of data transfer cycle (memory read/write, I/O read/write) relative to Port A is designated. Timing can be changed in a range of 2 - 4 clocks.

- Bit 2, 3, 6, 7 (Early end)

  The timing of the control signal $\overline{IORQ}$, $\overline{MREQ}$, $\overline{RD}$ and $\overline{WR}$ can be advanced by 1/2 clock.

(3) Write register WR2

WR2 is identified by the condition that all of Bits 0, 1, 2 and 7 of the basic register are "0".

(a) Basic register bit 3 (Port B designation)

A memory is designated by Port B when "0" is written for Bit 3, while I/O is designated when "1" is written. This designation makes the control signal ($\overline{MREQ}$ or $\overline{IORQ}$) active against the cycle including this port.

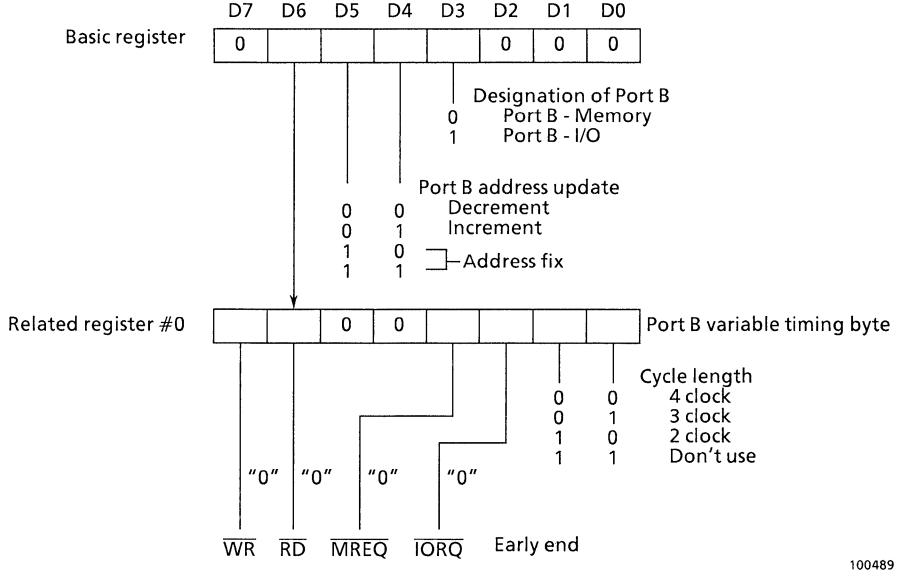


Figure 3.37 Write Register WR2

(b) Basic register Bit 4, 5 (Fixed or variable address designation)

Fixed or variable Port B address is designated by Bit 4 and Bit 5 for each transfer or search byte.

(c) Basic register Bit 6 (Pointer bit)

When Bit 6 is set to "1", next related register is accessed. In addtion, when Bit 6 is set to "0", DMA's variable cycle is not used.

(d) Related register #0 (Port B variable timing byte)

By setting values for this register, Port B cycle length and control signal timing can be designated.

- Bit 0, 1 (Cycle length)

  Length of data transfer cycle (memory read/write, I/O read/write) relative to Port B is designated. Timing can be changed in a range of 2 - 4 clocks.

- Bit 2, 3, 6, 7 (Early end)

  The timing of the control signal $\overline{IORQ}$, $\overline{MREQ}$, $\overline{RD}$ and $\overline{WR}$ can be advanced by 1/2 clock.

(4) Write register WR3

WR3 is identified by the condition that both Bit 0 and Bit 1 of the basic register are "0" and Bit 7 is "1".

(a) Basic register Bit 2 (Stop on match)

This bit is used for the search or transfer/search operation. When this bit is "1" and transferred data matches the match byte, the data transfer is stopped and the bus is released. When this bit is "0" and transferred data matches the match byte (if DMA is not stopped even when they matched), the status flag is set on the status byte to allow interrupt resulting from byte match.

(b) Basic register Bit 3 (Pointer bit)

When this bit is set at "1", the mask byte follows the basic register.

(c) Basic register Bit 4 (Pointer bit)

When this bit is set at "1", a match byte follows the basic register.
This bit designates a match byte used for comparison with all data to be searched.

(d) Basic register Bit 5 (Interrupt enabled)

When this bit is set at "1", DMA interrupt is enabled.

(e) Basic register Bit 6 (DMA enable)

When this bit is set at "1", DMA operation is enabled and a bus request can be made to MPU.

(f) Related register #0 (Mask byte)

This register is accesed by basic register Bit 3. It is possible to write a mask byte required for the search operation. The mask byte is capable of masking the match byte (data to be compared) during the search operation to extract bits to be compared.

When all bits of the mask byte are at "0", the comparison is made and when they are at "1", the masking is made. In addition, if no masking is required and all bits are compared, write 00H mask byte.

(g) Related register #1 (match byte)

This register is accessed by basic register Bit 4. The match byte is used as data to be compared when the data transfer mode is search or transfer/search. The match byte is masked by the mask byte of related register #0.

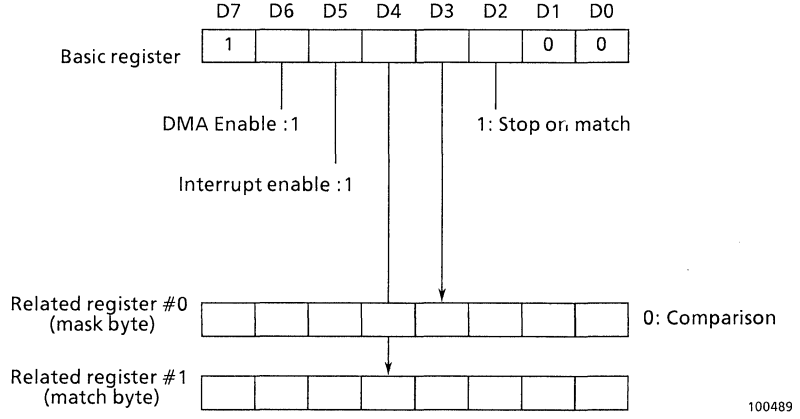


Figure 3.38 Write Register WR3

(5) Write register WR4

WR4 is identified by the condition that both Bit 0 and Bit 7 of the basic register are "1" and Bit 1 is "0".

(a) Basic register Bit 2 - 4 (Pointer bit)

When these bits are set at "1", the related registers #0 to #2 are accessed after the basic register.

(b) Basic register Bit 5, 6 (Operation mode designation)

Operation mode selected values can be set for Bit 5 and Bit 6. For values to be set, refer to Figure 3.39.

(c) Related register #0, #1 (Port B start address)

These registers are accessed by basic register Bit 2 and Bit 3. Related register #0 designates low order byte of Port B start address while related register #1 designated high order byte of Port B. Further, if low order 8 bits are sufficient for Port B start address, thr loading to related register #1 is not required.

(d) Related regiser #2 (Interrupt control byte)

This register control DMA interrupt or pulse generation. By setting Bit 3 and Bit 4, related registers #3 and #4 can be accessed.

- Bit 0 (Interrupt on match)

When Bit 0 is set at "1", DMA generates interrupt if transferred data matches the match byte in the search or transfer/search operation.

- Bit 1 (Interrupt on end of block)

When Bit 1 is set at "1", DMA generates interrupt if a value of the byte counter becomes "0" in DMA operation.

- Bit 2 (Pulse generation)

When Bit 2 is set at "1", pulse is generated on the $\overline{INT}$ line whenever data in number of bytes set on the pulse control byte is transferred.

- Bit 3, 4 (Pointer bits)

When Bit 3 is set at "1", pulse control byte is accessed after interrupt control byte. When Bit 4 is set at "1", the interrupt vector is accessed.

- Bit 5 (Vector value change by status)

When this bit is set at "1", interrupt vector value changes according to cause for generating interrupt. However, if the automatic restart or interrupt at the end of block was already set, this mode cannot be used.

- Bit 6 (Interrupt on RDY)

When this bit is set at "1", DMA generates interrupt prior to the bus request if it detects that RDY signal has become active. Therefore, the interrupt enable command becomes necessary and when RETI instruction is executed after the interrupt enable of WR6 (B7H) is sent out, the bus request is started.

(e) Related register #3 (Pulse control byte)

When Bit 3 of the interrupt control byte is set at "1", the pulse control byte is accessed after the interrupt control byte. The pulse control byte gives offset values to pulse that are first generated (Number of bytes shown by this control byte).

The pulse control byte compares low order 8 bits of the byte counter and if both coincide each other, pulses are output on the $\overline{INT}$ line.

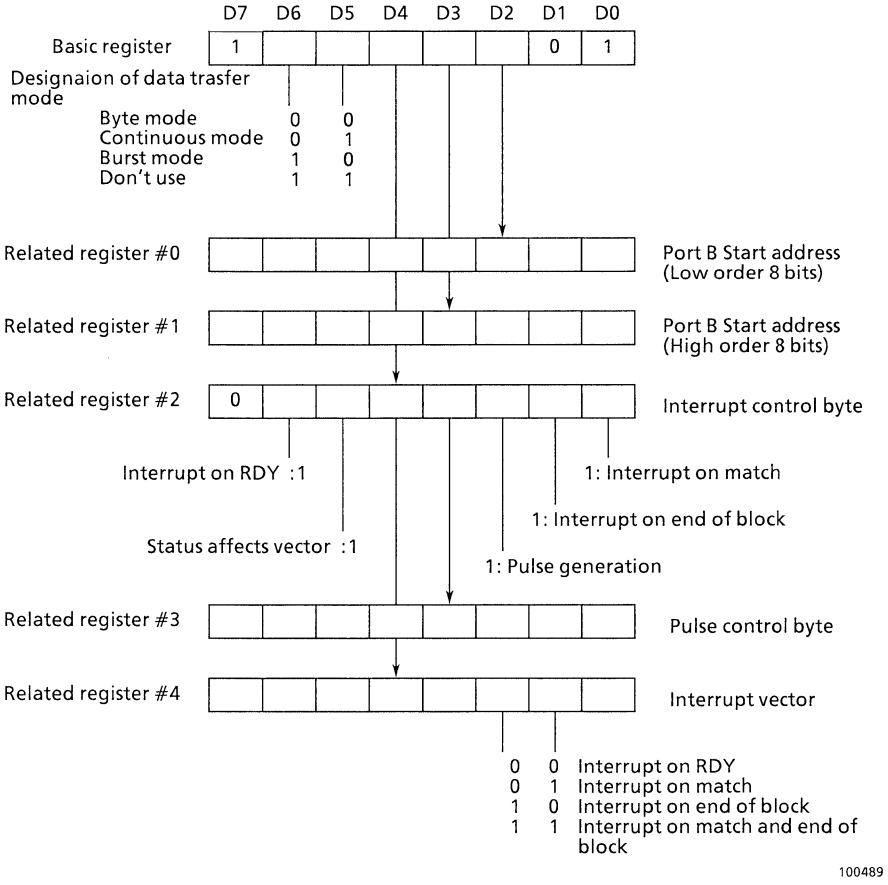


Figure 3.39 Write Register WR4

(f) Related register #4 (Interrupt vector)

When Bit 4 of the interrupt control byte is set at "1", the interrupt vector is accessed after the interrupt control byte. The interrupt vector is loaded on the data bus at time of the interrupt acknowledge by MPU ($\overline{IORQ}$ = "0", $\overline{M1}$ = "0"). If Bit 5 of the interrupt control byte is set at "1", Bit 1 and Bit 2 of the interrupt vector change according to the interrupt factor. However, when the automatic restart and interrupt on the end of block has been already programmed, the interrupt vector sent out at the end of block does not change and therefore, the mode for vector value change by status cannot be used.

(6) Write register WR5

WR5 is identified by the condition that Bit 1 and 7 of the basic register are "1" and Bit 0, 2 and 6 are "0". WR5 has no related register.

- Bit 3 (Effective polarity of RDY signal)

  When this bit is set at "0", RDY signal becomes "0" and active, and when it is set at "1", RDY signal becomes "1" and active.

- Bit 4 (Use of $\overline{CE}/\overline{WAIT}$ pin)

  When this bit is set at "0", $\overline{CE}$ function only is available. When it is set at "1", both $\overline{CE}$ and $\overline{WAIT}$ functions become available. When $\overline{BUSREQ}$ signal is at "1", CE function is available, while if $\overline{BUSREQ}$ signal is at "0" level, WAIT function is available.

- Bit 5 (Automatic register)

  When this bit is set at "0" level, DMA operation is stopped at time of the end of block (Byte counter = "Zero"). When it is at "1" level, the contents of the address register and byte counter are automatically loaded on the address counter and byte counter, and DMA operation is continued.

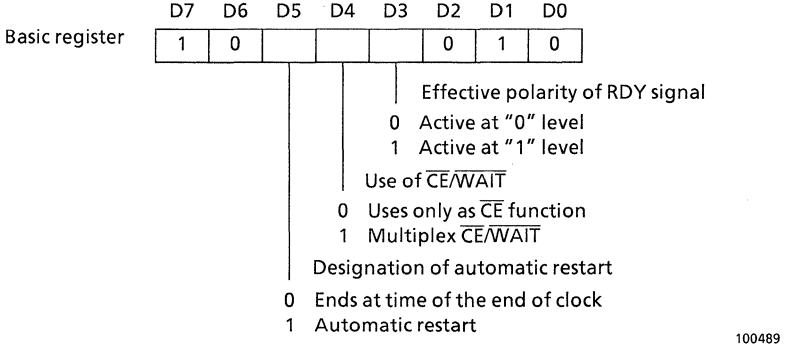


Figure 3.40 Write Register WR5

(7) Write register WR6

WR6 is identified by the condition that Bit 0, 1 and 7 of the basic register are at "1" level. In the case of WR6, the functions (commands) are divided by the combination of Bits 2 to 6.

(a) Reset (C3H)

This command is used to reset DMA. By executing this command, DMA performs the followings;

- Disables the interrupt control circuit and bus request control circuit.
- Releases the interrupt latch.
- Releases the forced RDY condition.
- Releases the automatic restart condition.
- Releases $\overline{CE}/\overline{WAIT}$ function (Bit 4 WR5) and retains $\overline{CE}$ function only.
- Returns the timings of both Port A and B to the Z80 standard timing.

After turning power to DMA ON, and performing the programming, it is necessary to execute the reset command once. In addition, if the DMA operation is interrupted during its execution, it is necessry to execute the reset command 6 times successively (this is because there are 5 related registers that are capable of directing in WR4).

Further, DMA is not always reset completely by this reset command. The read sequence is reset only by the read sequence reset command.

(b) Port A timing reset (C7H)

This command resets Port A variable timing byte and returns Port A timing to the Z80 standard timing.

(c) Port B timing reset (CBH)

This command resets Port B variable timing byte and returns Port B timing to the Z80 standard timing.

(d) Load (CFH)

When this command is executed, the content of the address register is loaded on the address counter and the byte counter is cleared. In addition, the internal forced RDY condition is also released.

Further, an address counter to which the loading can be made immediately is the source port address counter only. The loading to the destination port address counter is made when a value of this counter is initially updated (incremented/decremented) but if "address if fixed", the loading is not performed. However, the loading by the "fixed address destination port programming" technique is possible. If DMA becomes non-active when the load command is written, another DMA control byte is written before the load command.

(e) Continue (D3H)

Although this command clears the byte counter to "zero", both port address counters do not change. This command is used in transferring several data blocks to continued positions in the many buffer if it is desirable to know a break of every block, and continues DMA operation which has been interrupted by detection of match at the end of block or search.

In order to execute this command, interrupt at the end of each blocks is needed and new block length shall be entered in WR0 with the continue command.

In transferring data blocks, interrupt becomes necessary whenever transfer of each data block ended. In trasferring next data block after the interrupt, this continuity command is used inserted of the load command.

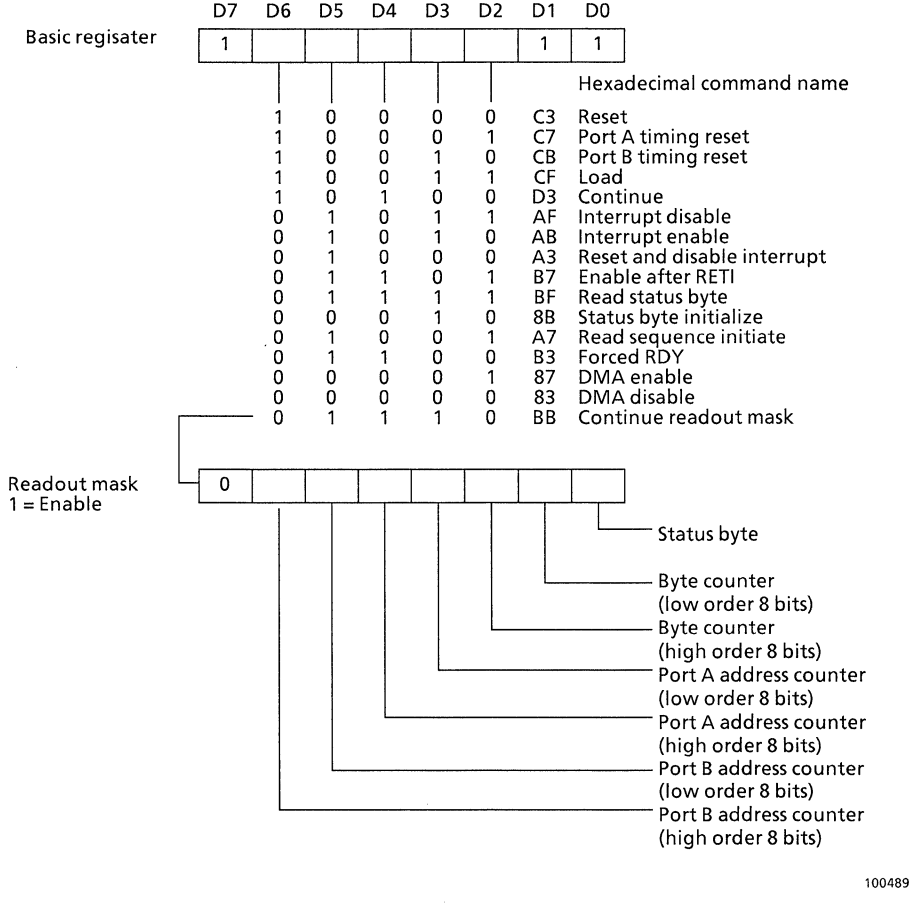


Figure 3.41 Write Register WR6

(f) Interrupts disable (AFH)

This command is used to simulate the Z80 MPU's interrupt acknowledge when DMA is operated in a system other than the Z80 MPU. When DMA sends the interrupt signal into any MPU other than the Z80 MPU, if the inerrupt disable command is written into the beginning of the service routine, $\overline{INT}$ signal returns to "1" level but next interrupt signal of DMA cannot be sent during the service routine is continuously carried out. Generation of next interrupt signal becomes possible when the interrupt enable command is written into the end of the service routine.

(g) Interrupts enable (ABH)

This is a command used on the Z80 system to enable the interrupt when the power source is ON. This command enables the interrupt control circuit of DMA. When the interrupt disable command is not used, if the interrupt enable command is once written, next interrupt is enabled automatically when RETI instruction is executed at the final stage of the interrupt service routine. However, if the interrupt disable command is used, it becomes necessary to write the interrupt enable command at the final stage of the interrupt service routine.

(h) Reset and interrupt disable (A3H)

This command is used on such systems as 8080 and 8085 to interface DMA and MPU which is provided with the interrupt acknowledge function but has no RETI instruction. This command, when executed, performs the followings:

- Reset of the in-interrupt service routine (IUS) latch
- Reset of the interrupt pending (IP) latch
- Release of the internal forced RDY signal conditions
- Succeesing interrupt disable by DMA (same as the interrupt disable command)

(i) Enable after RETI (B7H)

This command is used only when "interrupt on RDY signal" is programmed on WR4. DMA, when detecting that RDY signal becomes active, does not make the bus request but grenerates the interrupt signal. After the interrupt return, this command enables DMA to make the bus request again. This command is always used to make the bus request after the interrupt to RDY signal on the Z80 MPU system. This command also can be used on other MPU's, for instance, 8080.
The interrupt latch (IOR) to RDY signal is set during its interrupt cycle. This latch makes RDY signal active and DMA is not allowed to make the bus request until this latch is reset by Enable after RETI instruction.

The execution sequence of the Z80 MPU service routine is as follows and the bus request is mode after RETI instruction is executed:

```
    :
Enable after RETI
DMA Enable
    :
RETI Instruction
```

(j) Read status byte (BFH)

This command indicates that next read command is the status byte access.

(k) Status byte initialization (8BH)

This command indicates the reinitialization of bit 4 and 5 of the status byte. The reinitialization of the interrupt pending status (Bit 3) of the status byte can be effected by the interrupt acknowledge, interrupt process, interrupt reset and disable command writing. The reinitialization of DMA operation status (Bit 0) can be efected by the load command.

(l) Read mask continue (BBH)

This command denotes that next control byte which is to be written into DMA follows the read mask register. The read mask register is used for setting a new read sequence of RR0 to RR6 and is normally a part of the initial state setting when the power source for DMA is turned ON.

The read mask can be programmed by setting the related pointer bit of register to be read out "1" level. The read sequence start command is used for initialization.

(m) Read sequence initiate (A7H)

This command is used to initiate the read sequence pointer command as a measures to access the first (in low order) read register that is designated to be readable by the read mask for initialization of DMA by next MPU read command. Normally, this command is output to reset the read sequence immediately after loading of the read mask.

(n) Forced RDY (B3H)

In the burst mode or continuous mode, this command is used to make the internal RDY conditions active for the active RDY signal by an external logic circuit. This command is used for memory-to-memory transfer or memory search where RDY signals not required. It is not necessary to consider the effective polarity of RDY signal. Use of this command can eliminate an external logic circuit.

The forced RDY conditions are released by the following commands/conditions:

- Reset command
- Load command
- Interrurpt reset and disable command
- Ending by end-of-block
- Ending by byte match
- Bus release by DMA

(o) DMA enable (87H)

This command is used to enable the bus control circuit of DMA. The interrupt circuit is not affected nor the function and latch are reset. This bus request enable function is identical to that of Bit 6 of WR3. In the interrupt service routine, DMA enable command is the last command to DMA before MPU executes RETI instruction.

(p) DMA disable (83H)

This command inhibits the bus request by DMA. This command is used to stop DMA operation by external events, end-of-block or match by bytes and when reinitialization of the status byte is required.

3.4.2 Status register group

There are 7 read registers RR0 to RR6 available for DMA to know the operation execution or end status.

The readout of MPU is made according to the method to access DMA as the peripheral I/O using I/O command. Commands to be written into DMA are as follows:

① Read status byte (BFH)
② Read sequence initiate (A7H)
③ Status byte reinitialize (8BH)
④ Read mask continue (BBH)

The above commands are those which are shown for WR6.

(1) Read register PR0 - Status byte

(a) Bit 0 (DMA operation)

This bit indicates if DMA made the bus request after the last LOAD command. "1" indicates that DMA made the bus request while "0" indicates no bus request made.

(b) Bit 1 (RDY signal active)

"0" of this bit indicates that RDY signal is active. "1" indicates RDY signal being non-active.

(c) Bit 2 (Don't Care)

(d) Bit 3 (Interrupt pending)

This bit indicates the interrupt pending (IP) latch status. "0" indicates the interrupt pending.

(e) Bit 4 (Match detection)

When this bit is "0", it indicates the match after the last status byte reset or reintialization command.

(f) Bit 5 (End-of-block detection)

When this bit is "0", it indicates the end-of-block reached after the last status byte reset, load, continuity or reinitialization.

(g) Bit 6, 7 (Don't Care)

(2) Read register RR1, RR2 - Byte counter

The 16-bit counter consisting of two register RR1 or RR2 are cleared to zero by the load, continuity or resest command.

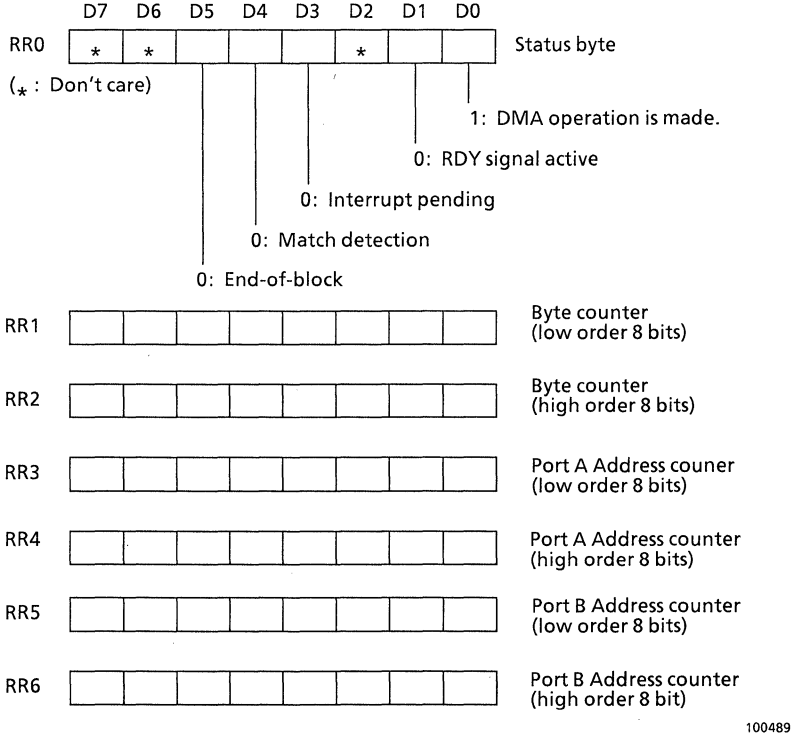


Figure 3.42 Read Register

When DMA starts the transfer or search, the byte counter is incremented by one at the end of each read cycle and judges the end-of-block by comparing with the program content of the block length register (WR0), when match is detected, DMA operation is stopped. If the pulse generation is used at this time, the content of the WR4 pulse control byte is, after transferred, compared with low order 4 byte (RR1) of the byte counter.

(3) Read register RR3, RR4 - Port A address counter

Write into the 16-bit counter consisting of two read registers RR3 and RR4 is made from WR0 port A start address register by the load command. Whenever one byte of DMA operation is carried out according to the designated content of WR0, this counter is updated (incremented or decremented) by one.

(4) Read register RR5, RR6 - Port B address counter

The 16-bit counter consisting of two read registers RR5 and RR6 indicates Port B address when DMA operation ended. Values in the port B address register (WR4) are loaded into this counter by the load command and the counter is updated by one every time when DMA operation is carried out by one byte. However, if address fix (Bit 4 and 5 of WR2) is programmed, the counter does not change. If port A or Port B is a fixed address destination port, in order to properly function the port it is necessary to program as described for the fixed address destination port.

3.4.3 Address counter and byte counter values when DMA operation ended.

Values of these counters when DMA operation ended are shown in Table 3.4 (a) and Table 3.4 (b).

Table 3.4 (a) Values of Those Counters when DMA Operation Ended

Data format	Data transfer mode	Value of block length register	Number of byte to be transferred	Value of byte counter	Address counter value of source port	Address counter value of destinaion port
Transfer	Byte	N	N + 1	N	As ± (N + 1)	As ± (N)
	Burst	N	N + 1	N	As ± (N + 1)	As ± (N)
	Continuity	N	N + 1	N	As ± (N + 1)	As ± (N)
Search	Byte	N	N + 1	N	As ± (N + 1)	
	Burst	N	N + 1	N	As ± (N + 1)	
			N + 2*	N + 1*	As ± (N + 2)*	
	Continuity	N	N + 1	N	As ± (N + 1)	
			N + 2*	N + 1*	As ± (N + 2)*	

As: Start address *: The values when N + 1 byte data is trasferred and RDY signal is active using 2-cycle valiable timing.

100489

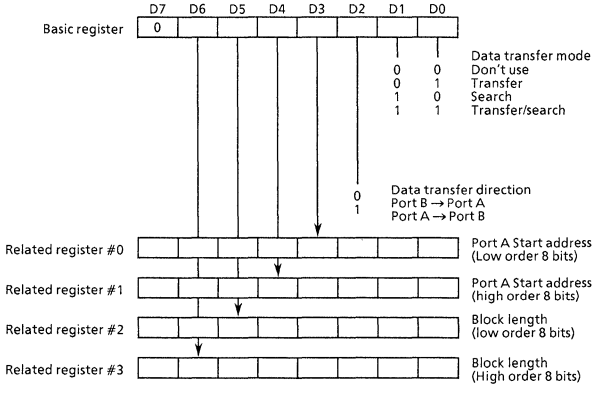
Table 3.4 (b) Values of Those Counters when DMA Operation Ended

Data format	Data transfer mode	Byte no. to be detected	Number of byte to be transferred	Value of byte counter	Address counter value of source port	Address counter value of destination port
Transfer	Byte	M	M	M-1	As ± (M)	As ± (M-1)
	Burst	M	M	M-1	As ± (M)	As ± (M-1)
	Continuity	M	M	M-1	As ± (M)	As ± (M-1)
Search	Byte	M	M	M-1	As ± (M)	
	Burst	M	M + 1	M	As ± (M + 1)	
			M*	M-1*	As ± (M)*	
	Continuity	M	M + 1	M	As ± (M + 1)	
			M*	M-1*	As ± (M)*	

As : Start address * : The values when match is detected and RDY signal is active.

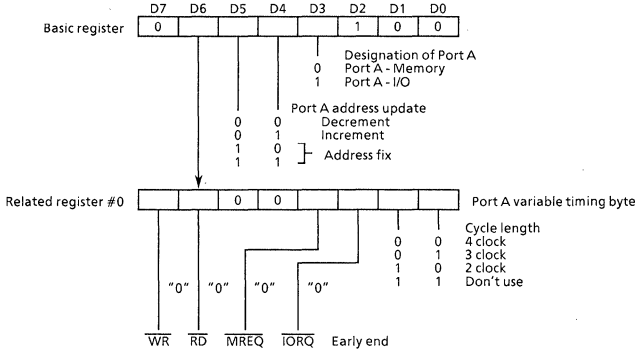
100489

3.4.4 List of command (1/3)



100489

Figure 3.35 Write Register WR0



100489

Figure 3.36 Write Register WR1

List of command (2/3)

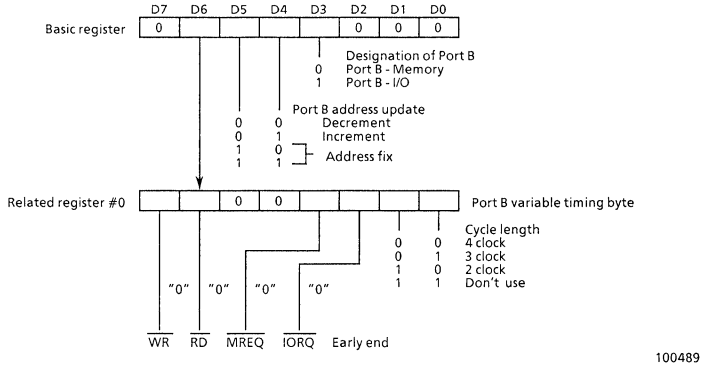


Figure 3.37 Write Register WR2

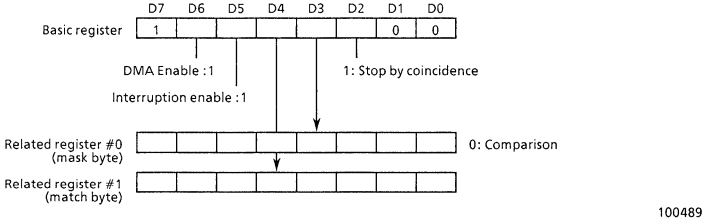


Figure 3.38 Write Register WR3

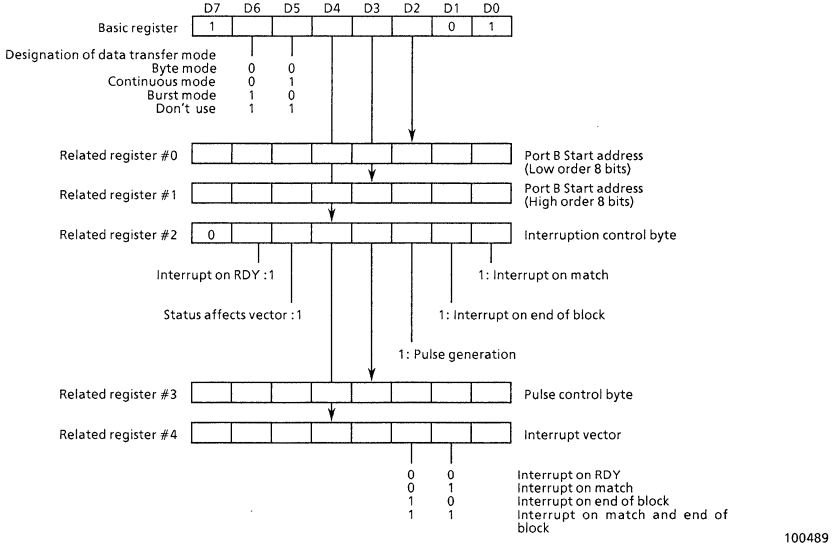


Figure 3.39 Write Register WR4

List of command (3/3)

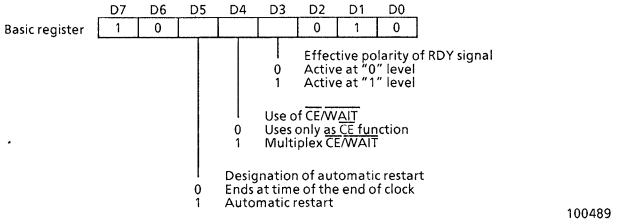


Figure 3.40 Write Register WR5

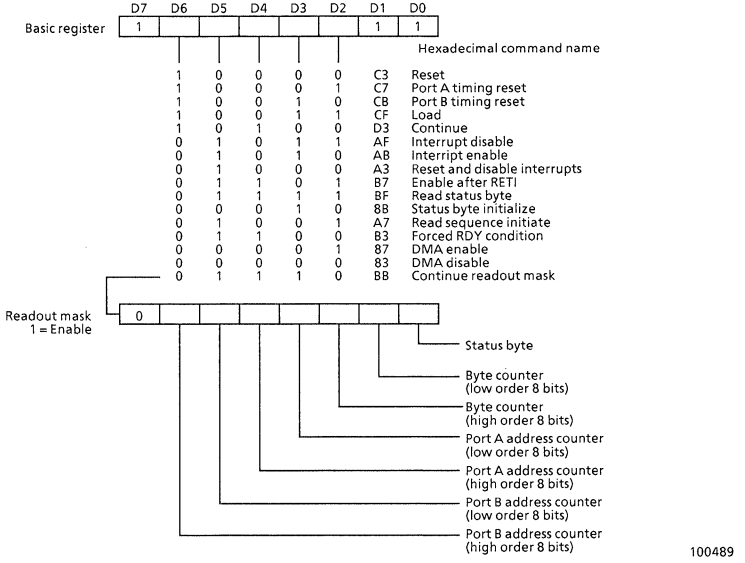


Figure 3.41 Write Register WR6

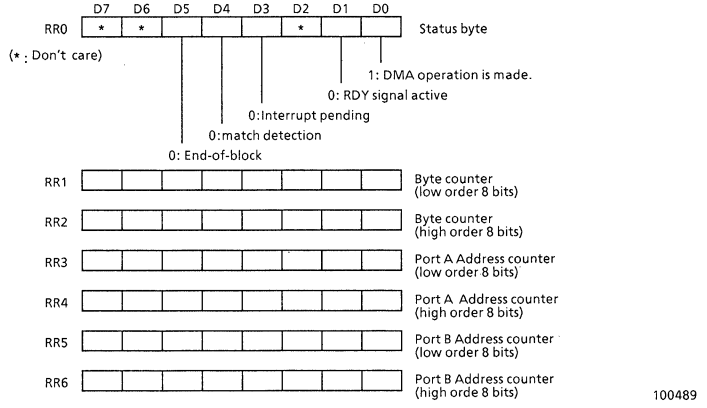


Figure 3.42 Read Register

3.5 METHOD OF USE

(1) Example of interface

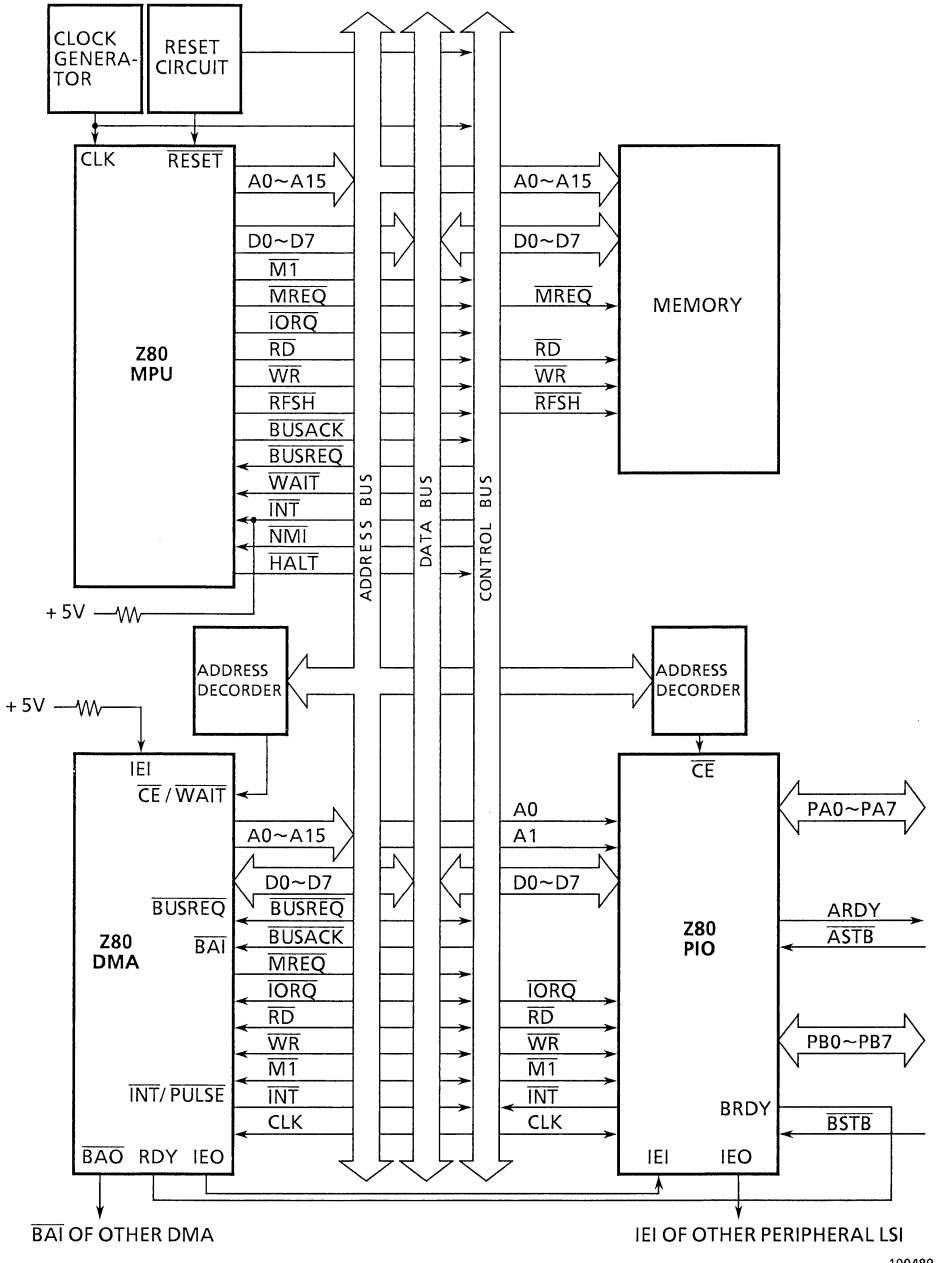


Figure 3.43 Example of Z80 System Interface

As the method of use of DMA using the Z80 family, a simple example of the Z80 system interface is shown below. Figure 3.43 shows the connection employing the Z80MPU, Z80DMA, Z80PIO, and MEMORY.

On the Z80 sistem, signal lines of MPU and peripheral devices are connected almost directly. In Figure 3.43, priority is in the order of DMA and PIO. This is because the connection is made according to the daisy chanin method which is peculiar to the Z80 system. In the case of DMA, espescially, in order to operate as a bus controller, $\overline{\text{BUSACK}}$ signal, which is the output of MPU, is input to BAI of DMA by the bus request daisy chain. When several units of DMA are used, BAO is connect to BAI of DMA which has the next higher priority. Hardware can be easily connected as shown in Figure 3.43. In addtion, memory mapped I/O also can be connected to DMA.

(2) Program example

As a program example of DMA operation, an example of transfer from I/O to memory using PIO is shown.

```
                        .Z80
                ;       Z80 DMA DATA TRANSFER
                ;       I/O TO MEMORY
                ;       (PIO PORT B)
                ;
0020            DPIOB   EQU   20H           ;DATA PIO CHANNEL B
0021            DPIOB   EQU   DPIOB+1       ;CONTROL PIO CHANNEL B
0030            DMA     EQU   30H           ;DMA ADDRESS
2000            DSTN    EQU   2000H         ;DESTINATION
0100            LENGTH  EQU   256           ;BLOCK LENGTH
                ;
                        ORG   1000H
1000' F3        STRT:   DI
1001' 3E 10             LD    A,10H
1003' ED 47             LD    I,A
1005' ED 5E             IM    2             ;INTERRUPT MODE 2
                ;
                ;   PIO B INITIALIZATION
1007' 3E 4F             LD    A,4FH         ;MODE 1
1009' D3 21             OUT   (CPIOB),A
100B' 3E 07             LD    A,07H
100D' D3 21             OUT   (CPIOB),A
                ;
                ;   DMA RESET
100F' 3E C3             LD    A,0C3H
1011' 06 06             LD    B,06H
1013' D3 30     DMRT:   OUT   (DMA),A
1015' 10 FC             DJNZ  DMRT
                ;
                ;   DMA INITIALIZATION
1017' 06 12             LD    B,DMAFIN-DMACTA
1019' 0E 30             LD    C,DMA
101B' 21 104E'          LD    HL,DMACTA
101E' ED B3             OTIR
```

```
1020'  FB                EI
1021'  C9                RET
                  ;
                  ;    INTERRUPT ON READY
1022'  76         IOR:   HALT
                  ;
                  ;    INTERRUPT ON MATCH
1023'  76         IOM:   HALT
                  ;
                  ;    INTERRUPT ON END OF BLOCK
1024'  CD 1037'   IOE:   CALL  SAV
1027'  3E 8B             LD    A,8BH
1029'  32 0030           LD    (DMA),A
102C'  06 04             LD    B,DMAFIN-DMACTB
102E'  0E 30             LD    C,DMA
1030'  21 105C'          LD    HL,DMACTB
1033'  ED B3             OTIR
1035'  C9                RET
                  ;
                  ;    INTERRUPT ON MATCH,END OF BLOCK
1036'  76         IME:   HALT
                  ;
                  ;    REGISTER SAVE
1037'  E3         SAV:   EX    (SP),HL
1038'  D5                PUSH  DE
1039'  C5                PUSH  BC
103A'  F5                PUSH  AF
103B'  DD E5             PUSH  IX
103D'  FD E5             PUSH  IY
103F'  CD 104D'          CALL  RUN
1042'  FD E1             POP   IY
1044'  DD E1             POP   IX
1046'  F1                POP   AF
1047'  C1                POP   BC
1048'  D1                POP   DE
1049'  E1                POP   HL
104A'  FB                EI
104B'  ED 4D             RETI
                  ;
104D'  E9         RUN:   JP    (HL)
                  ;
                  ;    DMA COMMAND TABLE
                  ;      PORT A - MEMORY
                  ;      PORT B - PIO CHANNEL B
104E'             DMACTA  EQU   $
104E'  C3                DEFB  0C3H          ;WR6   RESET COMMAND
104F'  7D                DEFB  7DH           ;WR0   PORT A TO PORT B (TEMP)
1050'  2000              DEFW  DSTN          ;      DESTINATION ADDRESS
1052'  00FF              DEFW  LNGTH-1       ;      BLOCK LENGTH
1054'  14                DEFB  14H           ;WR1   PORT A - "INCREMENT" ADDRESS
1055'  28                DEFB  28H           ;WR2   PORT B - "FIXED" ADDRESS
1056'  A0                DEFB  0A0H          ;WR3   ENABLE INTERRUPT
1057'  95                DEFB  95H           ;WR4   BYTE MODE TRANSFER
1058'  20                DEFB  DPIOB         ;      PORT B ADDRESS (L)
```

```
1059'  32              DEFB  32H           ;       IOE,STATUS AFFECTS VECTOR
105A'  FF              DEFB  INTV-STRT     ;INTV   INTERRUPT VECTOR
105B'  82              DEFB  82H           ;WR5    RDY ACTIVE 'LOW',CE/ ONLY
105C'          DMACTB  EQU   $
105C'  CF              DEFB  0CFH          ;WR6    LOAD ADDRESS TO PORT A

105D'  01              DEFB  01H           ;WR0    PORT B TO PORT A
105E'  CF              DEFB  0CFH          ;WR6    LOAD ADDRESS TO PORT B
105F'  87              DEFB  87H           ;WR6    ENABLE DMA
1060'          DMAFIN  EQU   $
               ;
                       ORG   STRT+0FFH
10FF'  1022'   INTV:   DEFW  IOR
1101'  1023'           DEFW  IOM
1103'  1024'           DEFW  IOE
1105'  1036'           DEFW  IME
                       END
```

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item	Rating	Unit
V_{CC}	Supply Voltage	− 0.5 to + 7	V
V_{IN}	Input Voltage	− 0.5 to VCC + 0.5	V
P_D	Power Diddipation (TA = 85°C)	250	mW
T_{SOLDER}	Soldering Temperature (10 sec)	260	°C
T_{STG}	Storage Temperature	− 65 to 150	°C
T_{OPR}	Operating Temperature	− 40 to 85	°C

100489

4.2 DC ELECTRICAL CHARACTERISTICS

$T_A = -40°C$ to $85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
VILC	Low Clock Input Voltage			− 0.3	—	0.6	V
VIHC	High Clock Input Voltage			V_{CC} − 0.6	—	V_{CC} + 0.3	V
VIL	Low Inpur Voltage (Except CLK)			− 0.5	—	0.8	V
VIH	High Input Voltage (Except CLK)			2.2	—	V_{CC}	V
VOL	Output Low Voltage	IOL = 2.0mA $\overline{BUSREQ}$ only 3.2mA		—	—	0.4	V
VOH1	Output High Voltage (I)	IOH = − 1.6mA		2.4	—	—	V
VOH2	Output High Voltage (II)	IOH = − 250μA		V_{CC} − 0.8	—	—	V
ILI	Input Leak Current	$V_{SS} \leqq V_{IN} \leqq V_{CC}$		—	—	± 10	μA
ILO	Output Leak Current	$V_{SS} + 0.4 \leqq V_{IN} \leqq V_{CC}$		—	—	± 10	μA
ICC1	Power Supply Current	V_{CC} = 5V f_{CLK} = (1) $V_{IHC} = V_{IH}$ = V_{CC}-0.2V $V_{ILC} = V_{IL}$ = 0.2V	AP-6/AM-6 /AT-6	—	6	10	mA
ICC2	Standby Supply Current	V_{CC} = 5V $V_{IHC} = V_{IH} = V_{CC}$-0.2V $V_{ILC} = V_{IL}$ = 0.2V		—	0.5	10	μA

100489

Note 1 : fCLK = 1/TcC (MIN.)

4.3 AC ELECTRICAL CHARACTERISTICS

4.3.1 AC characteritics (I)

When operate as peripheral devices (inactive state)

TA = −40℃ to 85℃, VCC = 5V ± 10%, VSS = 0V

NO.	SYMBOL	PARAMETER	AP-6/AM-6/AT-6 (6MHz)		UNIT
			MIN.	MAX.	
1	TcC	CLock cycle time	165	DC	ns
2	TwCh	High clock pulse width	65	DC	ns
3	TwCl	Low clock pulse width	65	DC	ns
4	TrC	Clock rise time	—	20	ns
5	TfC	Clock fall time	—	20	ns
6	Th	Hold time	0	—	ns
7	TsC (Cr)	$\overline{IORQ}$, $\overline{WR}$ and $\overline{CE}$ signals set-up time for clock rise	60	—	ns
8	TdD0 (RDf)	Delay from $\overline{RD}$ rise to data output	—	300	ns
9	TsWM (Cr)	Data input set-up time for clock rise (write and $\overline{M1}$ cycle)	30	—	ns
10	TdCf (D0)	Delay from $\overline{IORQ}$ fall to data output (INTA cycle)	—	100	ns
11	TsRD (Dz)	Delay from $\overline{RD}$ rise to data bus float state	—	70	ns
12	TsIEI (IORQ)	IEI set-up time for $\overline{IORQ}$ fall (INTA cycle)	100	—	ns
13	TdIEOr (IEIr)	Delay from IEI rise to IEO rise	—	70	ns
14	TdIEOf (IEIf)	Delay from IEI fall to IEO fall	—	70	ns
15	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall (When interrupt is generated immediately before $\overline{M1}$ cycle.)	—	100	ns
16	TsM1f (Cr)	$\overline{M1}$ signal set-up time for clock rise	70	—	ns
17	TsM1r (Cf)	$\overline{M1}$ signal set-up time for clock fall	−10	—	ns
18	TsRD (Cr)	$\overline{RD}$ signal set-up time for clock rise ($\overline{M1}$ cycle)	60	—	ns
19	TdI ($\overline{INT}$)	Delay from interruption generation to $\overline{INT}$ fall (at inactive state)	—	450	ns
20	TdBAIr (BAOr)	Delay from $\overline{BAI}$ rise to $\overline{BAO}$ rise	—	100	ns
21	TdBAIf (BAOf)	Delay from $\overline{BAI}$ fall to $\overline{BAO}$ fall	—	100	ns
22	TsRDY (Cr)	RDY signal set-up time for clock rise	50	—	ns

100489

4.3.2 AC charateristics (II) (1/2)

When operate as bus controller (active state)

TA = −40℃ to 85℃, VCC = 5V ± 10%, VSS = 0V

NO.	SUMBOL	PARAMETER	AP-6/AM-6/AT-6 (6MHz)		UNIT
			MIN.	MAX.	
1	TcC	Clock cycle time	165	DC	ns
2	TwCh	High clock pulse width	65	DC	ns
3	TwCl	Low clock pulse width	65	DC	ns
4	TrC	Clock rise time	—	20	ns
5	TfC	Clock fall time	—	20	ns
6	TdA	Delay of address output	—	90	ns
7	TdC (Az)	Delay from clock rise to address bus float state	—	80	ns
8	TsA (MREQ)	Address set-up time for $\overline{MREQ}$ fall (memory cycle)	35	—	ns
9	TsA (IRW)	Address set-up time for $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$ fall (I/O cycle)	110	—	ns
10	TdRW (A)	Address hold time from $\overline{RD}$, $\overline{WR}$ rise	35	—	ns
11	TdRW (Az)	Address hold time from $\overline{RD}$, $\overline{WR}$ rise (at float state)	65	—	ns
12	TdCf (D0)	Delay from clock fall to data output	—	130	ns
13	TdCr (Dz)	Delay from clock rise to data bus float state (write cycle)	—	70	ns
14	TsDI (Cr)	Data input set-up time up to clock rise (read cycle in which $\overline{RD}$ ended at clock rise)	30	—	ns
15	TsDI (Cf)	Data input set-up time up to clock fall (read cycle in which RD ended at clock fall)	40	—	ns
16	TsDO (WfM)	Data output set-up time up to $\overline{WR}$ fall (memory cycle)	25	—	ns
17	TsDO (WfI)	Data output set-up time up to $\overline{WR}$ fall (I/O cycle)	-55	—	ns
18	TdWr (D0)	Data hold time from WR rise	30	—	ns
19	Th	Hold time	0	—	ns
20	TdCr (Mf)	Delay from clock rise to $\overline{MREQ}$ fall	—	70	ns
21	TdCf (Mf)	Delay from clock fall to $\overline{MREQ}$ fall	—	70	ns
22	TdCr (Mr)	Delay from clock rise to $\overline{MREQ}$ rise	—	70	ns
23	TdCf (Mr)	Delay from clock fall to $\overline{MREQ}$ rise	—	70	ns

100489

AC characteristics (II) (2/2)

NO.	SYMBOL	PARAMETER	AP-6/AM-6/AT-6 (6MHz)		UNIT
			MIN.	MAX.	
24	TwMl	Low $\overline{MREQ}$ pulse width	135	—	ns
25	TwMh	High $\overline{MREQ}$ pulse width	65	—	ns
26	TdCf (If)	Delay from clock fall to $\overline{IORQ}$ fall	—	70	ns
27	TdCr (If)	Delay from clock rise to $\overline{IORQ}$ fall	—	65	ns
28	TdCr (Ir)	Delay from clock rise to $\overline{IORQ}$ rise	—	70	ns
29	TdCf (Ir)	Delay from clock fall to $\overline{IORQ}$ rise	—	70	ns
30	TdCr (Rf)	Delay from clock rise to $\overline{RD}$ fall	—	70	ns
31	TdCf (Rf)	Delay from clock fall to $\overline{RD}$ fall	—	80	ns
32	TdCr (Rr)	Delay from clock rise to $\overline{RD}$ rise	—	70	ns
33	TdCf (Rr)	Delay from clock fall to $\overline{RD}$ rise	—	70	ns
34	TdCr (Wf)	Delay from clock rise to $\overline{WR}$ fall	—	60	ns
35	TdCf (Wf)	Delay from clock fall to $\overline{WR}$ fall	—	60	ns
36	TdCr (Wr)	Delay from clock rise to $\overline{WR}$ rise	—	70	ns
37	TdCf (Wr)	Delay from clock fall to $\overline{WR}$ rise	—	70	ns
38	TwWl	Low $\overline{WR}$ pulse width	135	—	ns
39	TsWA (Cf)	$\overline{WAIT}$ signal set-up time for clock fall	60	—	ns
40	TdCr (B)	Delay from clock rise up to $\overline{BUSREQ}$ signal	—	90	ns
41	TdCr (Iz)	Delay from clock rise to $\overline{IORQ}$, $\overline{MREQ}$, $\overline{RD}$, $\overline{WR}$ signal float state	—	70	ns

AC Test conditions

100489

VIH = 2.4V, VIL = 0.4V, VIHC = VCC − 0.6V, VILC = 0.6V
VOH = 2.2V, VOL = 0.8V, CL = 100pF

4.4 CAPACITANCE

TA = 25°C

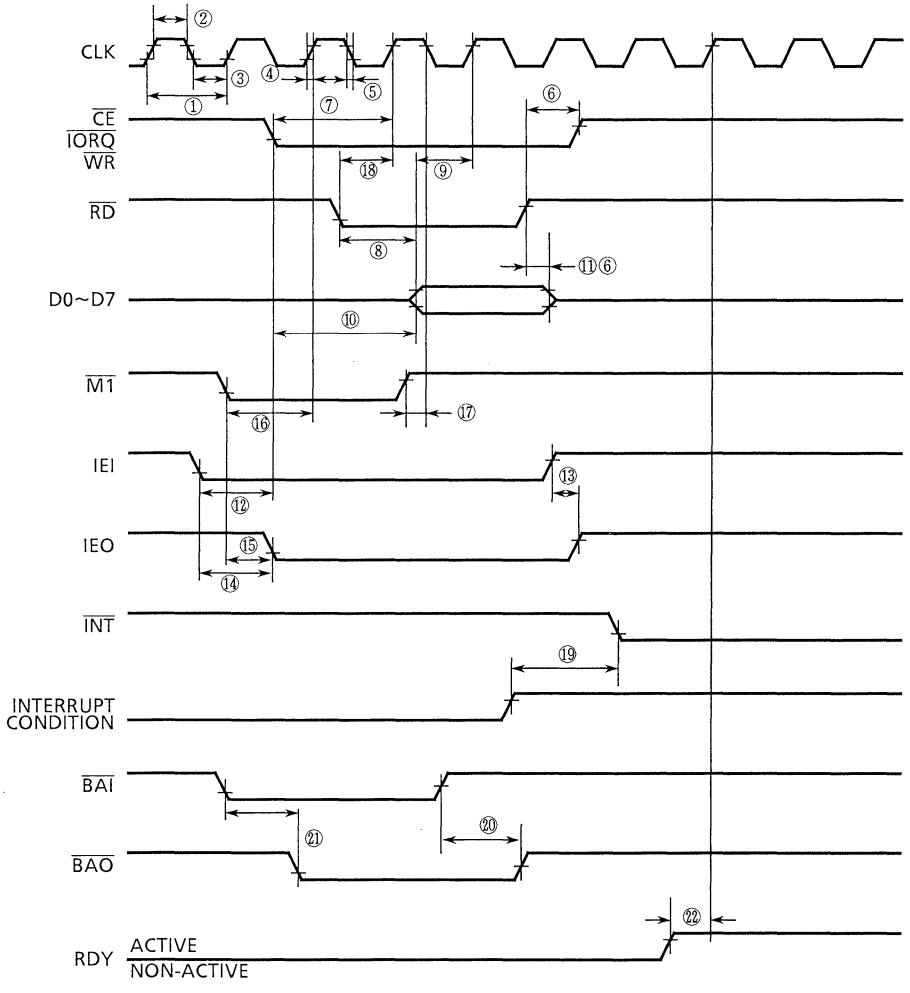
SYNBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	Clock Input Capacitance	f = 1MHz All terminals except that to be measured should be earthed.	—	—	5	pF
CIN	Input Capacitance		—	—	5	pF
COUT	Output Capacitance				10	pF

100489

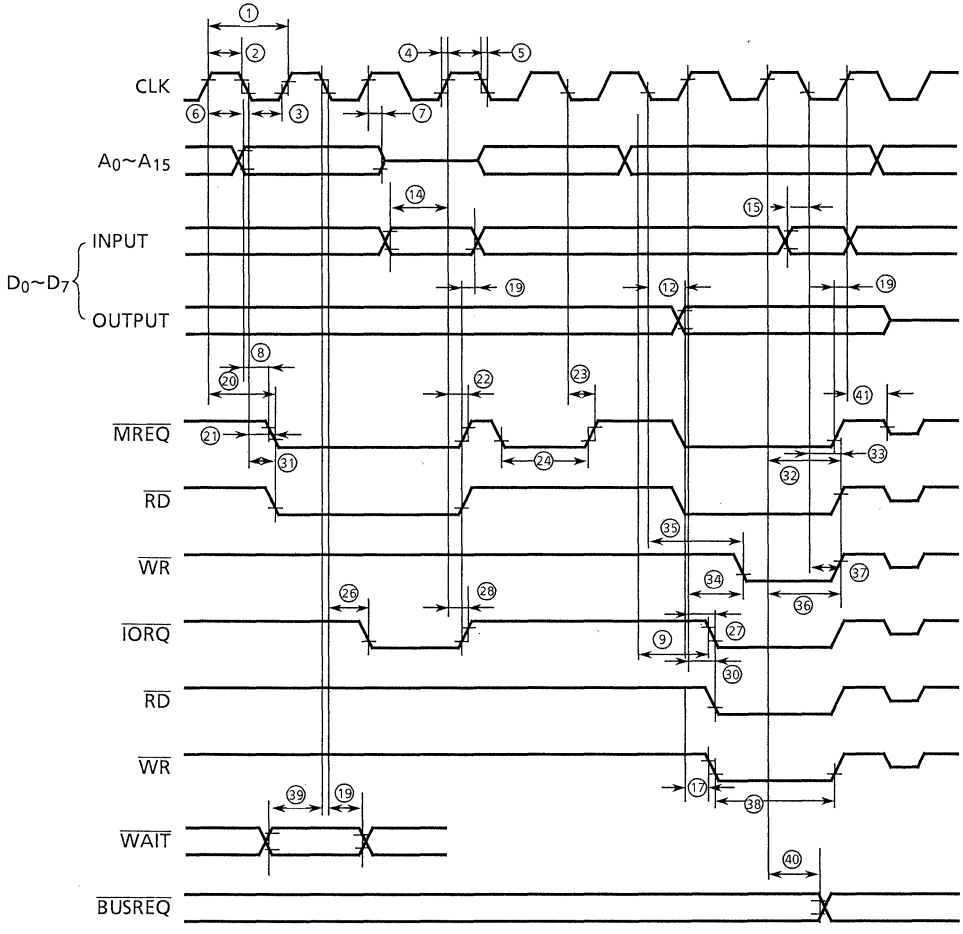
4.5 TIMING DIAGRAM

Numbers shown in the following figures correspond with those in the 4.3 A.C. Electrical Characteristics Table.

(1) When operate as peripheral devices (inactive state)



(2) When operate as bus controller (active state)



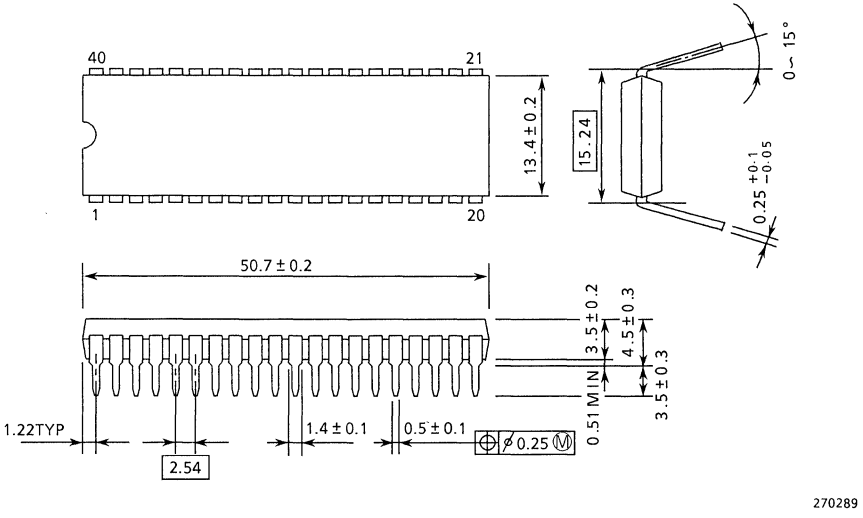
100489

5. EXTERNAL DIMENSION VIEW

5.1 DIP PACKAGE

DIP40-P-600

Unit : mm



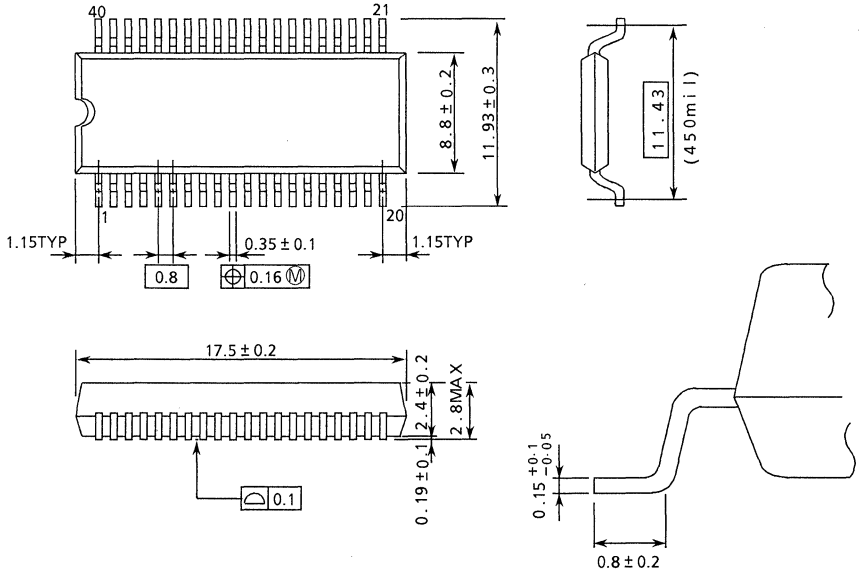
270289

Note 1 : This dimension is measured at the center of bending point of leads.
Note 2 : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.40 leads.

5.2 SOP PACKAGE

SSOP40-P-450

Unit : mm



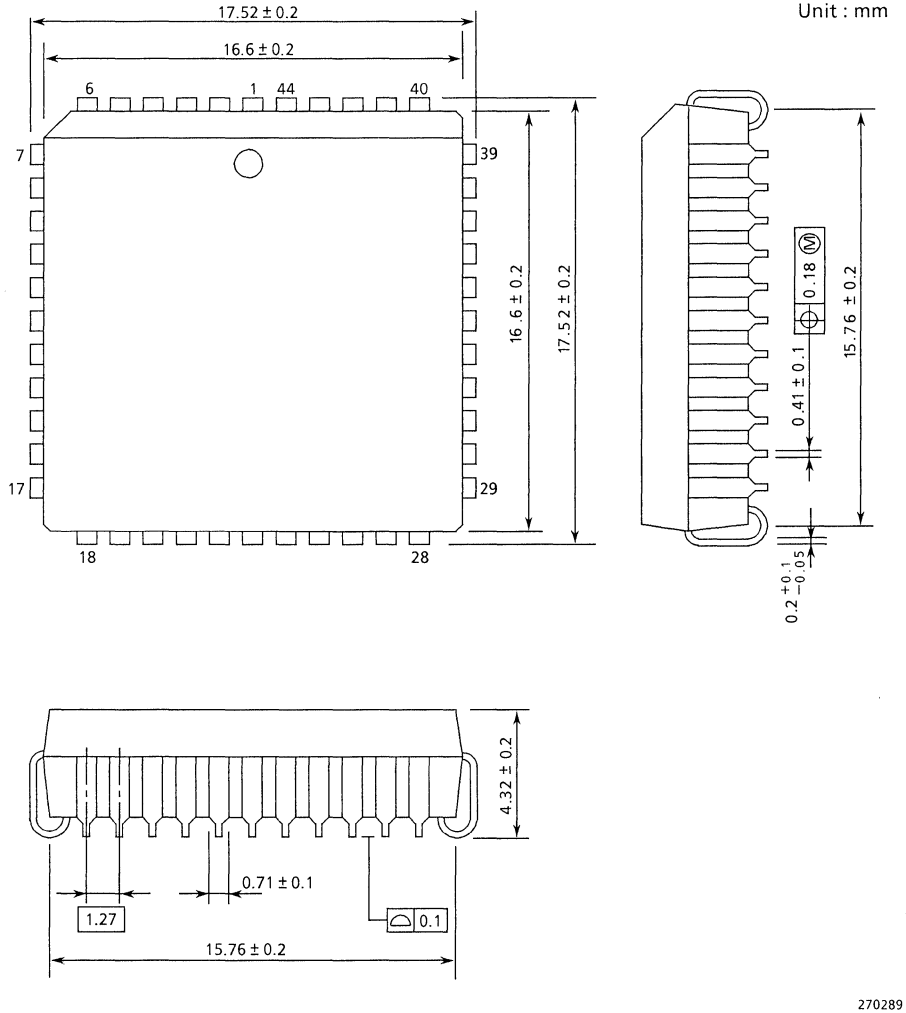
270289

Note : Package Width and length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

5.3 44-PIN PLCC PACKAGE

QFJ44-P-S650

Unit : mm



270289

6. PRECAUTIONS

No special care is required in the Z80 system designing employing DMA but it is necessary to fully understand the basic timings and commands shown in 3. Operational Description.

For the basic timings, please refer to the precautions in 3.3.2 (1).

For the commands, care shall be taken to the programming as there are many registers and setup is considerably complictated.

As the precaution for the programming, the "fixed address destinaion port programming" is briefly described. When Port A is assumed to be the "fixed address destination port", address can be loaded in the following stepe:

(1) Write Port A address into WR0.

(2) Designate Port A as the source port. (Set up temporarily)

(3) Load Port A address on the address counter. (Load command CFH)

(4) Write Port B start address into WR4.

(5) Convert Port A into the destination port.

(6) Load Port B start address on the address counter. (source port address)

TMPZ84C20AP-6 / TMPZ84C20AM-6 / TMPZ84C20AT-6
TMPZ84C20AP-8
TLCS-Z80 PIO : PARALLEL INPUT / OUTPUT CONTROLLER

1. GENERAL DESCRIPTION AND FEATURES

The TMPZ84C20A (hereinafter referred to as PIO) is CMOS version of Z80 PIO and has been designed to provide low power operation.

The PIO is a general purpose parallel input/output port device with two programmable independent 8-bit ports, which provides a direct interface between the Z80 microprocessor (hereinafter referred to as MPU) and peripheral devices.

This PIO provides excellent data transfer processing by the interrupt and allows the interrupt in Mode 2 of MPU.

The TMPZ84C20A is fabricated using Toshiba's CMOS Silicon Gate Technology.

The principal functions and features of the TMP84C20A are as follows.

(1) Compatible with the functions and pin connections of Zilog Z80 PIO.

(2) Low power consumption
3mA Typ. (@5V @6MHz) ··· TMPZ84C20AP-6/AM-6/AT-6
4mA Typ. (@5V @8MHz) ··· TMPZ84C20AP-8
10μA Max.(@5V, Stand-by)

(3) Operationg temperature
−40 °C to 85°C 6MHz VERSION
−10 °C to 70°C 8MHz VERSION

(4) DC to 6MHz operation ··· TMPZ84C20AP-6/AM-6/AT-6
DC to 8MHz operation ··· TMPZ84C20AP-8

(5) 2 programmable independent 8-bit input/output ports with handshake functions.

(6) 4 operation modes for each port:
Mode 0 (Byte Output Mode)
Mode 1 (Byte Input Mode)
Mode 3 (Byte Input/Output Mode) . . . . . Port A only
Mode 4 (Bit Mode)

(7) Built-in interrupt priority control circuit in daisy chain structure

(8) Port B outputs capable of driving Darlington transistors

(9) All input/output lines are TTL compatible.

(10) Single 5V power supply. Single-phase clock

(11) 40 pin DIP, SOP, 44pin PLCC Package.

Note : Z80 is a trademark of Zilog Inc., U.S.A.

2. PIN CONNECTIONS AND PIN FUNCTIONS

2.1 PIN CONNECTIONS (Top View)

The pin connections of the TMPZ84C20A are as shown in Figure 2.1, Figure 2.2.

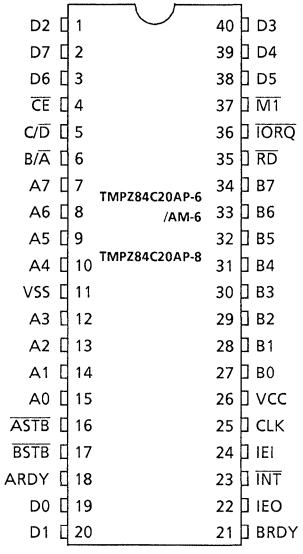


Figure 2.1 DIP, SOP Pin Connection

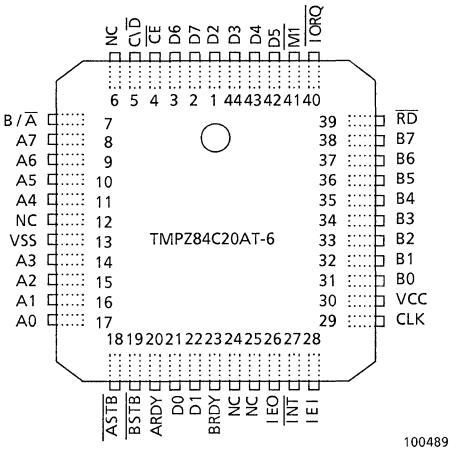


Figure 2.2 44-Pin PLCC Package

2.2 PIN NAMES AND FUNCTIONS

I/O pin names and functions are as shown in Table 2.1.

Table 2.1 Pin Names and Functions (1/3)

Pin Name	Number of Pin	Input/Output 3-state	Function
D0~D7	8	I/O 3-state	8-bit bidirectional data bus. Data transfer between MPU and PIO.
$\overline{CE}$	1	Input	Chip enable. Used for accessing MPU and PIO. When MPU selects this PIO, this terminal becomes L level (Refer to 3.4 Basic timing.) Normally, this terminal is connected to the address decoder output.
C/$\overline{D}$	1	Input	Control/data select. Indicates if signal on the data bus is control signal or data. Selects data at L level and command at H level. Normally, connected to address bit A1 of MPU.

100489

Table 2.1 Pin Names and Functions (2/3)

Pin Name	Number of Pin	Input/Output 3-state	Function
B/$\overline{A}$	1	Input	Port A/Port B select. Selects Port A at L level and Port B at H level. Normally, connected to address bit A0 of MPU.
A0~A7	8	I/O 3-state	Port A bus. Data transfer between Port A PIO and external device.
$\overline{ASTB}$	1	Input	Port A strobe input Handshake signal from the external device. Signal meaning differs depending upon operation mode. (Refer to 3.4 Basic timing.)
$\overline{BSTB}$	1	Input	Port B strobe input Handshake signal from the external device. Signal meaning is the same as $\overline{ASTB}$ but differs if Port A is in Mode 2. (Refer to 3.4 Basic timing.) Register A ready.
ARDY	1	Output	Port A ready. Handshake signal to the external device. Signal meaning differs depending upon operation mode. (Refer to 3.4 Basic timing.)
$\overline{M1}$	1	Input	Machine cycle 1. When both $\overline{M1}$ and $\overline{IORQ}$ are at L level, indicates that MPU is executing the interrupt acknowledge cycle. (Refer to 3.4 Basic timing.) Normally, connected to $\overline{M1}$ of MPU.
$\overline{IORQ}$	1	Input	I/O request. Used to access between MPU and PIO. This terminal becomes L level when I/O addresses are on the address in the write cycle and read cycle. Further, when $\overline{IORQ}$ and $\overline{M1}$ are both at L level, it indicates that MPU is executing the interrupt acknowredge cycle. (Refer to 3.4 Basic timing.) Normally, connected to $\overline{IORQ}$ of MPU.
$\overline{RD}$	1	Input	Read signal. Used to access between MPU and PIO. Controls the transfer direction. (Refer to 3.4 Basic timing) Normally, connected to $\overline{RD}$ of MPU.
B0~B7	8	I/O 3-state	Port B bus. Data transfer between Port B of PIO and external device. Capable of driving -1.5mA (@VoH = 1.5V) Darlington transistors.
CLK	1	Input	System clock. Signal-phase clock input. In DC state (either at H or L level), PIO is in a stand-by state and power consumption becomes extermely less.

100489

Table 2.1 Pin Names and Functions (3/3)

Pin name	Number of Pin	Input/Output 3-state	Function
IEI	1	Input	Interrupt enable input. Together with IEO and $\overline{INT}$, forms daisy chain interrupt control signal. Connected to IEO of high priority peripheral LSI. However, to give higher priority than other peripheral LSI's to this PIO, connect this terminal to the + 5V power. (Refer to 3.3.2 Interruption.)
$\overline{INT}$	1	Output	interrupt request. interrupt request signal for MPU. Connect to $\overline{INT}$ of MPU. (Open drain)
IEO	1	Output	Interrupt enable output. Together with IEI and $\overline{INT}$, forms daisy chain interrupt control signal. Connected to IEI or low prioirity peripheral LSI. However, if this PIO has the lowest priority than any other peripheral LSI's, this IEO is not used. (Refer to 3.3.2 Interruption.)
BRDY	1	Output	Port B ready Handshake signal to the external device. Signal meaning is the same as that of ARDY. However, it differs when Port A is in Mode 2. (Rerfer to 3.4 Basic timing.)
V_{CC}	1	Power supply	+ 5V
V_{SS}	1	Power supply	0V

100489

3. FUNCTIONAL DESCRIPTION

3.1 PIO BLOCK DIAGRAM

Figure 3.1 shows the PIO block diagram.

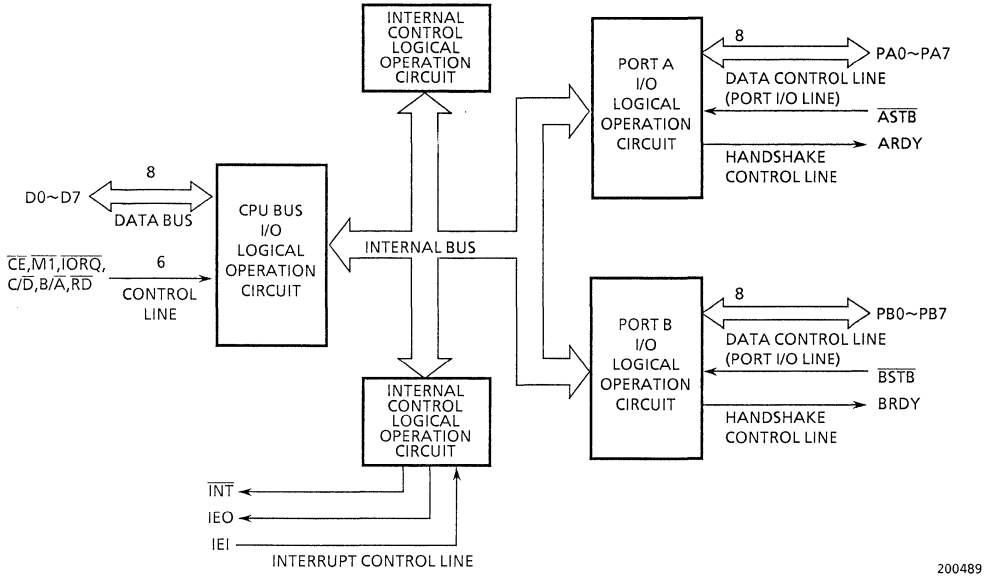


Figure 3.1 PIO Block Diagram

3.2 PIO SYSTEM CONFIGURATION

The PIO system consists of the four logic circuits:

(1) MPU bus I/O logic circuit

(2) Internal control logic circuit

(3) Interrupt control logic circuit

(4) Port I/O logic circuit

[1] MPU Bus I/O Logic Circuit

The MPU bus I/O logic circuit transfers data between the MPU and the PIO.

[2] Internal Control Logic Circuit

The internal control logic circuit controls the PIO operating functions like the PIO selecting chip enable and the read/write circuits.

[3] Interrupt Control Logic Circuit

The interrupt control logic circuit performs the MPU interrupt-associated processing such as determining interrupt priorities. The priorities with other LSI peripherals are determined by the physical location chain connection.

[4] Port I/O Logic Circuit

The port I/O logic circuits are used to directly connect to peripheral devices. Each consists of the following 7 registers and 1 flip-flop. Data are written in the registers by the MPU as specified in the program. Figure 3.2 shows the internal configuration of the ports.

- Data output register (8 bits)
- Data input register (8 bits)
- Mode control register (2 bits)
- Interrupt vector register (8 bits)
- Interrupt control register (2 bits)
- Mask control register (8 bits)
- Data I/O control register (8 bits)
- Handshake control logic circuit

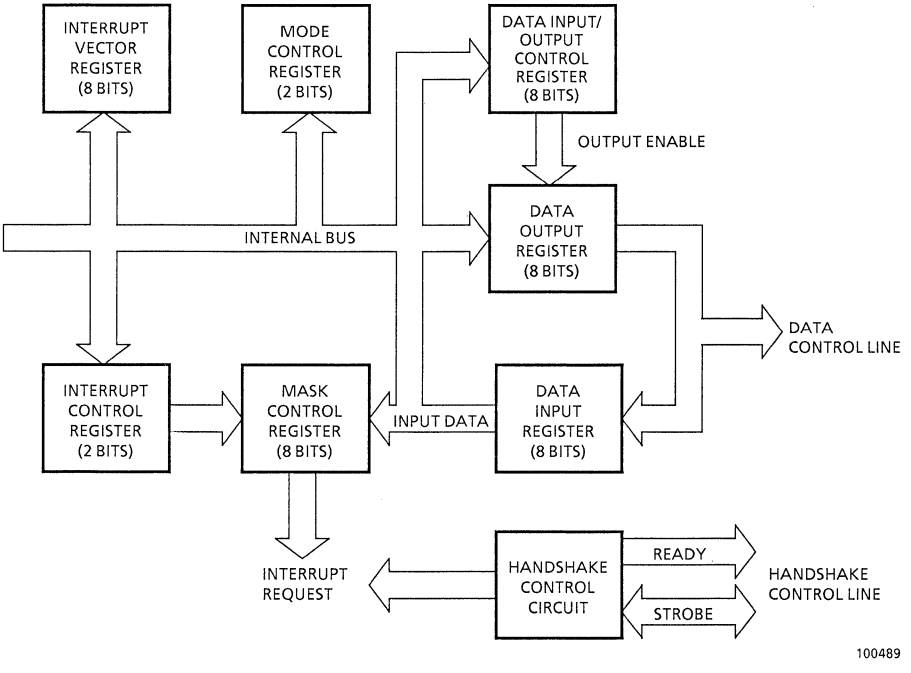


Figure 3.2 Port Internal Configuration

(1) Data Output Register

This register holds the data to be teansferred from the MPU to peripheral devices.

(2) Data Input Register

This register latches the data to be transferres from peripheral devices to the MPU.

The input data to the MPU is read through this register.

(3) Mode Control Register

This register specifies the operation mode. The operation mode is set by MPU control.

(4) Interrupt Vector Register

This register holds the vector which makes up the address of the table storing the start adress of the interrupt processing routine.

This register is used only for interrupt processing.

(5) Interrupt Control Register

This register specifies how the I/O ports are to be monitored. This register is used only in the PIO mode 3.

(6) Mask Control Register

This register specifies which I/O port pin is to be monitored. This register is used only in the PIO mode 3.

(7) Data I/O Control Register

This register specifies whether each port pin is to be used as output or input. This register is used only in PIO mode 3.

(8) Handshake Control Logic

This cirsuit controls the data transfer to the peripheral devices connected to the 8-bit I/O ports.

3.3 PIO BASIC OPERATIONS

[1] Reset

The PIO provides the following two reset capabilities:

When PIO is connected with the MPU (TMPZ84C00A, 01A, 02A) of Z80 series, or ASSP (TMPZ84C011A, 013A, 015A, 710A, 810A), it is necessary to connect with external logic as Figure 3.3.

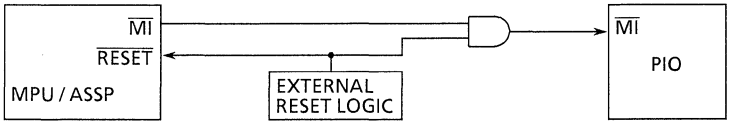


Figure 3.3 External Signal Reset Logic

(1) Power-on Reset

The PIO contains the circuit which automatically resets the PIO at the time of power-on.

(2) Hardware Reset

Making the $\overline{M1}$ pin low for 2 system clock periods with the $\overline{RD}$ and $\overline{IORQ}$ pins being high resets the PIO on the rising edge of the $\overline{M1}$ pin.

Reset State

(a) The operation mode is set to mode 1 for both ports.
(b) Interrupt is disabled.
(c) All the bits of the data I/O register of each port are reset.
(d) All the bits of the mask control register of each port are set and masked.
(e) The port I/O lines of each port are put in the high-impedance state (floating).
(f) The RDY pin of each port goes low.

The reset state is held until the control word is written. For the function of the control word, see Subsection 3.5 "Operational Procedure".

[2] Interrupt

The PIO can cause an interrupt when the MPU is operating in mode 2. The interrupt request signal ($\overline{INT}$) from the PIO is accepted when the MPU is in the inetrrupt enabled state (caused after the execution of EI instruction). Receiving the $\overline{INT}$ signal, the MPU latches the interrupt vector (8-bit data) sent from PIO, specifies the start address of the interrupt processing routine based on the vector, and calls the routine to start the processing.

Thus, since the start address of the interrupt processing routine can be specified by the interrupt vector sent from the PIO, the user can change the vector value to call any desired address.

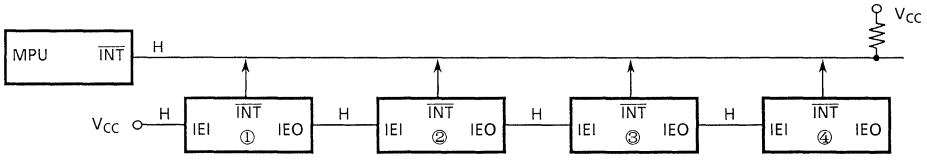
Interrupt processing is terminated when the MPU executes the RETI instruction. The PIO has the circuit to decode the RETI instruction to detect the termination of interrupt processing by constantly monitoring the data bus.

The interrupt priority among the Z80 peripharal LSIs is determined by the daisy chain structure. In daisy chain, the peripheral LSIs are connected one after another as shown in Figure 3.4. The more a peripheral LSIs is physically located near the MPU, the higher the priority of the peripheral is. Within the PIO, port A is given higher priority than port B. The Z80 peripheral LSIs have the signal lines IEO and IEI connected to the IEO of a higher peripheral LSIs and IEI of a lower peripheral LSI respectively. However, the IEI of the highest peripheral LSIs is connected to the IEI pin and the IEO of the lowest peripheral LSI is connected to the IEO pin. In this state, the PIO interrupt follows the conditions:

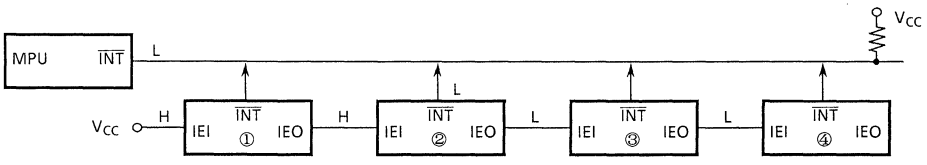
- When both IEI and IEO are high, no interrupt has occurred. This time, the interrupt request signal ($\overline{INT}$) is high. In this state, the PIO can request interrupt.
- When the PIO sends the $\overline{INT}$ signal, it sets the IEO line to the low level. When the interrupt request is accepted by the MPU, $\overline{INT}$ goes back to the low level.
- When the IEI goes low, the IEO also goes low.
- When the IEI is low, the PIO cannot request an interrupt.
- If the IEI goes low during interrupt occurrence, the interrupt processing is suspended.

The operations of the four Z80 peripheral LSIs (the states of IEI, IEO and $\overline{INT}$ signal) daisy-chained as shown in Figure 3.4 are as follows:

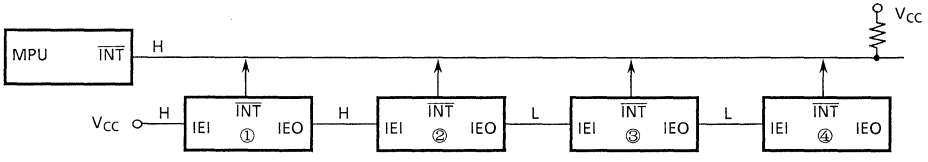
(1) Before interrupt occurrence



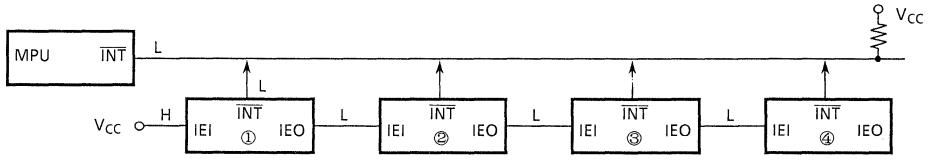
(2) Interrupt request from LSI-2 to the MPU



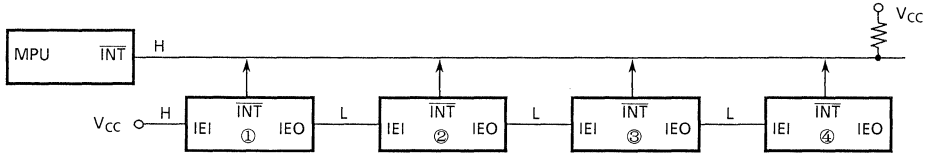
(3) The MPU acknowledges (enables) the interrupt. Interrupt processing for LSI-2 is performed.



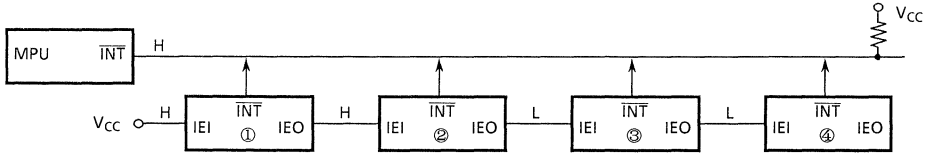
(4) Interrupt request from LSI-1 to the MPU. The interrupt processing for LSI-2 is suspended.



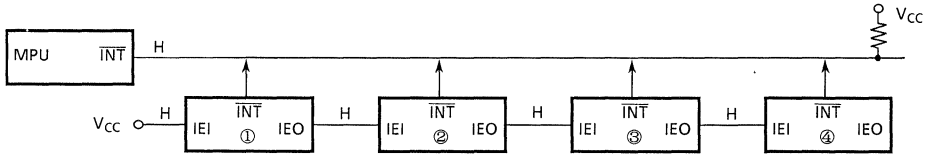
(5) The MPU acknowledges (enables) the interrupt. Interrupt processing for LSI-1 is performed.



(6) Interrupt processing for LSI-1 terminates (upon execution of the RETI instruction). Interrupt processing for LSI-2 is restarted.



(7) Interrupt processing for LSI-2 terminates (upon execution of the RETI instruction).



Interrupt priority is given to LSI-1, LSI-2, LSI-3 and LSI-4 in this order.

100489

Figure 3.4 Signal States in Daisy Chain Structure

[3] Operation Modes

The PIO operates in one of the 4 operation modes. The mode is selected by writing the mode control word.

- Mode 0 (byte output mode)
- Mode 1 (byte input mode)
- Mode 2 (byte I/O mode)
- Mode 3 (bite mode)

(1) Mode 0 (byte output mode)

In mode 0, the PIO sends the data received from the MPU to the external device through the port data output register. The contents of this register can be rewritten by using an output instruction. If the data on the bus change, the register contents remain unchanged until the next output instruction is executed. When the MPU executes an output instruction, the write signal is generated in the PIO in the write cycle. Using the signal, data on the data bus can be latched in the data output register.

(2) Mode 1 (byte input mode)

In this mode, the PIO sends the data received from the external device to the MPU through the port data input register. The data transfer to the MPU is suspended until the MPU has read the current data.

(3) Mode 2 (byte I/O mode)

Mode 2 is a combination of mode 0 and mode 1. This mode is used only for port A. In this mode, all 4 handshake control lines are used. Port A's handshake control lines are used for data output and the port B's handshake control lines are used for data input. For data transfer, port A is used. Port B is set in mode 3 (bit mode) in which no handshake control line is used.

In this mode, the interrupt timing occurs almost at the same time in mode 0 and mode 1. In an input operation, the port B's handshake control lines are used, so that the interrupt vetor written in port B is transferred. Therefore, the interrupts in input and output can be controlled by different vectors.

(4) Mode 3 (bit mode)

In mode 3, the 8-bit port I/O lines are controlled for each bit. Since no handshake control lines are used, ordinaly read/write operations can be performed. I/O operations can be performed on the port as well. In a write operation, the data sent from the MPU to the PIO are latched in the data output register corresponding to the bit set for output in the same timing as in mode 0.

An interrupt occurs in the interrupt enabled state and when the bit set for input satisfies the condition specified in the interrupt control word. However, if port A is operating in mode 2, port B cannot cause an interrupt in the bit mode. Note that, to use the interrupt capability, the mask control register bit corresponding to the bit set for output must be set to "1" to disable its interrupt.

3.4 PIO STATUS TRANSITION AND BASIC TIMING

[1] Status Transition

Figure 3.5 shows the PIO status transition diagram.

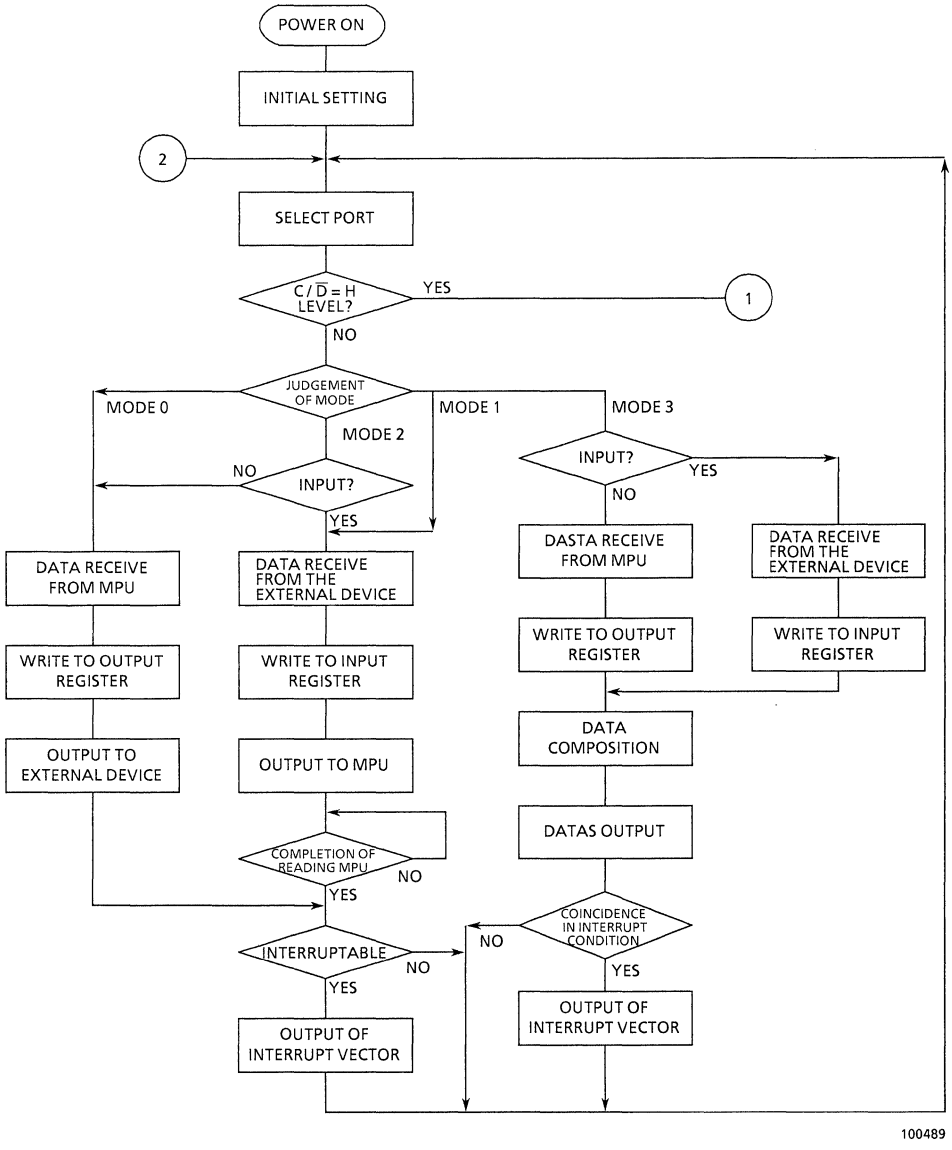


Figure 3.5 (a) PIO Status Transition

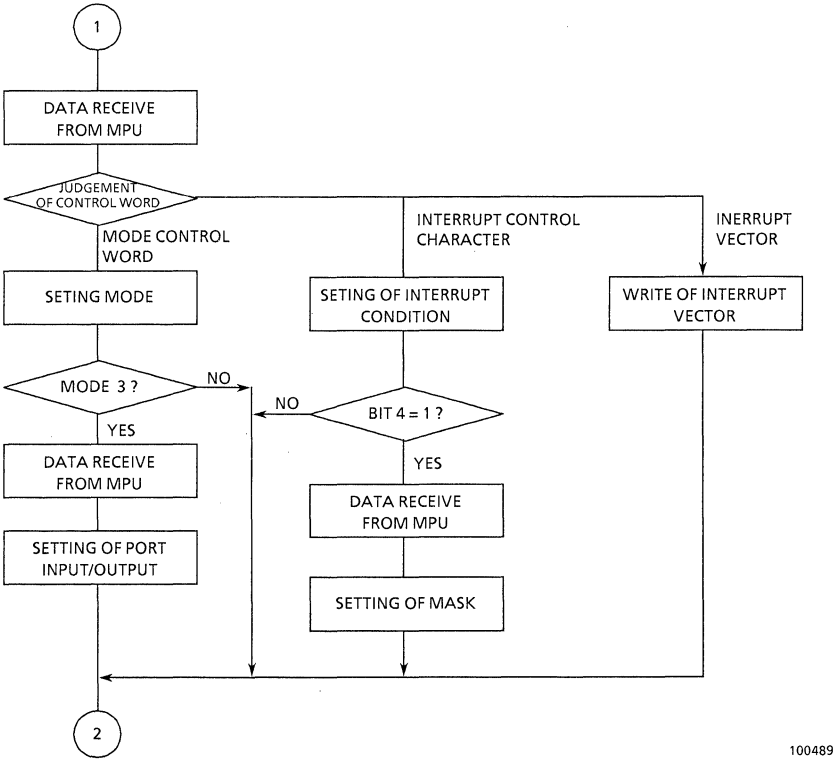


Figure 3.5 (b) PIO Status Transition

[2] Write Cycle

The $\overline{IORQ}$, $\overline{RD}$, C/$\overline{D}$ (A0), and $\overline{CE}$(A7 through A0) signals generate the write signal (*$\overline{WR}$) inside the PIO.

The MPU sets the PIO's $\overline{IORQ}$ signal to the low level at system clock T2, to start the write cycle. At this time, to indicate that this cycle is a write sycle, the PIO's $\overline{M1}$ signal must be set to the high level. At the same time, the MPU sends signals to the PIO's B/$\overline{A}$ (A1) and C/$\overline{D}$ (A0) to specify the port or select control signal or the data. This allows the port data ouput register of the PIO's selected port to latch the data at system clock T3. TW is a wait state automatically inserted by the MPU.

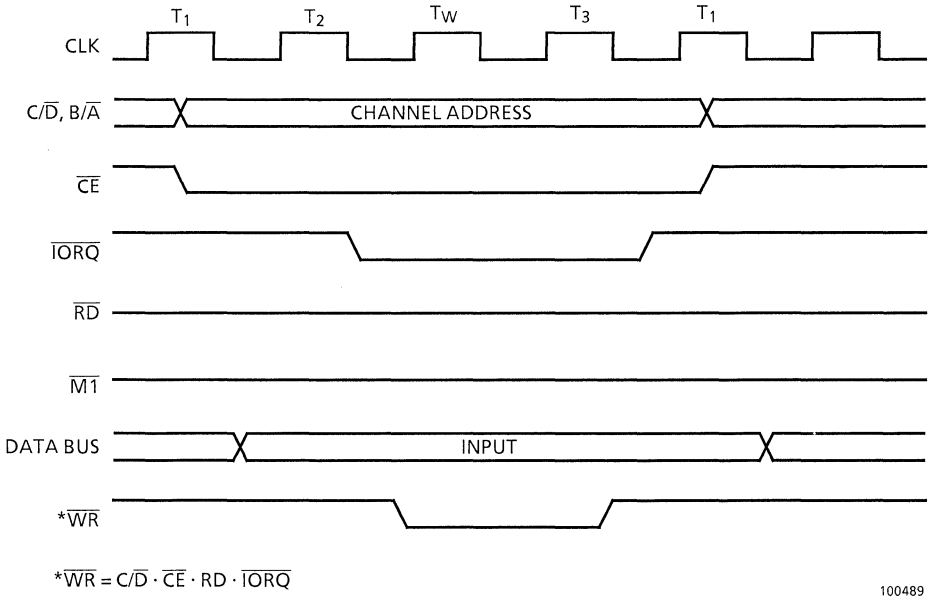


Figure 3.6 Write Cycle Timing

[3] Read Cycle

The MPU sets the PIO's, $\overline{RD}$ pin, $\overline{CE}$ signal, and $\overline{IORQ}$ pin to the low level at system clock T2 to start the read cycle. At this time, to indicate that this cycle is a read cycle, the PIO's M1 pin must be set to the high level. The PIO outputs data in the $\overline{CE}$, $\overline{IORQ}$, and $\overline{RD}$ signals. TW is a wait state automatically inserted by the MPU.

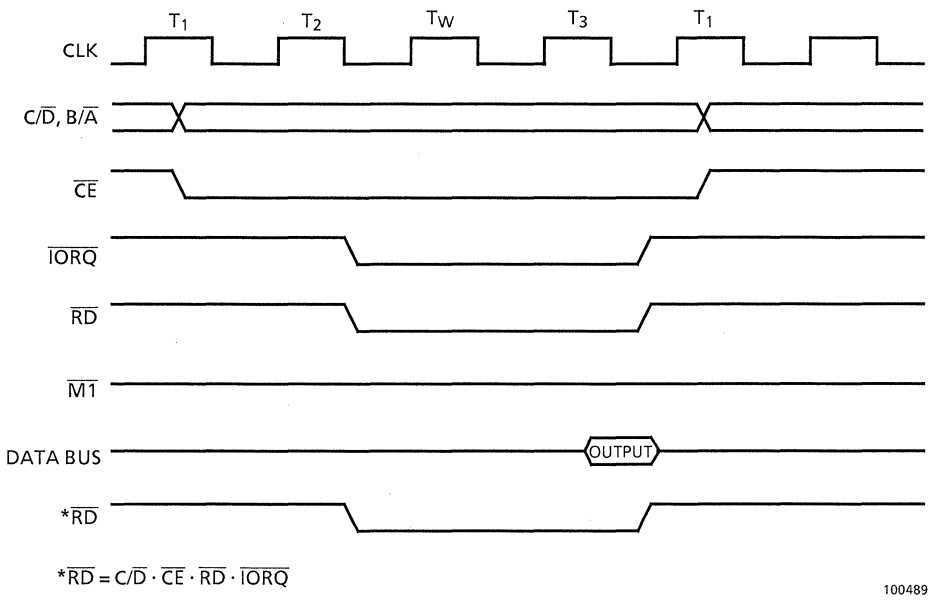


Figure 3.7 Read Cycle Timing

[4] Mode 0 (Byte Output Mode)

The mode 0 output cycle starts when the MPU executes an output instruction. When an output instruction is executed, he write signal (*$\overline{WR}$) is generated in the PIO in the write cycle. This signal latches the data on the data bus to the data output register of the selected port. The RDY pin goes high on the first falling edge of the system clock after the rise of the write signal (*$\overline{WR}$). This indicates that the data in the data output register are already on the port I/O pin. The peripheral device sets the RDY pin to the low level on the first falling edge of the system clock after the rise of the $\overline{STB}$ pin to be input to the PIO to indicate that the peripheral device has received the data from the port I/O pin, waiting for the next output instruction. If, at this time, the PIO is enabled for interrupts, it sets the $\overline{INT}$ pin to the low level on the rising edge of the $\overline{STB}$ signal to output the interrupt request signal to the MPU. Figure 3.8 shows the timing chart of mode 0.

[5] Mode 1 (Byte Input Mode)

The input cycle starts when the MPU has completed the previous data read operation. The peripheral device sets the PIO's $\overline{STB}$ pin to the lower level, putting data on the port I/O line.

The RDY pin is driven low on the first falling edge of the system clock after the rise of the $\overline{STB}$ pin, disabling the peripheral device to send the next data. If, at this time, the

PIO is enabled for interrupts, it sets the $\overline{INT}$ pin to the low level on the rising edge of the $\overline{STB}$ pin, making an interrupt request to the MPU. When the MPU executes the input instruction in the interrupt processing routine, the read signal (*$\overline{RD}$) is generated in the PIO in the read cycle. This signal puts the data in the data input register of the selected port on the data bus. The MPU receives this data. The PIO sets the RDY pin to the high level on the first falling edge of the system clock after the rise of the read signal (*$\overline{RD}$) to wait for the input of the next data. Figure 3.9 shows the mode 1 timing chart.

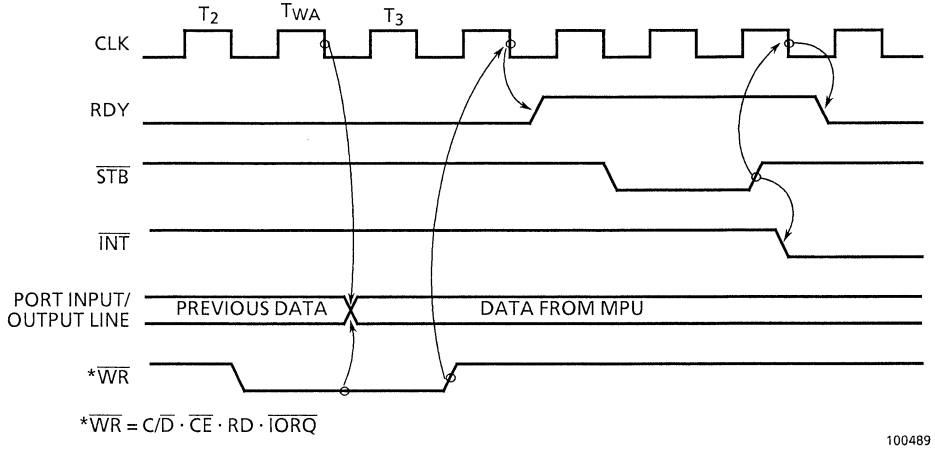


Figure 3.8 Mode 0 Timing Chart

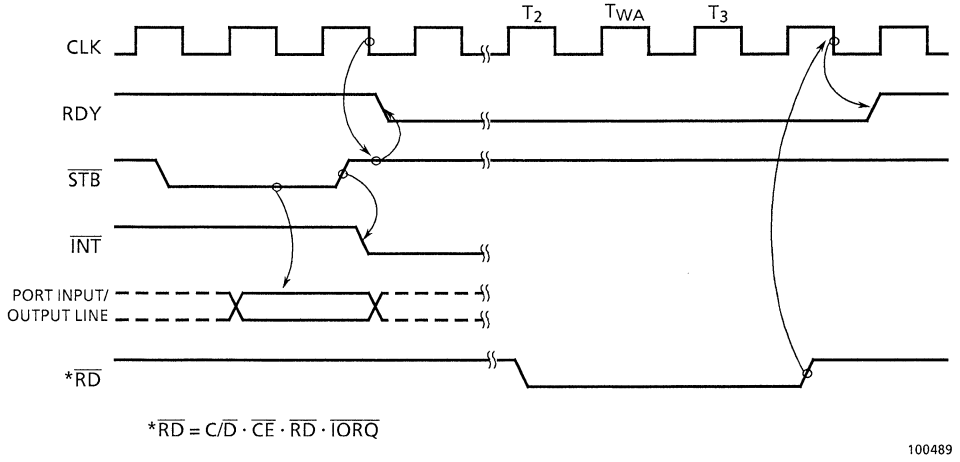


Figure 3.9 Mode 1 Timing Chart

[6] Mode 2 (Byte I/O Mode)

Mode 2 is a combination of mode 0 and mode 1. The timing for output operation is generally the same as in mode 0 except that, in mode 2, data is output only when the $\overline{ASTB}$ pin is low while, in mode 0, data is always on the port I/O line. The peripheral device can receive data on the rising edge of the $\overline{ASTB}$ signal being used as the latch signal.

The input timing is the same as in mode 1.

The port A handshake line is used as output control and the port B handshake line is as input control.

The value of the interrupt vector generated by the $\overline{BSTB}$ signal during port A input operation is the same as the value of the interrupt vector generated when port B is used in mode 3. Hence, all port B bits are masked by setting the mask control word to disable port B for the interrupt capability.

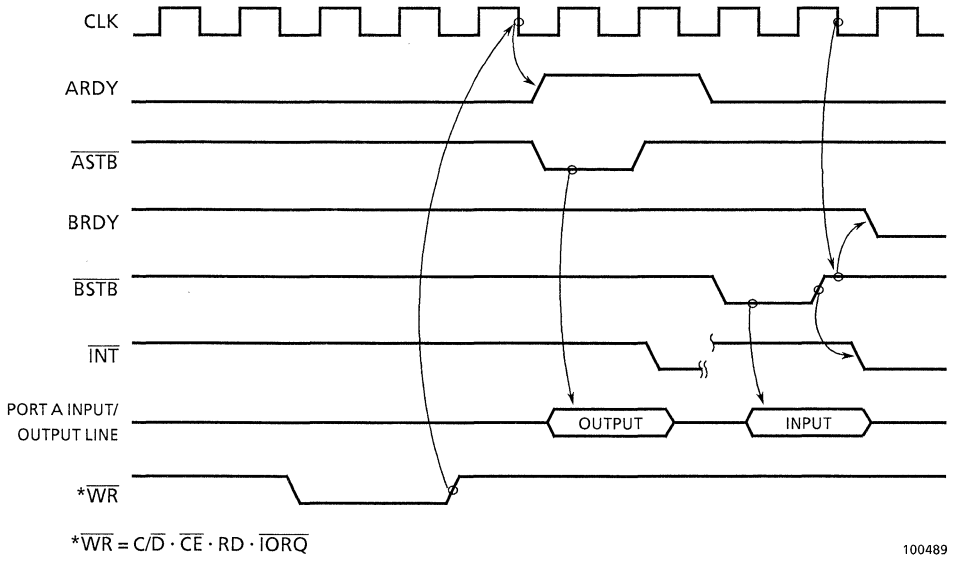


Figure 3.10 Mode 2 Timing Chart

[7] Mode 3 (Bit Mode)

In this mode, no handshake line is used. Therefore, the ordinary port read/write operations can be performed, permitting access to the ports any time. The write data from the MPU is latched to the data output register corresponding to the bit set for output in the same timing as in mode 0. Except when port B is used in mode 2, the $\overline{STB}$ pin of the port operating in mode 3 is fixed to the low level. The transfer data consists of the data in the data output register and in the data input register. That is, the data of the bit set for output and the data of the bit set for input make up the transfer data.

An interrupt occurs when the interrupt enabled state is on and the bits set for input safety the condition specified by the mask control word, etc. However, if port A is operating in mode 2, port B is disabled for interrupt in the bit mode. Note that, to use the interrupt capability, the bit on the mask register corresponding to the bit set for output must be set to "1" to disable it for interrupts.

An interrupt request occurs when the logic condition becomes true. If the logic condition becomes true immediately before the $\overline{M1}$ pin becomes low or while $\overline{M1}$ pin low, an interrupt request occurs on the rising edge on the $\overline{M1}$ pin.

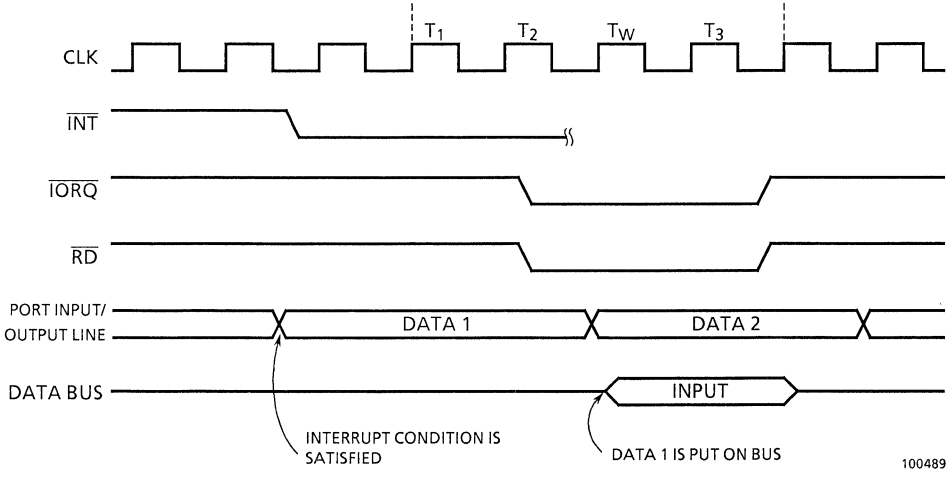


Figure 3.11 Mode 3 Timing Chart

[8] Interrupt Acknowredge Cycle

Outputting the interrupt request signal ($\overline{INT}$), the PIO sets the IEO signal to the low level, disabling the low-priority peripheral LSIs for interrupt requests.
Receiving the interrupt request signal ($\overline{INT}$) from the PIO, the MPU sets the PIO's $\overline{M1}$ and $\overline{IORQ}$ pins to the low level to indicate that the MPU has acknowledged the interrupt request. The $\overline{IORQ}$ pin goes low 2.5 system clocks later than the $\overline{M1}$ pin. To stabilize the daisy-chained signal lines (IEI and IEO), the ports and peripheral LSIs cannot change the interrupt request.

The $\overline{RD}$ pin remaines high to make distinction between the instruction fetch and interrupt acknowledge cycles. While the $\overline{RD}$ pin is high, the interrupt control logic in the PIO determines the interrupt requesting port of the highest priority. When the $\overline{IORQ}$ pin goes low with the IEI pin being high, the interrupt vector is put on the data bus from the interrupt requesting port. At the same time, two system clocks are automatically inserted by the MPU as a wait state to stabilize the daisy chain structure.

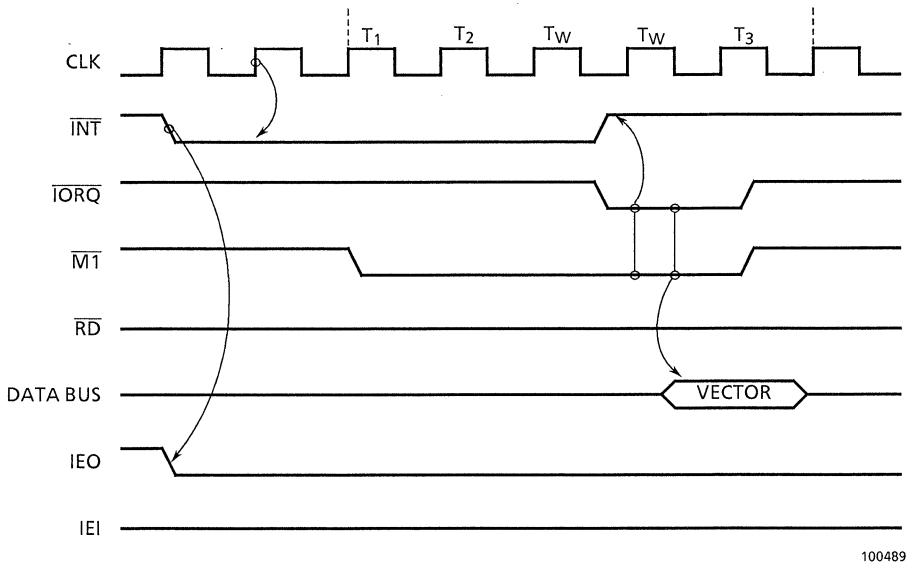


Figure 3.12 Interrupt Acknowledge Cycle Timing Charts

[9] Return from Interrupt Cycle

Return from interrupt processing is performed when the MPU executes the RETI instruction. This RETI instruction must be used at the end of the interrupt processing routine. When the MPU executes this instruction, the PIO's IEI and IEO return to the states active before interrupt processing.

The RETI instruction consists of two bytes and its code EDH and 4DH. The PIO decodes the RETI instruction to determine whether there is any interrupt requesting port. In the daisy chain structure, the IEI and IEO of the interrupting LSI remain high and low respectively at the time the instruction code EDH has been decoded. If the code following EDH is 4DH, only the peripheral LSI which has sent an interrupt vector immediately before, that is, the LSI whose IEI is high and IEO is low, returns from interrupt processing. This restarts the interrupt processing of the supended peripheral LSIs of lower interrupt priory.

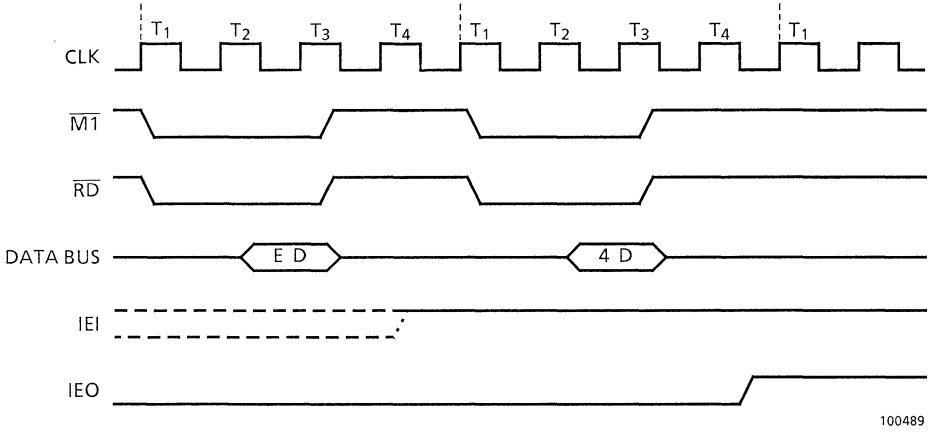


Figure 3.13 Interrupt Cycle Return Timing Chart

3.5 PIO OPERATIONAL PROCEDURE

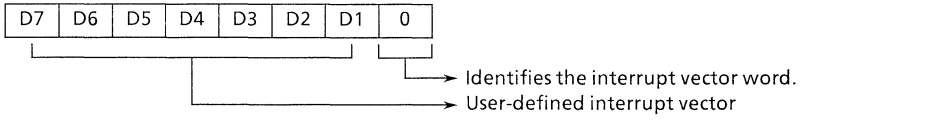
To operate the PIO the control words shown below must be written in it as the initial settings. They must be wrriten in the PIO's ports, A and B, separately. Spesify the I/O address listed in Table 3.1 to write control words and data in the PIO.

Table 3.1 I/O Adreesses

I/O function	B/$\overline{A}$	C/$\overline{D}$	$\overline{CE}$
Port A data	0	0	0
Port B command	0	1	0
Port B data	1	0	0
Port B command	1	1	0

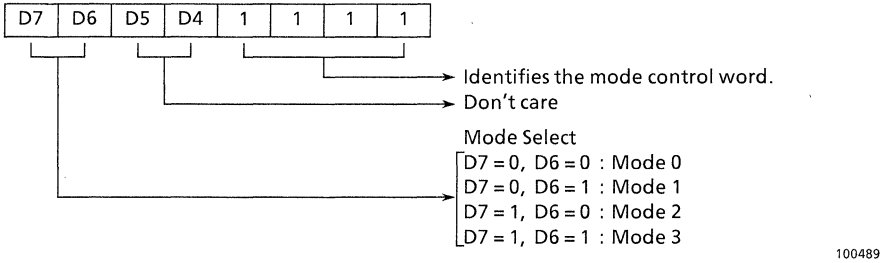
100489

(1) Interrupt Vector Word



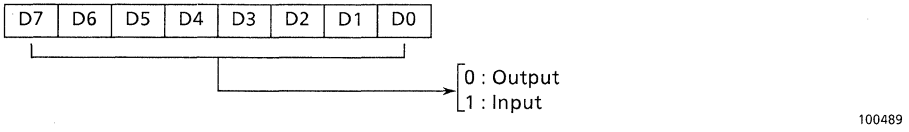
- Using this vector and the contents of the address indicated by the MPU's register, the MPU generates the start address of the interrupt processing routine.
- D0 through D7 are written in the interrupt processing register.
- This word is not needed when the interrupt capability is not used.

(2) Mode Control Word



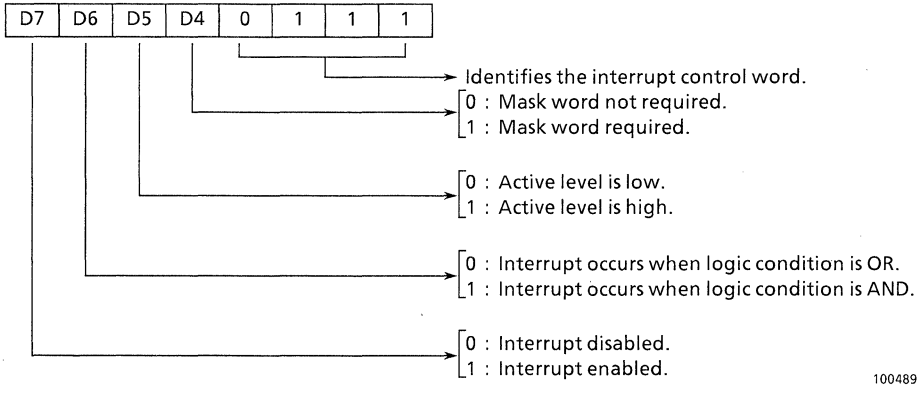
- This word specifies an operation mode.
- D7 and D6 are written in the mode control register.

(3) Data I/O Control Word



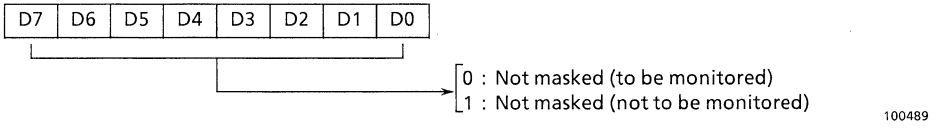
- This word is needed only in mode 3.
- When mode 3 is specified by the mode control word, the data I/O control word is written after it.
- Each port is specified for output or input.
- D0 through D7 are written in the data I/O register.

(4) Interrupt Control Word



- This word is for interrupt control such as interrupt condition setting.
- D4, D5, and D6 are used only in mode 3.
- With D6 = 0, interrupt occurs when one of the bits not masked (the bit to be monitored) by the mask control word goes active.
- With D6 = 1, interrupt occurs when all bits not masked (the bit to be monitored) by the mask control word goes active.
- With D4 = 1, the suspended interrupts are all reset regardless of the mode.
- D5 and D6 are written in the control register.

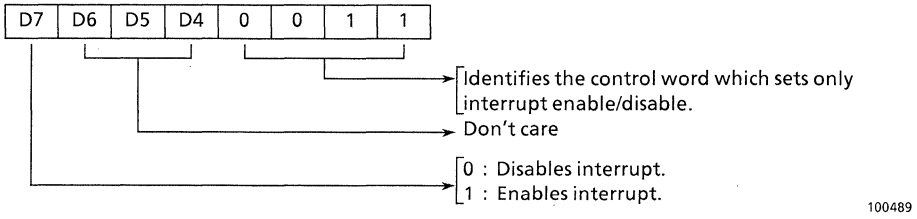
(5) Mask Control Word



- This word is needed only in mode 3.
- When D4 = 1 is set by the interrupt control word, the mask control word must be written after it.
- This word specifies whether to monitor the port I/O line specified for input by the data I/O control word.
- When the bit is set to 0, the corresponding input line is monitored and regarded as the input associated with interrupt occurrence.
- When the bit is set to 1, the corresponding input line is masked to provide the input not related to interrupt occurrence.
- The PIO checkes only the input line with the bit being 0 to see if the interrupt condition is satisfied. If the condition is satisfied, the PIO requests an interrupt.
- D0 through D7 are written in the mask control register.

When port A is put in mode 2, all 4 handshake lines are used, so that port B must be set in mode 3 which uses no handshake lines. At the same time, all mask control word bits must be set to 1 (masked).

Note : Only interrupt enable/disable can be set by the following control word:



3.6 USING PIO

The following is a programming example to operate the PIO's port in mode 3. This program is followed by the main routine and the interrupt processing program.

- The MPU is used in the mode 2 interrupt.
- The table storing the start address of the interrupt processing routine is 0802H.
- Interrupts occur when both PIO's port input lines A6 and A5 go high.
- The I/O addresses of the PIO are the address listed in Table 3.1.

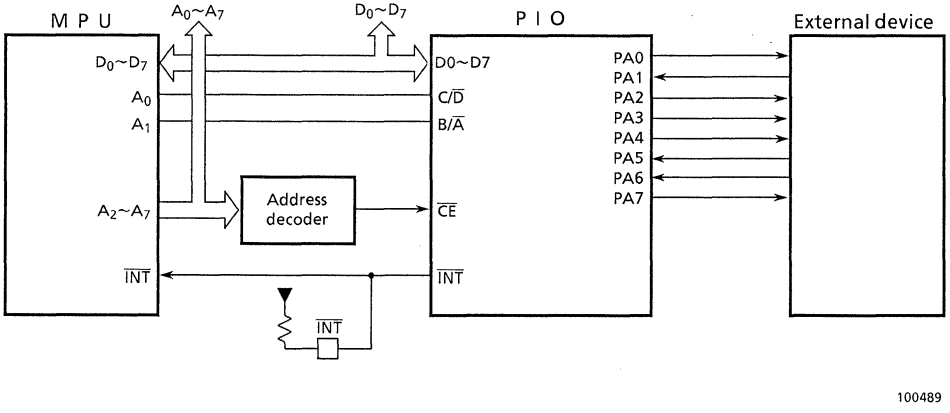


Figure 3.14 PIO Connection

```
        ORG 100H
        LD  SP,100H  ····· Sets the stack pointer.
        LM  2        ····· Sets for MPU mode 2 interrupt.
        LD  A,08H    ····· Writes data in MPU I register.
        LD  I,A
;
        LD  A,02H    ····· Writes the interrupt vector word.
        OUT (01H),A
        LD  A,OCFH   ····· Writes the mode control word.
        OUT (01H),A
        LD  A,62H    ····· Writes the data I/O control word, Sets PIO.
        OUT (01H),A
        LD  A,F7H    ····· Writes interrupt control word.
        OUT(01H),A
        LD  A,9FH    ····· Writes the mask control word.
        OUT (01H),A
;
        EI           ····· Set interrupt enable.
```

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATING

Symbol	Item		Rating	Unit
V_{CC}	Supply Voltage		−0.5~+7	V
V_{IN}	Input Voltage		−0.5~VCC+0.3	V
PD	Power Dissipation (6MHz VERSION : TA = 85°C) (8MHz VERSION : TA = 70°C)		250	mW
TSOLDER	Soldering Temperature (10 sec)		260	°C
TSTG	Storage Temperature		−65~150	°C
TOPR	Operating Temperature	6MHz VERSION	−40~85	°C
		8MHz VERSION	−10~70	

050489

4.2 DC ELECTRICAL CHARACTERISTICS

6MHz VERSION : $T_A = -40°C \sim 85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$
8MHz VERSION : $T_A = -10°C \sim 70°C$, $V_{CC} = 5V \pm 5\%$, $V_{SS} = 0V$

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
VILC	Low Clock Input Voltage			−0.3	—	0.6	V
VIHC	High Clock Input Voltage			V_{CC}−0.6	—	V_{CC}+0.3	V
VIL	Low Input Voltage (Except CLK)			−0.5	—	0.8	V
VIH	High Input Voltage (Except CLK)			2.2	—	V_{CC}	V
VOL	Output Low Voltage	IOL = 2.0mA		—	—	0.4	V
VOH1	Output High Voltage (I)	IOH = −1.6mA		2.4	—	—	V
VOH2	Output High Voltage (II)	IOH = −250μA		V_{CC}−0.8	—	—	V
ILI	Input Leakage Current	$V_{SS} \leqq V_{IN} \leqq V_{CC}$		—	—	±10	μA
ILO	3-STATE Output Leakage Current in Float	$V_{SS}+0.4 \leqq V_{OUT} \leqq V_{CC}$		—	—	±10	μA
IOHD (2)	Darlington Drive Current	VOH = 1.5V REXT = 1.1KΩ		−1.5	—	−5.0	mA
ICC1	Power Supply Current	V_{CC} = 5V f_{CLK} = (1) $V_{ILC} = V_{IL}$ = 0.2V $V_{IHC} = V_{IH} = V_{CC}-0.2V$	AP-6/ AM-6/ AT-6	—	3	8	mA
			AP-8	—	4	10	
ICC2	Standly Supply Current	$V_{ILC} = V_{IL} = 0.2$ $V_{IHC} = V_{IH} = V_{CC}-0.2V$		—	0.5	10	μA

Note : (1) fCLK = 1/tcC (MIN.) (2) Port B only

050489

4.3 AC ELECTRICAL CHARACTERISTICS

6MHz VERSION : $T_A = -40°C \sim 85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$
8MHz VERSION : $T_A = -10°C \sim 70°C$, $V_{CC} = 5V \pm 5\%$, $V_{SS} = 0V$

(1/2)

NO.	SYMBOL	PARAMETER	AP-6/AM-6/AT-6 (6MHz)		AP-8 (8MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	
1	T_{CC}	Clock cycle time	165	DC	125	DC	ns
2	TwCh	High clock pulse width	65	DC	50	DC	ns
3	TwCl	Low clock pulse width	65	DC	50	DC	ns
4	TfC	Clock falling time	—	20	—	15	ns
5	TrC	Clock rising time	—	20	—	15	ns
6	TsCS (RI)	$\overline{CE}$, B/$\overline{A}$ and C/$\overline{D}$ set-up time for $\overline{RD}$, $\overline{IORQ}$	50	—	50	—	ns
*7	Th	Hold time	40	—	40	—	ns
8	TsRI (C)	$\overline{RD}$, $\overline{IORQ}$ set-up time for clock rise	70	—	60	—	ns
9	TdRI (DO)	Delay from $\overline{RD}$, $\overline{IORQ}$ fall to data output	—	300	—	200	ns
10	TdRI (DOs)	Delay from $\overline{RD}$, $\overline{IORQ}$ rise to data float	—	70	—	70	ns
11	TsDI (C)	Data set-up time for clock rise	40	—	40	—	ns
12	TdIO (DOI)	Delay from $\overline{IORQ}$ fall of INTA cycle to data output	—	120	—	85	ns
13	TsM1 (Cr)	$\overline{M1}$ = L set-up time for clock rise	70	—	50	—	ns
14	TsM1 (Cf)	$\overline{M1}$ = H set-up time for clock fall ($\overline{M1}$ sycle)	0	—	0	—	ns
15	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall	—	100 (1)	—	100 (1)	ns
16	TsIEI (IO)	IEI set-up time for $\overline{IORQ}$ fall (INTA cycle)	100	—	80	—	ns
17	TdIEI (IEOf)	Delay from IEI fall to IEO rise	—	120	—	100	ns
18	TdIEI (IEOr)	Delay from IEI rise to IEO fall	—	150	—	120	ns
19	TcIO (C)	$\overline{IORQ}$ = H set-up time for clock fall (in case of making READY to active by next cycle)	170	—	120	—	ns
20	TdC (RDYr)	Delay from clock fall to READY rise	—	170	—	150	ns
21	TdC (RDYf)	Delay from clock fall to READY fall	—	120	—	110	ns
22	TwSTB (C)	STROBE pulse width	120 (2)	—	100 (2)	—	ns
23	TsSTB (C)	Set-up time of $\overline{STROBE}$ rise for clock fall (in case of making READY to active by next cycle)	150	—	100	—	ns

200489

(2/2)

NO.	SYMBOL	PARAMETER	AP-6/AM-6/AT-6 (6MHz)		AP-8 (8MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	
24	TdIO (PD)	Delay from $\overline{IORQ}$ rise to port data stable (Mode 0)	—	160	—	140	ns
25	TsPD (STB)	Data set-up time for $\overline{STROBE}$ rise (Mode 1)	190	—	150	—	ns
26	TdSTB (PD)	Output data delay time from $\overline{STROBE}$ fall (Mode 2)	—	180	—	150	ns
27	TdSTB (PDr)	Delay from $\overline{STROBE}$ rise to data float (Mode 2)	—	160	—	120	ns
28	TdPD (INT)	Delay from port data match to $\overline{INT}$ fall (Mode 3)	—	430	—	350	ns
29	TdSTB (INT)	Delay from $\overline{STROBE}$ rise to $\overline{INT}$ fall (Mode 2)	—	350	—	250	ns

200489

Note :

1 Item with * mark (No.7) is not compatible with NMOS Z80 PIO.

2 (1) If the daisy chain is at N stage,
2.5 TcC > (N-2)TdIEI(IEOf) + TdM1(IEO) + TsIEI(IO) + TTL buffer delay must be satisfied.

(2) In Mode 2, TwSTB > TsPD(STB) must be satisfied.

(3) At test condition : Input : VIH = 2.4V, VIHC = VCC-0.6V, VIL = 0.4V, VILC = 0.6V
Output : VOH = 2.2V, VOL = 0.8V, CL = 100pF

4.4 CAPACITANCE

TA = 25°C

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	Clock Input Capacitance	f = 1MHz All terminals except that to be measured should be earthed.	—	—	10	pF
CIN	Input Capacitance		—	—	5	pF
COUT	Output Capacitance		—	—	10	pF

050489

4.5 TIMING DIAGRAM

Numbers shown in the following figures correspond with those in the 4.3 A.C Electrical Characteristics Table.

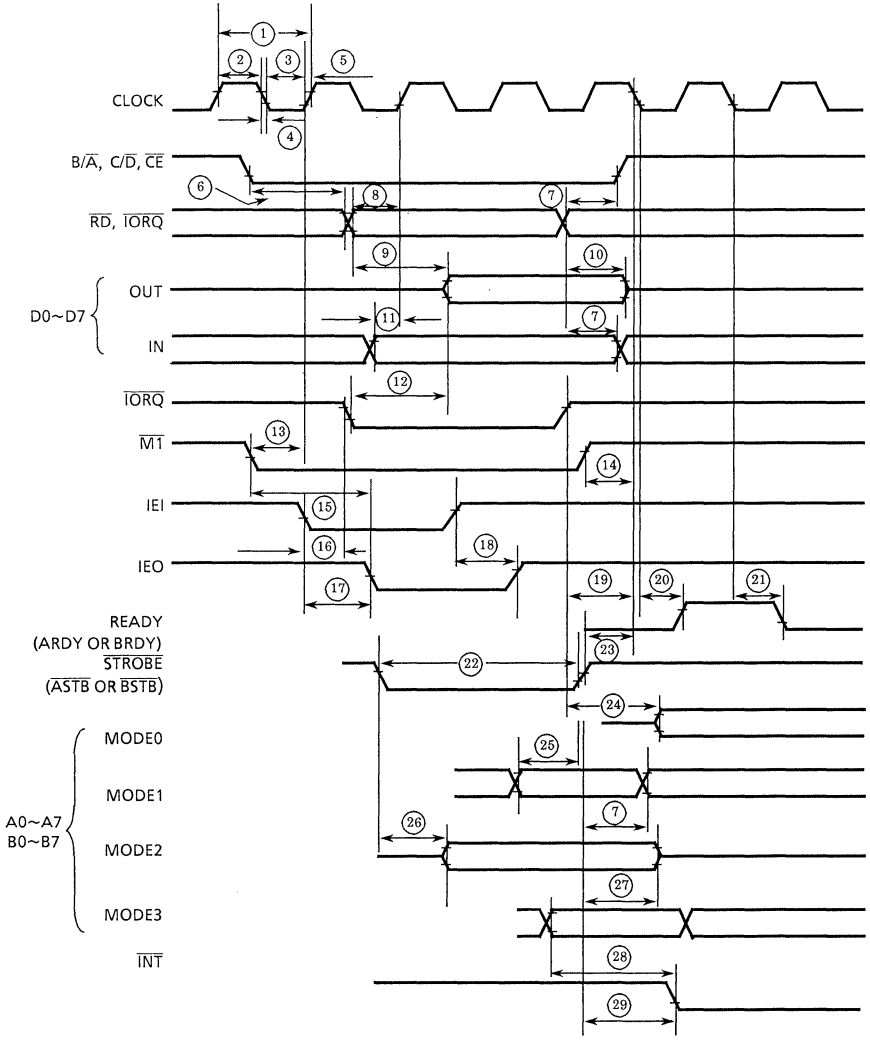


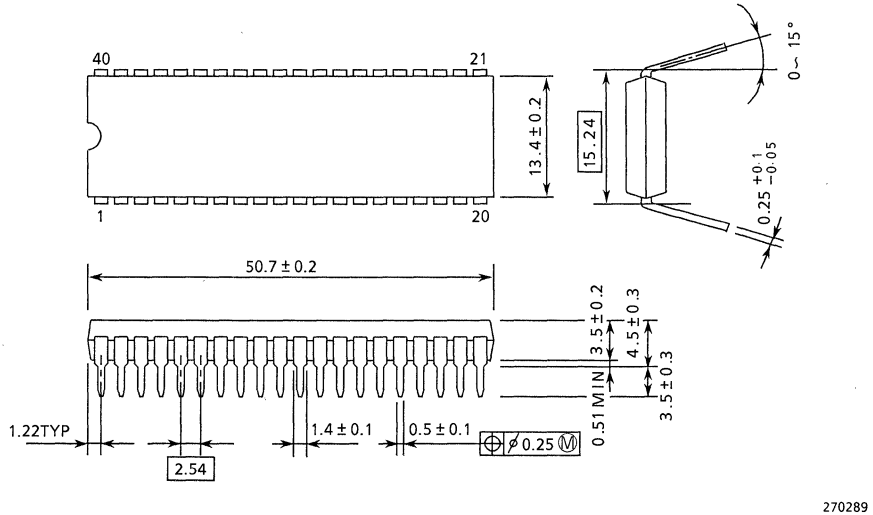
Figure 4.1 Timing Diagram

5. PACKAGE DIMENSION

5.1 40-PIN PLASTIC PACKAGE

DIP40-P-600

Unit : mm



270289

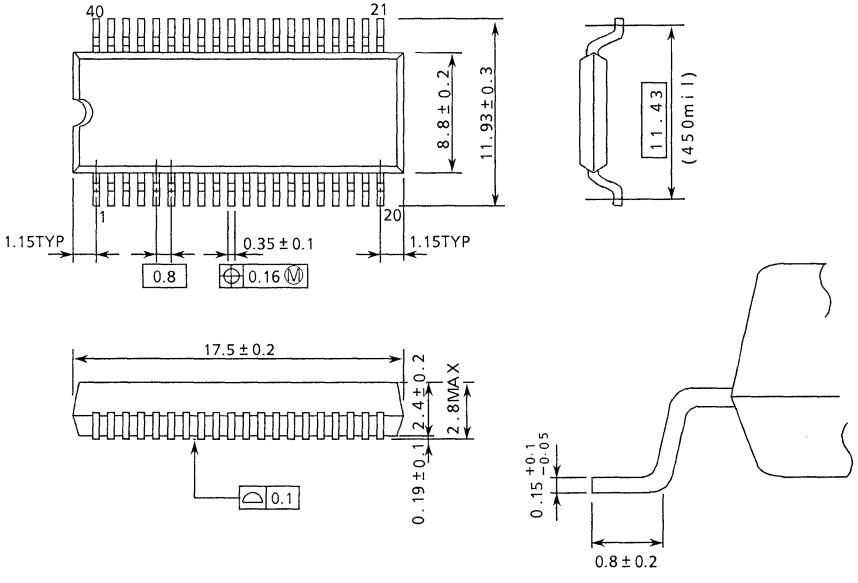
Note 1 : This dimention is measured at the center of bending point of leads.

Note 2 : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.40 leads.

5.2 40-PIN SMALL OUTLINE PACKAGE

SSOP40-P-450

Unit : mm



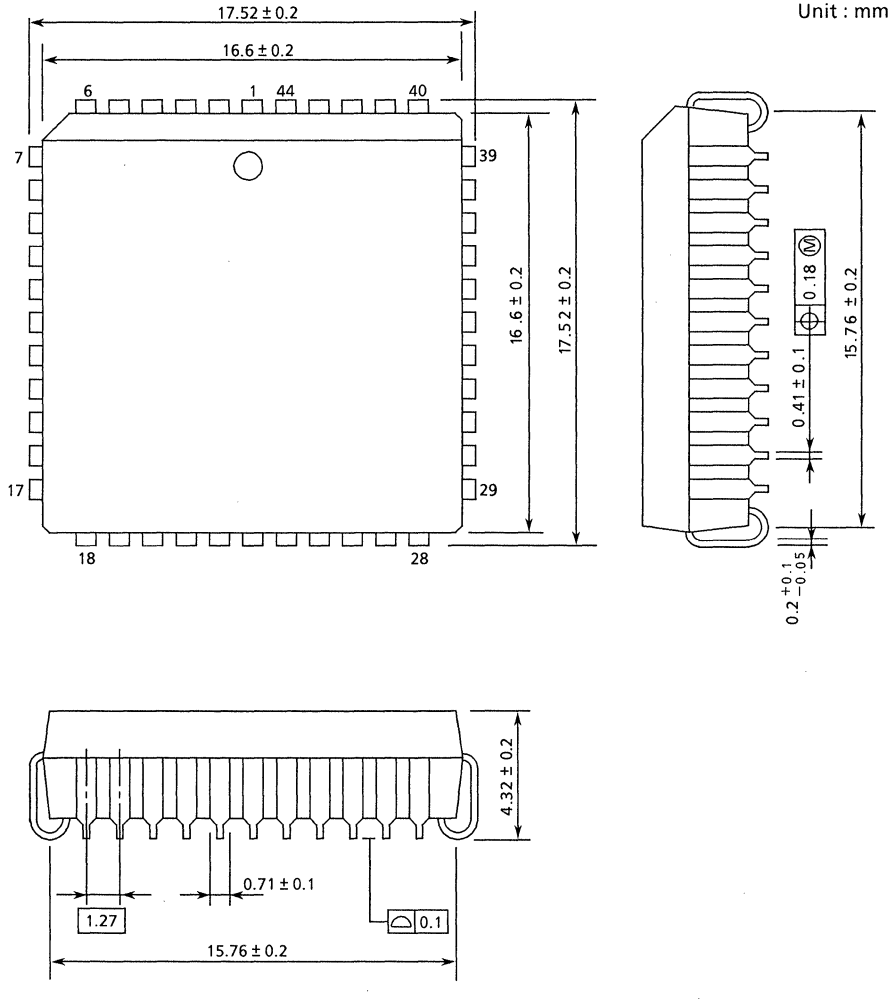
270289

Note : Package Width and length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

5.3 44-PIN PLCC PACKAGE

QFJ44-P-S650

Unit : mm



TMPZ84C30AP-6 / TMPZ84C30AM-6 / TMPZ84C30AT-6
TMPZ84C30AP-8
TLCS-Z80 CTC : COUNTER TIMER CIRCUIT

1. GENERAL DESCRIPTION AND FEATURES

The TMPZ84C30A (hereinafter referred to as CTC) is CMOS version of Z80 CTC and has been designed to provide low power operation.

The TMPZ84C30A is fablicated using Toshiba's CMOS Silicon Gate Technology.

The principal functions and features of the CTC are as follows.

(1) Compatible with the Zilog Z80 CTC.

(2) Low power consumption.
4mA Typ. (@5V @6MHz) ··· TMPZ84C30AP-6/AM-6/AT-6
5mA Typ. (@5V @8MHz) ··· TMPZ84C30AP-8
10μA Max. (@5V, Stand-by)

(3) DC to 6MHz operation ··· TMPZ84C30AP-6/AM-6/AT-6
DC to 8MHz operation ··· TMPZ84C30AP-8

(4) Single 5V power supply and single-phase clock. 5V ±10% (6MHz VERSION)
5V ± 5% (8MHz VERSION)

(5) Capable of driving Darlington transistors.

(6) Four independent counter/timer channels each of which is capable of independently selecting Timer Mode and Counter Mode.

(7) Each channel is provided with a prescaler to divide system clock into 16 or 256.

(8) Built-in interrupt control logical operation circuit allows priority processing of interrupt in Daisy-chain structure and automatic loading of 8 bit interrupt vector on the system bus.

(9) Four channels occupy 4 successive positions in the Daisy-chain structure. Most significant channel is Channel 0 and least significant channel is Channel 3.

(10) In both modes, at the zero count, the content of the time constant register is automatically loaded on the down counter.

(11) In either Counter Mode or Timer Mode, the content of the down counter is readable by the microprocessor (hereinafter referred to as MPU).

(12) Interrupt function available in Z80 MPU Mode 2.

(13) In Timer Mode, the timer operation is selectable at the rise or fall of the starting trigger. In addition, in Counter Mode the decrement (-1) of the content of the down counter either at the rise or fall of external clock is selectable.

(14) Programming to generate interrupt by zero count by the down counter in each channel is possible.

(15) 40 pin DIP package, 40 pin SOP, 44 pin PLCC package.

Note : Z80 is a trademark of Zilog Inc., U.S.A.

2. PIN CONNECTIONS AND PIN FUNCTIONS

2.1 PIN CONNECTIONS (Top View)

The pin connections of the CTC are as shown in Figure 2.1 and Figure 2.2.

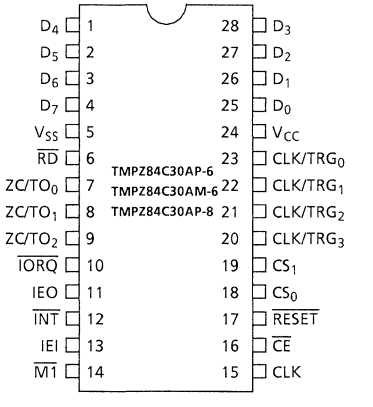


Figure 2.1 DIP, SOP Pin Connections

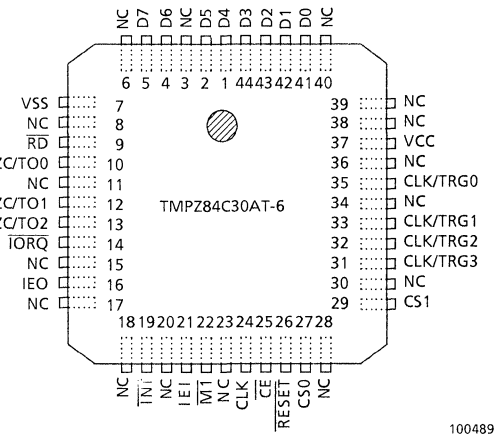


Figure 2.2 44-Pin PLCC Package

2.2 PIN NAMES AND FUNCTIONS

I/O pin names and functions are as shown in Table 2.1.

Table 2.1 Pin Names and Functions (1/2)

Pin Name	Number of Pin	Input/Output 3-state	Function
D0~D7	8	I/O 3-state	8-bit bi-directional data bus.
$\overline{RD}$	1	Input	The read signal. This signal is used in combination with $\overline{IORQ}$ and $\overline{CE}$ signals for transfer of data and channel control words between MPU and CTC.
ZC/TO0 ~ ZC/TO2	1	Output	The Zero Count/Timer Out signal. In either the Timer mode, or counter mode, pulses are output from these pins when the down counter has reached zero.
$\overline{IORQ}$	1	Input	I/O request signal. This signal is used in combine with $\overline{RD}$ and $\overline{CE}$ signals for transfer of data and channel control words between MPU and CTC.
IEO	1	Output	Interruption enable output. Controls interruptions by subordinate peripheral LSI's in the Daisy-chain structure. This terminal becomes H level only when the IEI terminal is at H level and MPU is not providing the interruption service to channels in CTC.

100489

Table 2.1 Pin Names and Functions (2/2)

Pin Name	Number of Pin	Input/output 3-state	Function
$\overline{INT}$	1	Output	Interruption request. This terminal becomes L level if a down-counter for any channel in CTC counts zero when the IEI terminal is at H level and interruption is authorized by a program.
IEI	1	Input	The interrupt Enable Input signal. This signal is used with the IEO to form a priority daisy chain when there is more than one interrupt-driven peripheral LSI.
$\overline{M1}$	1	Input	Machine cycle 1. Informs the machine cycle from MPU. In combination with the $\overline{RD}$ signal, indiates that MPU fetches commands from the memory, and in combination with the $\overline{IORQ}$ signal, indicates that MPU is in the interruption acknowledge cycle. This terminal is used, in combination with the $\overline{IORQ}$ signal, to send the interruption vector to MPU.
CLK	1	Input	Single-phase clock input. Single-phase Z80 standard system clock is inputted to this terminal. When this CLK terminal is in the DC state (high or low level), the CTC is placed in the stationary state.
$\overline{CE}$	1	Input	Chip enable. Used to write MPU-CTC channel control word, interruption vector, and time constant or to read the content of a downcounter for each channel in combination with the $\overline{IORQ}$ and $\overline{RD}$ terminals.
$\overline{RESET}$	1	Input	Reset signal . When the reset signal is inputted to this terminal, all channels stop to operate and interruption enable bits in all channel control registers are reset. This $\overline{RESET}$ terminal must be kept at "L" level for at least 3 system clocks.
CS0~CS1	2	Input	Channel selection. Any one of four channels of the CTC is selected by 2-bit code at time of read/write.
CLK / TRG0~ CLK / TRG3	4	Input	The external clock/timer trigger. These 4 CLK/TRG pins correspond to 4 channels. In the counter mode, the down counter is decremented by 1 and in the timer mode, the timer is activated at each active edge (a rising of falling edge) of the signal which are input by these pins. It can be selected by program whether the active edge is a rising or falling edge.
Vcc	1		The power supply (+ 5V) pin
Vss	1		The ground (0V) pin

100489

3. FUNCTIONAL DESCRIPTION

3.1 BLOCK DIAGRAM

The block diagram of CTC is shown in Figure 3.1.

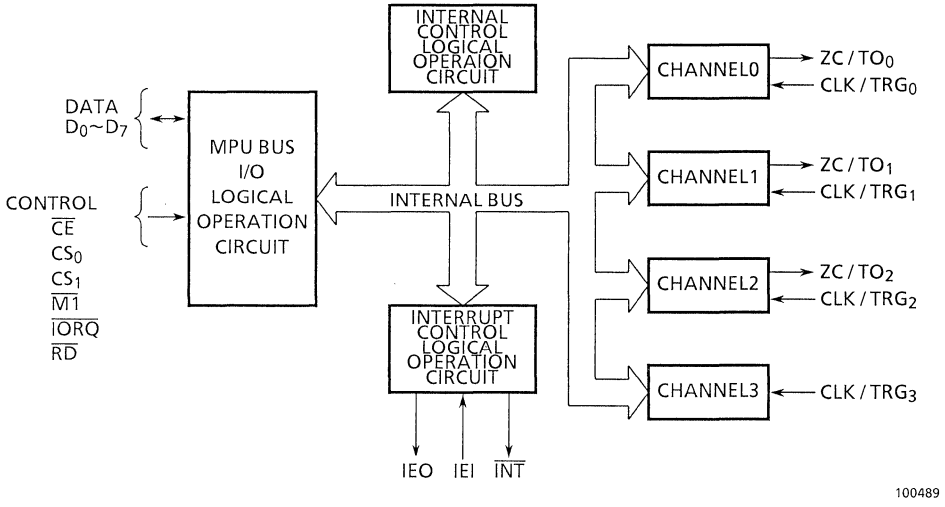


Figure 3.1 Block Diagram

3.2 SYSTEM CONFIGURATION

The CTC system consists of the following 4 logic circuits.

(1) MPU bus Input/Output Logic Circuit

(2) Internal control logic circuit

(3) Interrupt Control Logic Circuit

(4) 4 Independent Counter/Timer Channel Logic Circuit

3.2.1 MPU bus input/output logic circuit

This circuit transfers data between the MPU and the CTC.

3.2.2 Internal control logic circuit

This circuit controls the CTC operational functions such as the CTC selecting enable, reset, and read/write circuit.

3.2.3 Interrupt control logic circuit

This circuit performs the MPU interrupt related processing such as priority determination. The order of priority with other LSIs determined according to the physical location in daisy chain connection.

3.2.4 Counter/timer channel logic circuit

This circuit consists of the following 2 registers and 2 counters.
Figure 3.2 shows the configuration of this circuit.

- Time-constant register (8 bits)
- Channel control register (8 bits)
- Down-counter (8 bits)
- Prescaler (8 bits)

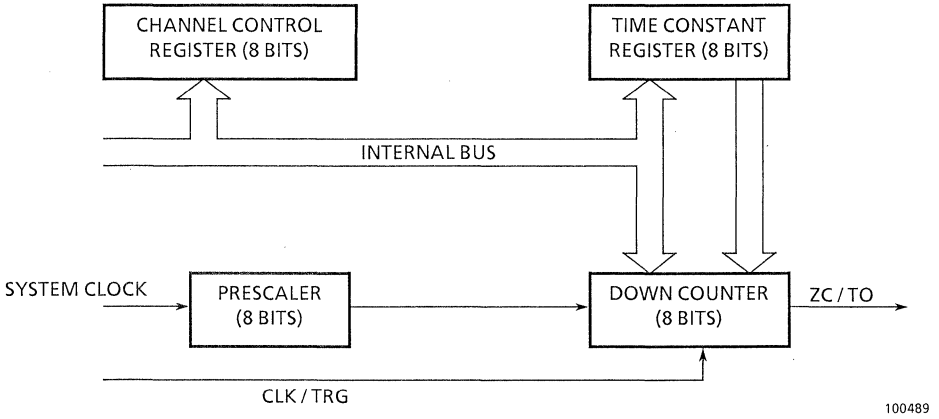


Figure 3.2 Configuration Counter/Timer Channel Logic Circuit

(1) Time-constant register

This register holds the time constant to be written in the down counter. When the CTC is initialized or the down-counter has reached zero, the time constant is loaded into down-counter. The time constant is set immediately after the MPU has written the channel control word in the channel control register. For a time constant, an integer from 1 to 256 can be used.

(2) Channel control register

This register is used to choose the channel mode or condition according to the channel control word sent from the MPU.

(3) Down-counter

The contents of the time-constant register are loaded into the down counter. In the counter mode, these contents are decremented for each edge of the external clock. In the timer mode, they are decremented for each prescaler clock output. The contents of the time-constant register are loaded at intialization or when the down-counter has reached zero.

The contents of the down-counter can be read any time. Also, the system can be programmed so that an interrupt request is generated each time the down-counter has reached zero.

(4) Prescaler

The prescaler, used only in the timer mode, divides the system clock by 16 or 256. The dividing number is programmed by channel control word. The output of the prescaler becomes the clock input to the down-counter.

3.3 CTC BASIC OPERATIONS

3.3.1 Reset

The state of the CTC is unstable after it is powered on. To stabilize the CTC, the low level signal needs to be applied to the $\overline{\text{RESET}}$ pin. On any channel, the channel control word and time-constant data must be written to be started before it is started in the counter or timer mode. To program the system to enable interrupts, the interrupt vector word must be written in the interrupt controller. When these data have been written in the CTC, it is ready to start.

3.3.2 Interrupt

The CTC can cause an interrupt when the MPU is operating on the mode 2. The CTC interrupt can be programmed for each channel. Each time the channel's down-counter has reached zero, the CTC outputs the interrupt request signal ($\overline{\text{INT}}$). When the MPU accepts the CTC's interrupt request, the CTC outputs the interrupt vector. Based on this interrupt vector, the MPU specifies the start address of the interrupt processing routine and calls it to start interrupt processing.

The MPU specifies the start address of the interrupt processing routine by the interrupt vector output from the CTC, so that the user can change the vector value to call any desired address.

The interrupt processing is terminated when the MPU executes an RETI instruction. The CTC has the circuit which decodes the RETI instruction. By constantly monitoring the data bus the CTC can detect the termination of the interrupt procrssing.

The order of interrupt priority with the Z80 peripheral LSIs is determined by the daisy chain connection. That is, the peripheral LSIs are connected one after another and the one physically near MPU is given a higher priority.

Inside the CTC, channel 0 is given the highest priority, followed by channels 1, 2 and 3 in this order.

The CTC and other peripheral LSIs have the signal lines IEO and IEI. Connect the IEO of a higher peripheral LSI to the IEI of a lower peripheral LSI. Connect the IEI of the highest peripheral LSI to VCC. Leave the IEO of the peripheral LSI unused. In this connection, the CTC interrupt is caused under the following conditions:

- When both IEI and IEO are high, no interrupt is caused. At this time, the $\overline{INT}$ signal is high. An interrupt can be requested in this state.
- When the CTC outputs the interrupt request signal ($\overline{INT}$), the IEO of the CTC becomes low. When the MPU accepts the interrupt, the $\overline{INT}$ goes high again.
- When the IEI goes low, the IEO also goes low.
- While the IEI is low, no interrupt can be requested.
- When the IEI goes low while an interrupt is being serviced, the interrupt processiong is aborted.

3.3.3 Operation modes

The CTC oprates in either the counter mode or the timer mode. Mode is selected by writing the channel control word.

(1) Counter mode

In the counter mode, the number of edge of the pulse applied to the channel's CLK/TRG pin is counted. When pulses have been input, the contents of the down-counter are decremented synchronizing with the rising edge of the next system clock. The pulse's rising edge or falling esge to be counted can be specified by the channel control word.

When the contents of the down-counter has reached zero, the high level pulse is output from the ZC/TO pin. When the interrupt is enabled by the channel control word, the $\overline{INT}$ pin goes low and an interrupt is requested. When the contents of the down-counter has reached zero, the time constant data written in the time constant register is automatically loaded into the down-counter. To load a new time constant value into the down-counter, write the data to the time constant register, and it is loaded into the down-counter after the current count operation is terminated.

(2) Timer mode

In the timer mode, the time intervals which are integral multiples of the system clock period. A timer interval is measured according to the system clock. The system clock is supplied to the prescaler which divides it by a factor of 16 or 256. The output of the prescaler provides the clock to decrement the down-counter by 1. The time constant data

is automatically loaded into the down-counter each time it has reached zero as in the counter mode. When the contents of the down-counter has reached zero, the high level pulse is output from the ZC/TO pin.

This pulse period is given by the following expression:

tc * P * TC

Where, tc = System clock period
P = Prescaler value (16 or 256)
TC = Time constant data (256 for 00H)

The user can select, by means of the channel control word, to start the timer automatically or to start the timer on the edge of the pulse at CLK/TRG pin. In case the user select the CLK/TRG pin, the user can also select the rising edge or falling edge of the pulse.

3.4 CTC STATUS TRANSITION DIAGRAM AND BASIC TIMING

3.4.1 Transition diagram

Figure 3.3 shows the CTC status transition diagram.

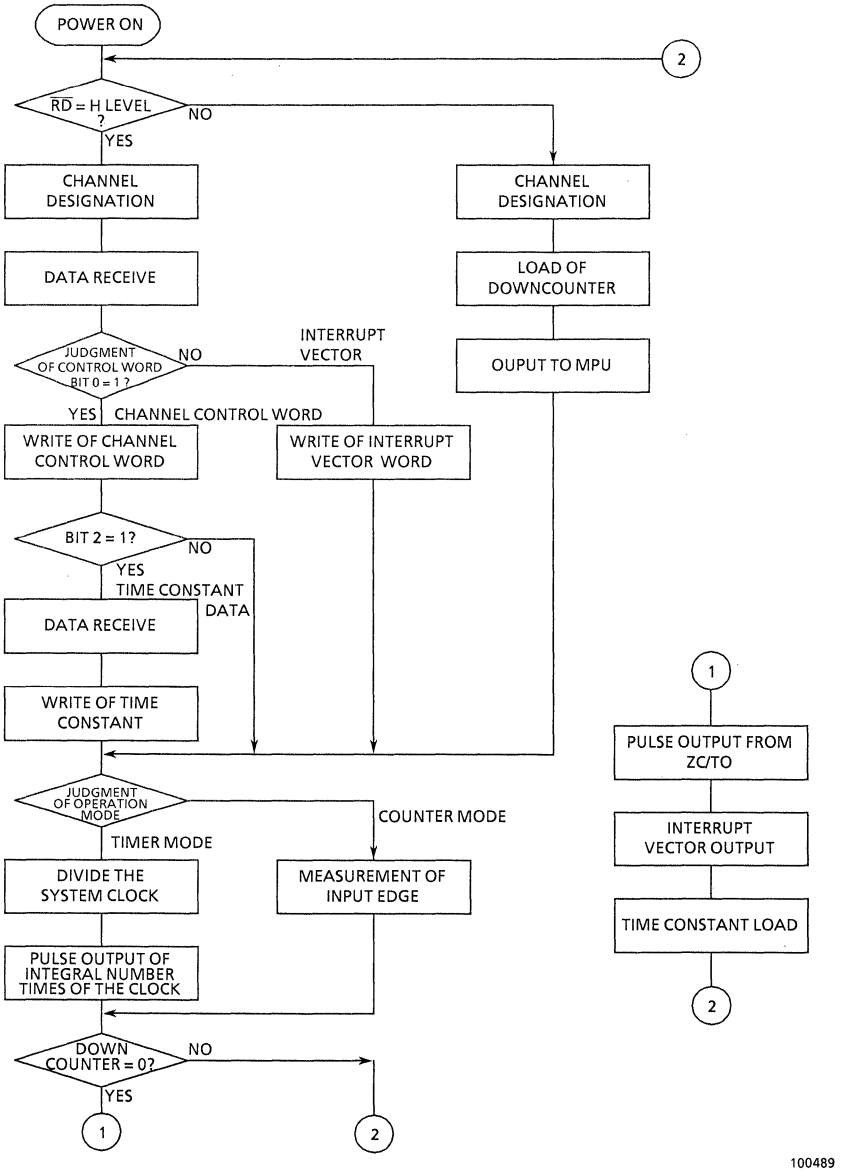


Figure 3.3(a) CTC Transition Diagram (a)

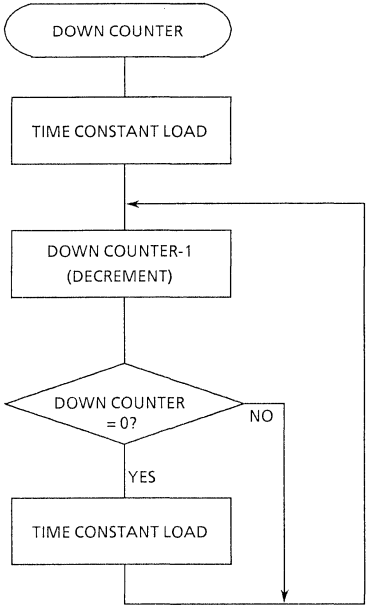


Figure 3.3(b) CTC Transition Diagram (b)

3.4.2 Write cycle

The write cycle is used to write a channel control word, an interrupt vector, or a time constant. The MPU drives the $\overline{IORQ}$ pin of the CTC low in the subsequent system clock cycle T2 to start the write cycle.

It is required to make the $\overline{M1}$ pin of the CTC high to indicate that the write cycle is on.

At the state of the cycle, the channel is specified by CS1 or CS0 of the CTC. Thus, the CTC's internal registers are ready to accept data in system clock T3. Tw is the state to be automatically inserted by the MPU.

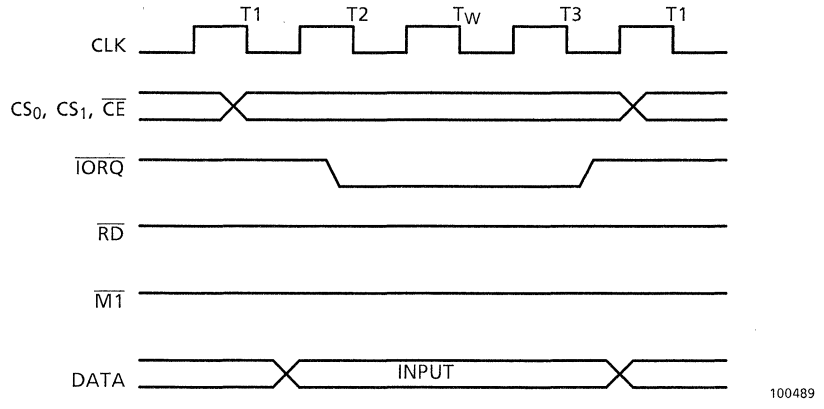


Figure 3.4 Write Timing

3.4.3 Read cycle

The read cycle is used to read the contents of the down-counter. During clock cycle T2, the MPU initiates a read cycle by driving the $\overline{RD}$ and $\overline{IORQ}$ pins low. It is required to make the $\overline{M1}$ pin high to indicate that the read cycle is on. At the start of the read cycle, the channel is specified by CS1 or CS0 of the CTC.

At the rising edge of system clock TW, the contents of the down-counter at the time of the rising edge of T2 are put on the data bus. TW is the wait state to be automatically inserted by the MPU.

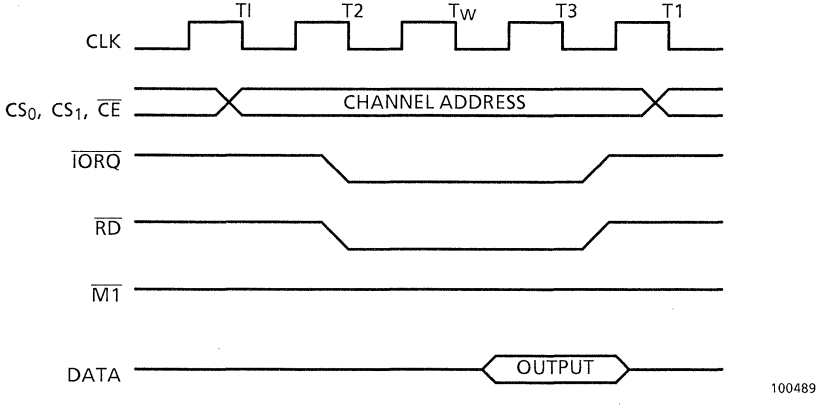


Figure 3.5 Read Timing

3.4.4 Counter mode

In the counter mode, the down-counter is decremented synchronizing with the system clock, at the edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the down-counter is decremented one system clock later. When the down-counter has reached zero, a high level pulse is output from the ZC/TO pin.

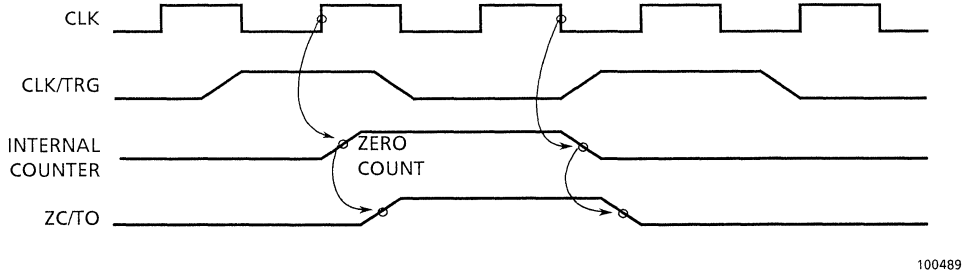


Figure 3.6 Couner Mode Timing

3.4.5 Timer mode

The timer starts operating at the second rising edge of the system clock from the rising edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be apllied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the timer starts one system clock cycle later.

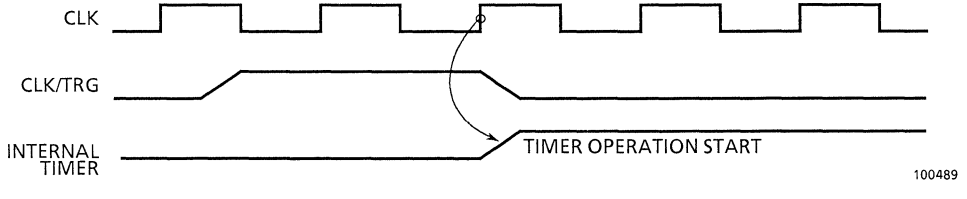


Figure 3.7 Timer Mode Timing

3.4.6 Interrupt acknowledge cycle

Having received the interrupt request signal ($\overline{INT}$) from the CTC, the MPU drives the CTC's $\overline{M1}$ pin and $\overline{IORQ}$ pin low to provide the acknowledge signal. The $\overline{IORQ}$ pin goes low 2.5 system clocks later than the $\overline{M1}$ pin. To stabilize the signal lines (IEI and IEO) in daisy chain connection, the interrupt request cannot be changed on each channel while the $\overline{M1}$ pin is low. The $\overline{RD}$ pin is held high to make distinction between the instruction fetch cycle and the interrupt acknowledge cycle. While the $\overline{RD}$ pin is high, the CTC's interrupt control circuit determines the interrupt-requesting channel of highest priority. When the CTC's IEI is high and the $\overline{M1}$ pin and $\overline{IORQ}$ pin go low, the interrupt vector is output from the interrupt requesting channel of highest priority. At this time, 2 system clock cycles are automatically inserted by the MPU as a wait state maintain the stabilization of the daisy chain connection.

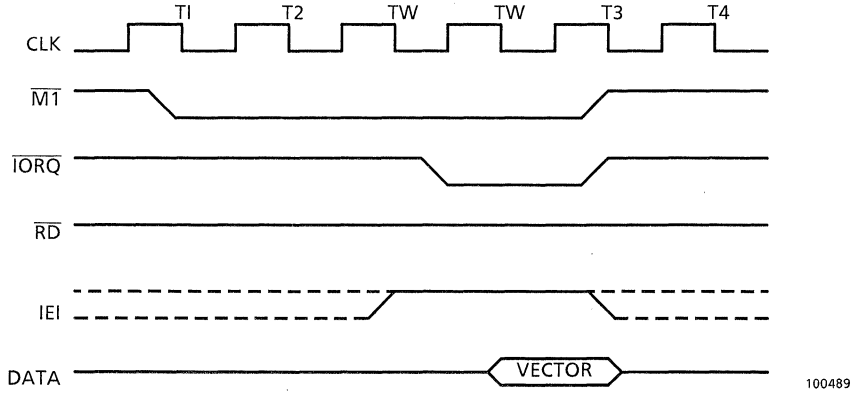


Figure 3.8 Interrupt Acknowledge Timing

3.4.7 Return from interrupt processing

Return from the interrupt processing is performed when the MPU executes the RETI instruction. This RETI instruction must be used at the end of the interrupt processing routine. When this instruction is executed by the MPU, the CTC's IEI and IEO return to the active state brfore the interrupt has been serviced.

The RETI instruction is a 2-byte instruction. Its code is EDH 4DH. The CTC decodes this instruction to check if there is the next interrupt request channel.

In the daisy chain structure, the interrupting LSI's IEI and IEO are held high and low respectively at the time the instruction code EDH has been decoded.

The code following EDH is 4DH, only the peripheral LSI which has sent the last interrupt vector (that is, the LSI whose IEI is high and IEO is low) returns from the interrupt processing. This restarts the processing of the suspended interrupt of the peripheral LSI of the next higher priority.

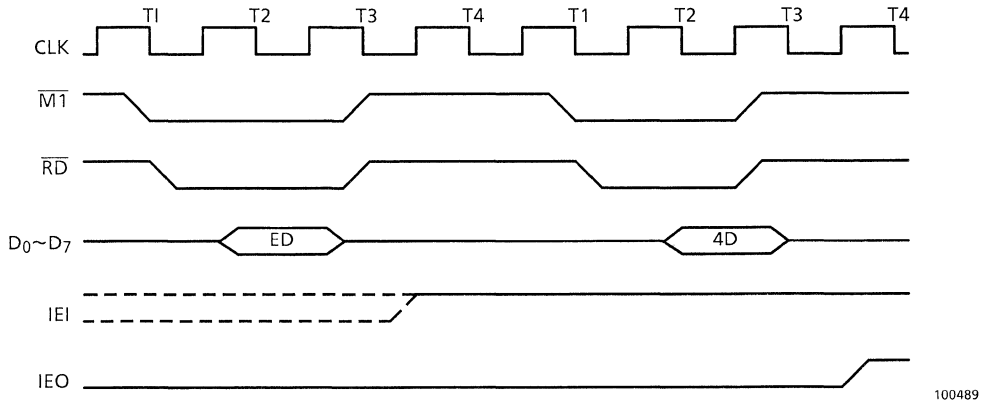


Figure 3.9 Interrupt Return Timing

3.5 CTC OPERATIONAL PROCEDURE

To operate the CTC in the counter mode or the timer mode, the channel control word and time-constant data must be written in the CTC. To enable interrupt by the channel control word, the interrupt vector must be written in the CTC.

(1) Channel control word

To write the channel control word in the CTC, the channel must be specified by the corresponding channel codes. Table 3.1 shows the channel codes.

Table 3.1 Channel codes

Channel	Input Terminal	
	CS1	CS0
0	0	0
1	0	1
2	1	0
3	1	1

270289

The channel control word to be written in the CTC consists of 8 bits. The system data bus D0 through D7 correspond to bit 0 through 7 respectively. Figure 3.10 shows the meaning of each bit. Table 3.2 shows the function of each bit.

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt	Counter/ timer	Prescaler	Edge	Trigger	Time constant	Stop	1

100489

Figure 3.10 Channel Control Word

For the channel control words, D0 must always be 1.

Table 3.2 Meanings and Function of Channel Control Words (1/2)

Bit	Meaning and function	
	0	1
Bit 7 (D7)	Disable channel interrupt	Enables channel interrupt. In either counter or timer mode, the interupt is requested every time the down-counter has reached zero. When this bit is set to "1", the interupt vector must be written in the CTC before the down-counter starts. When the channel control word whose D7 bit is "1" is written in an already operating channel, the interrupt occurs only when the down-counter has reached zero for the first time after the writing of the new channel control word.
Bit 6 (D6)	Puts the channel in the timer mode. Puts the system clock into the prescaler and outputs the divided signal to the down-counter.	Puts the channel in the counter mode. The down-counter is decremented for each edge trigger applied to the CLK/TRG pin. In the counter mode, the prescaler is not used.
Bit 5 (D5)	Used only in the timer mode. The prescaler is set to divide the system clock by 16.	Used only in the timer mode. The prescaler is set to divide the system clock by 256.
Bit 4 (D4)	In the timer mode, the timer operation starts on the falling edge of the trigger PULSE (CLK/TRG). In the counter mode, the down-counter is decremented on the falling edge of the external clock pulse (CLK/TRG).	In the timer mode, the timer operation sarts on the rising edge of the trigger pulse (CLK/TRG). In the counter mode, the down-counter is decremented on the rising edge of the trigger pulse (CLK/TRG).
Bit 3 (D3)	Used only in the timer mode. The timer operation sarts on the rising edge of the trigger pulse clock after a time constant is loaded onto the down-counter.	Used only in the timer mode. The timer operation is started at the leading edge of the external triger pulse that inputs 2 system clocks after a time constant is loaded onto the down-counter. When a time lag between the system clock and trigger pulse satisfies a setup time, the prescaler starts to operate from the second leading edge of the trigger pulse. If a time lag between the system clock and trigger pulse does not satisfy the setup time, the prescaler starts to operate at the leading edge of the trigger pulse after 3 system clocks. If the trigger pulse is input before loading of a time constant, the operation is the same as that when Bit 3 = 0.

100489

Figure 3.2 Meanings and Function of Channel Control Words (2/2)

Bit	Meaning and function	
	0	1
Bit 2 (D2)	This bit (0) indicates that there is no time constant written after channel control word. However, when the channel is in the reset state and this bit cannot be changed to "0" in the channel control word which is given first after the channel reset. To change other state without changing a time constant, input a channel control word with this bit changed to 0.	This bit (1) indicates that there is a time constant written immediately after a channel control word. If a time constant is written while the downcounter is operating a new time constant is set in the time constant register. The counting which is in progress is carried out continuously when the downcounter becomes zero and a new time constant is loaded onto the downcouner.
Bit 1 (D1)	Continues the current channel operation.	Stops the down-counter operation. When this bit is set to "1", the channel operation stops but all the channel control register bits remain unchanged. When bit 2 = "1" and bit 1 = "1", the channel operation remains stopped until a new time constant is written. The channel is set up after the new time constant is programmed. The channel is restarted according to the state of bit 3. When bit 2 = "0" and bit 1 = "1", the channel operation does not start until a new channel control word is written.

100489

(2) Time-constant data

In either the time mode or the counter mode, the time-constant data must be loaded into the time constant register. When bit 2 (D2) of the channel control word is "1", the time constant is loaded into the time constant register immediately after the channel control word is written. A time constant value must be an integer in a range of 1 to 256. When the 8 bits of a time constant are all "0"s, such a time constant is assumed to be 256. Figure 3.11 shows the bit configuration of time-constant data.

D7	D6	D5	D4	D3	D2	D1	D0
TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0

100489

Figure 3.11 Time-Constant Data

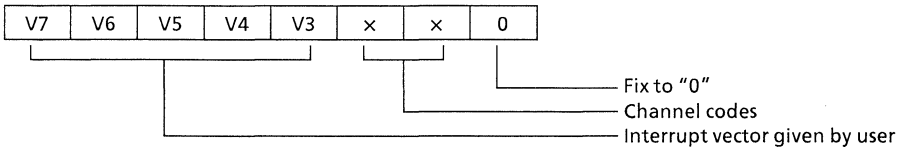
(3) Interrupt Vector

In interrupt in the MPU mode-2, the interrupting channel must give an interrupt vector to the MPU. An interrupt vector is written in the channel-0 interrupt vector register with bit 0 (D0) ="0". The vector is written in the same way as the channel control word is written on channel 0. However, bit 0 (D0) of the vector should always be "0". Bit 7 (D7) thrutgh bit 3 (D3) are user-defined values. Bit 2 (D2) and bit 1 (D1) are automatically given and contain the code of the interrupt-requesting channel having the highest priority. Table 3.3 shows the channel codes. Figure 3.12 shows the interrupt vector bit configuration.

Table 3.3 Channel Codes

Bit 2 (D2)	Bit 1 (D1)	Channel number	
0	0	0	↑ (high)
0	1	1	
1	0	2	Priority
1	1	3	↓ (low)

100489



100489

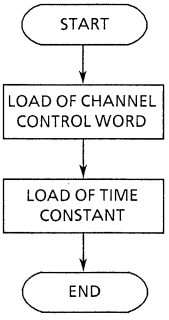
Figure 3.12 Interrupt Vector

3.6 USING CTC

(1) Counter mode

The following describes how to use the CTC by referring to a program using channel 0 with interrupt disabled.

(a) The counter programming procedure is shown in Figure 3.13.



100489

Figure 3.13 Counter Programming Procedure

(b) The block diagram for converting the 100kHz system clock into the 10kHz equivalent is shown in Figure 3.14.

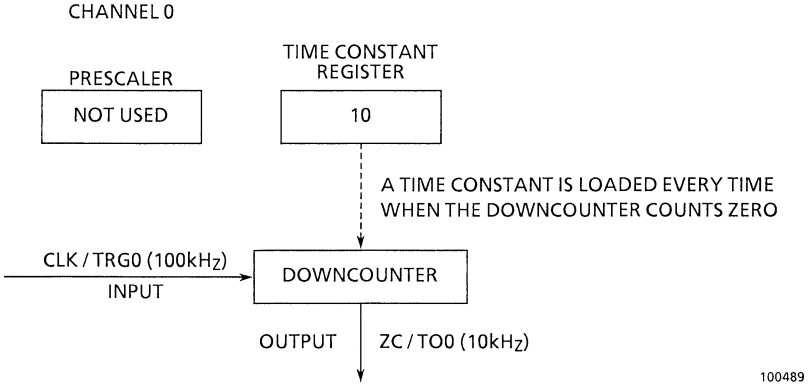


Figure 3.14 Down-Counter Block Diagram

(c) The channel control word configuration is shown in Figure 3.15.

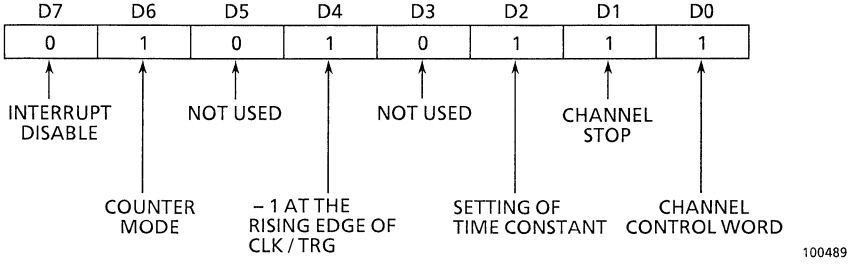


Figure 3.15 Channel Control Word Configuration

(2) Timer mode

(a) The timer programming procedure with interrupt disabled is shown in Figure 3.16

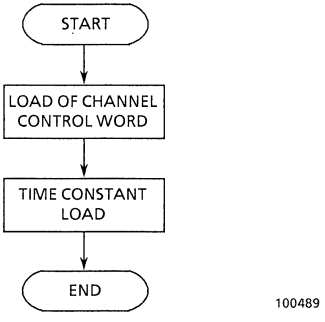


Figure 3.16 Timer Programing Procedure

(b) The block diagram for converting the 4MHz system clock into the 1 kHz equivalent is shown in Figure 3.17.

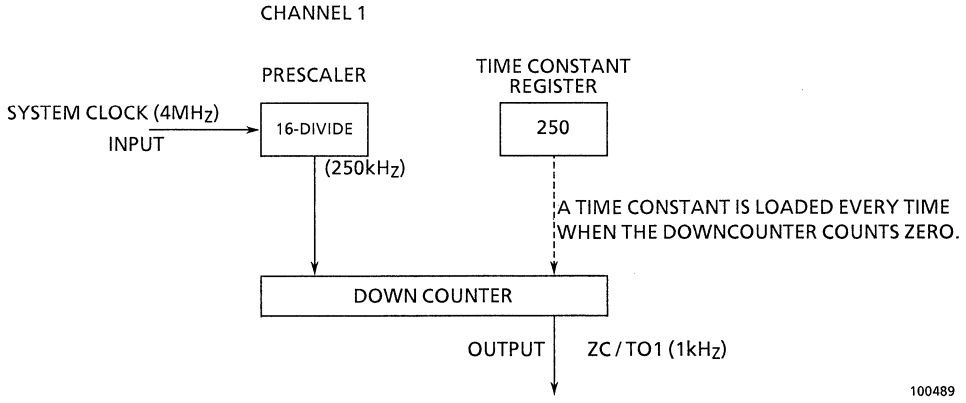


Figure 3.17 Timer Block Diagram

(c) The channel control word is shown in Figure 3.18.

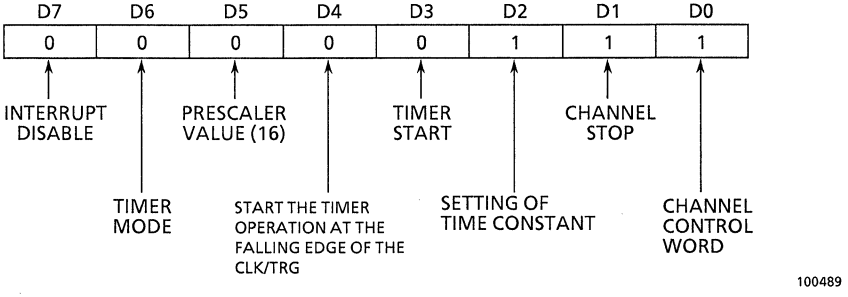


Figure 3.18 Channel Control Word

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item		Rating	Unit
Vcc	Supply Voltage		− 0.5 to + 7	V
VIN	Input Voltage		− 0.5 to Vcc + 0.5	V
PD	Power Dissipation (6MHz VERSION : TA = 85°C) (8MHz VERSION : TA = 70°C)		250	mW
TSOLDER	Soldering Temperature (10 sec)		260	°C
TSTG	Storage Temperature		− 65 to 150	°C
TOPR	Operating Temperature	6MHz VERSION	− 40 to 85	°C
		8MHz VERSION	− 10 to 70	

050489

4.2 DC ELECTRICAL CHARACTERISTICS

6MHz VERSION : $T_A = -40°C \sim 85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$
8MHz VERSION : $T_A = -10°C \sim 70°C$, $V_{CC} = 5V \pm 5\%$, $V_{SS} = 0V$

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
VILC	Low Clock Input Voltage			− 0.3	—	0.6	V
VIHC	High Clock Input Voltage			Vcc − 0.6	—	Vcc + 0.3	V
VIL	Low input Voltage (Except CLK)			− 0.5	—	0.8	V
VIH	High input Voltage (Except CLK)			2.2	—	Vcc	V
VOL	Output Low Voltage	IOL = 2.0mA		—	—	0.4	V
VOH1	Output High Voltage (I)	IOH = − 1.6mA		2.4	—	—	V
VOH2	Output High Voltage (II)	IOH = − 250μA		Vcc − 0.8	—	—	V
ILI	Input Leak Current	Vss ≦ VIN ≦ Vcc		—	—	± 10	μA
ILO	3-state Output Leakage Current in Float	Vss + 0.4 ≦ VOUT ≦ Vcc		—	—	± 10	μA
IOHD*	Darlington Drive Current, (2)	VOH = 1.5V, REXT = 1.1KΩ Applied to ZC / TO0~ZC / TO2		− 1.5	—	− 5.0	mA
Icc1	Power Suuply Current	Vcc = 5V fCLK = (1) VIH = VIHC = Vcc − 0.2V VIL = VILC = 0.2V	AP-6/ AM-6/ AT-6	—	4	7	mA
			AP-8	—	5	10	
Icc2	Standby Supply Current	Vcc = 5V, CLK = VIH = Vcc − 0.2, VIL = VILC = 0.2V		—	0.5	10	μA

Note : (1) $f_{CLK} = 1 / TcC$ (MIN.)
(2) Applied to ZC / TO0, ZC / TO1 and ZC / TO2.

050489

4.3 AC ELECTRICAL CHARACTERISTICS

6MHz VERSION : $T_{OPR} = -40°C \sim 85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$
8MHz VERSION : $T_A = -10°C \sim 70°C$, $V_{CC} = 5V \pm 5\%$, $V_{SS} = 0V$

(1/2)

NO.	SYMBOL	PARAMETER	AP-6/AM-6 /AT-6 (6MHz)		AP-8 (6MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	
1	TcC	Clock cycle time	165	DC	125	DC	ns
2	TwCh	High clock pulse width	65	DC	50	DC	ns
3	TwC1	low clock pulse width	65	DC	50	DC	ns
4	TfC	Clock falling time	—	20	—	15	ns
5	TrC	Clock rising time	—	20	—	15	ns
6	Th	Hold time	0	—	0	—	ns
7	TsCS (C)	Set-up time for clock rise	100	—	100	—	ns
8	TsCE (C)	$\overline{CE}$ Set-up time for clock rise	100	—	80	—	ns
9	TsIO (C)	Set-up time up to $\overline{IORQ}$ fall for clock rise	70	—	65	—	ns
10	TsRD (C)	Set-up time up to $\overline{RD}$ fall for clock rise	70	—	55	—	ns
11	TdC (DO)	Delay from clock rise to data output	—	130	—	110	ns
12	TdC (DOZ)	Delay from $\overline{IORQ}$, $\overline{RD}$ rise to data float	—	90	—	85	ns
13	TsDI (C)	Data input set-up time for clock rise	40	—	40	—	ns
14	TsM1 (C)	$\overline{M1}$ set-up time for clock rise	70	—	55	—	ns
15	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall (in case of generating only interruption immediately before M1 cycle)	—	130	—	110	ns
16	TdI0 (DOI)	Delay from $\overline{IORQ}$ fall to data output (INTA cycle)	—	110	—	85	ns
17	TdIEI (IEOf)	Delay from IEI fall to IEO fall	—	100	—	75	ns
18	TdIEI (IEOr)	Delay from IEI rise to IEO rise (after ED decode)	—	110	—	80	ns
19	TdC (INT)	Delay from clock rise to $\overline{INT}$ fall	—	(1) + 120	—	(1) + 100	ns
20	TdCLK (INT)	Delay from CLK/TRG rise to $\overline{INT}$ fall (Counter mode) tsCTR (C) Satisfied tsCTR (C) not Satisfied	 — —	 (19) + (26) (1) + (19) + (26)	 — —	 (19) + (26) (1) + (19) + (26)	 ns ns

050489

(2/2)

NO.	SYMBOL	PARAMETER	AP-6/AM-6/AT-6 (6MHz)		AP-8 (8MHz)		UNIT
			MIN.	MAX.	MIN.	MAX.	
21	TcCTR	CLK / TRG Frequency (COUNT MODE)	2TcC	—	2TcC	—	ns
22	TrCTR	CLK / TRG rising time	—	40	—	30	ns
23	TfCTR	CLK / TRG falling time	—	40	—	30	ns
24	TwCTRl	Low CLK/TRG pulse width	120	—	90	—	ns
25	TwCTRh	High CLK/TRG pulse width	120	—	90	—	ns
26	TsCTR (Cs)	Set-up time up to CLK/TRG rise for clock rise requiring immediate count (counter mode)	150	—	110	—	ns
27	TsCTR (Ct)	Set-up time up to CLK/TRG rise for clock rise requiring immediate start of prescaler (timer mode)	150	—	110	—	ns
28	TdC (ZC / TOr)	Delay from clock rise to ZC/TO rise	—	140	—	110	ns
29	TdC (ZC / TOf)	Delay from clock fall to ZC/TO fall	—	140	—	110	ns

050489

Note 1 : AC test condition
VIH = 2.4V, VIHC = V_{CC}-0.6V, VIL = 0.4V, VOH = 2.2V, VOL = 0.8V CL = 100pF

Note 2 : If the daisy chain is at N stage,
2.5TcC > (N-2) TdIEI (IEOf) + TdM1 (IEO) + TsIEI + TTL buffer delay must be satisfied.

Note 3 : (1) : Tcc, (19) : Tdc (INT), (26) : TsCTR (Cs)

4.4 CAPACITANCE

TA = 25°C

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	Clock Input Capacitance	f = 1MHz	—	—	5	pF
CIN	Input Capacitance	All terminals except that to be measured should be earthed.	—	—	6	pF
COUT	Output Capacitance		—	—	10	pF

050489

4.5 AC TIMING CHARTS

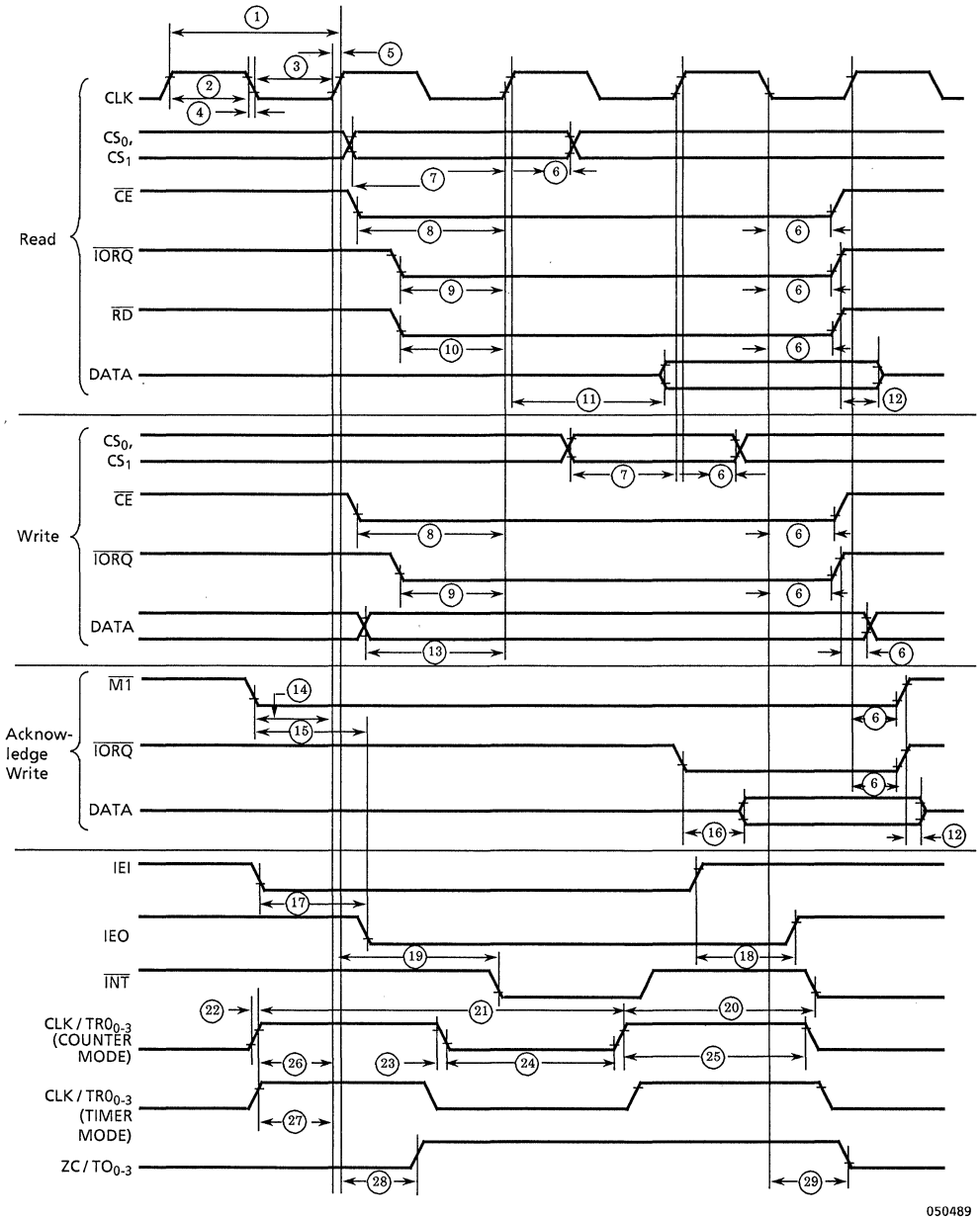


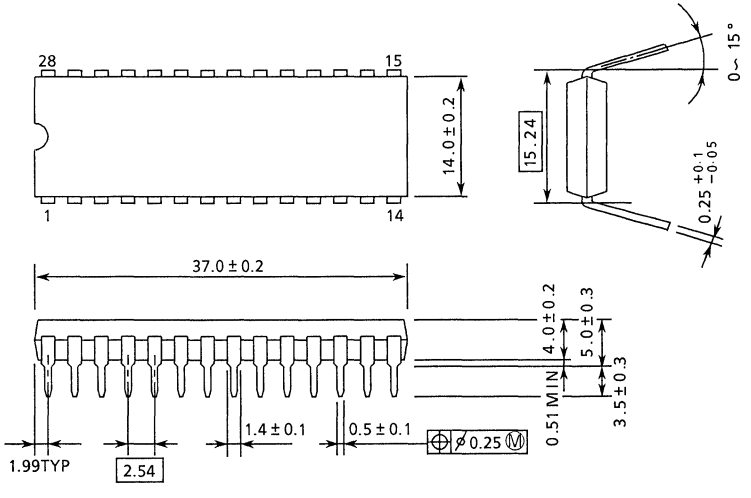
Figure 4.1 Timing Diagram

5. PACKAGE DIMENSION

5.1 DIP PACKAGE

DIP28-P-600

Unit : mm



270289

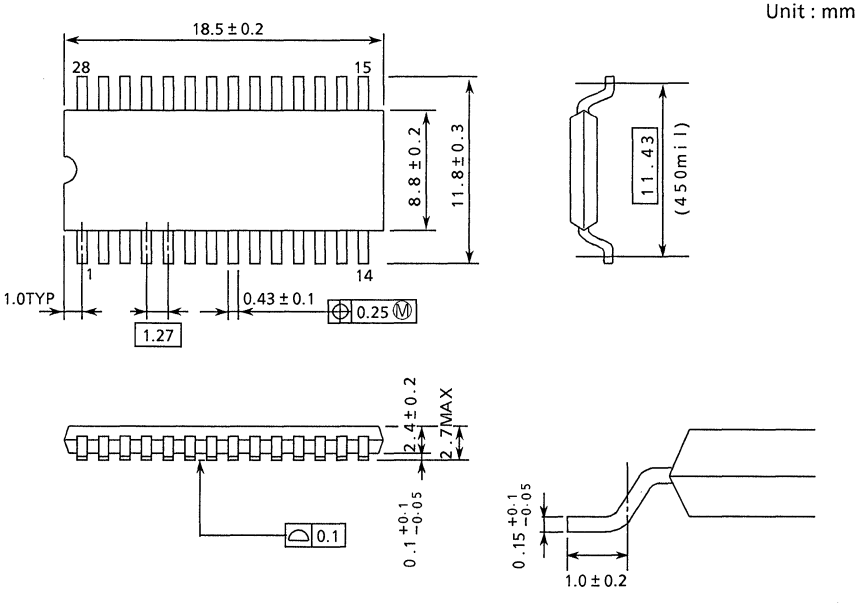
Note 1 : This dimension is measured at the center of bending points of leads.

Note 2 : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.28 leads.

5.2 SOP PACKAGE

SOP28-P-450

Unit : mm



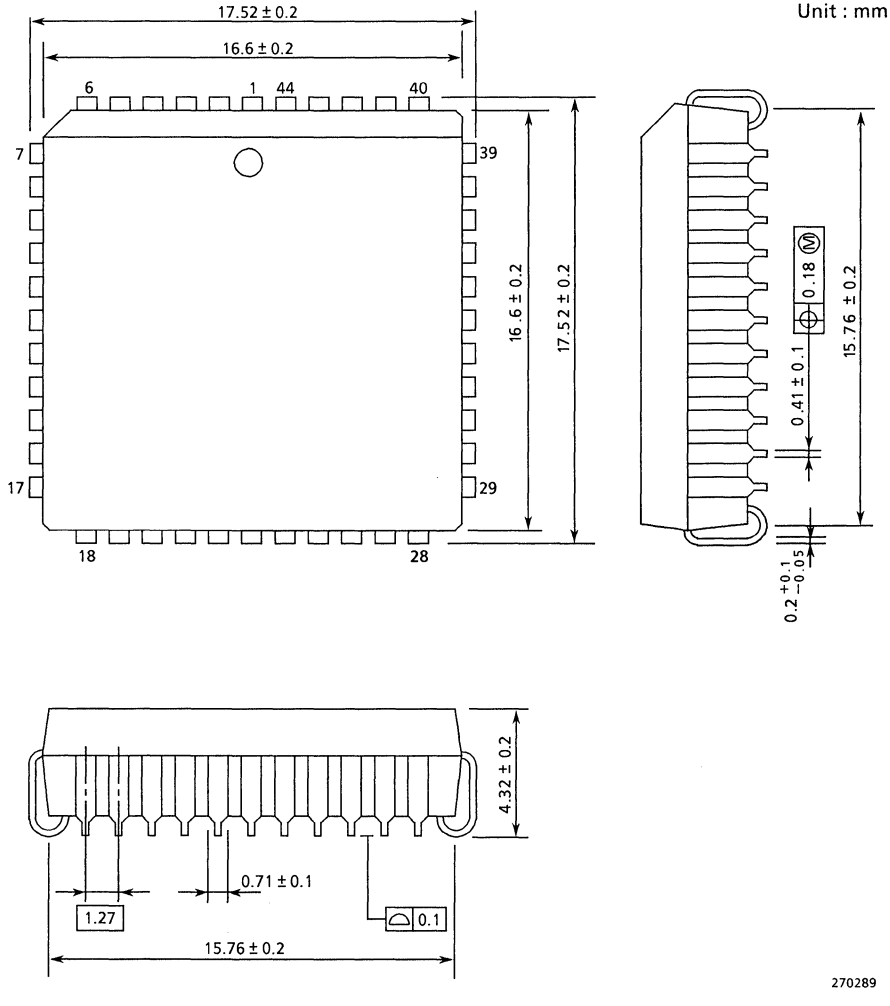
270289

Note : Package Width and length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

5.3 44-PIN PLCC PACKAGE

QFJ44-P-S650

Unit : mm



TMPZ84C40AP-6 / 41AP-6 / 42AP-6 / 43AF-6 / 44AT-6
TMPZ84C40AM-6 / 41AM-6 / 42AM-6
TMPZ84C40AP-8 / 41AP-8 / 42AP-8

TLCS-Z80 SIO: SERIAL INPUT/OUTPUT CONTROLLER

1. GENERAL DESCRIPTION AND FEATURES

The TMPZ84C40A (SIO/0), TMPZ84C41A (SIO/1), TMPZ84C42A (SIO/2), TMPZ84C43A, (hereinafter referred to as SIO) are CMOS version of Z80 SIO and have been designed to provide low power operation.

SIOs are designed for the adaptation to the various serial data communications which are needed to the microcomputer system.

SIOs are able to handle the asynchronous signal, the synchronous byte unit protocol and the synchronous bit unit protocol like HDLC and SDLC.

SIOs are fabricated using Toshiba's CMOS Silicon Gate Technology.

The principal functions and features of the SIOs are as follows.

(1) Compatible with the Zilog Z80 SIO.

(2) Compatible with the CCITT-X.25.

(3) Compatible with the HDLC/SDLC.

(4) Data transfer rate up to 800K bit/Sec.

(5) 2 independent full-duplex channels can be used.

(6) Built-in CRC generation and checking function.

(7) On chip daisy-chain structure interrupt circuit.

(8) Low power consumption

4mA Typ. (@5V @6MHz)··· TMPZ84C40/41/42AP-6, TMPZ84C40/41/42AM-6, TMPZ84C43AF-6, TMPZ84C44AT-6

(9) Single power supply: 5V±10% (6MHz VERSION),
5V±5% (8MHz VERSION)

(10) Extended operating temperature: −40°C to 85°C (6MHz VERSION),
−10°C to 70°C (8MHz VERSION)

(11) 40 pin DIP package, 40 pin SOP package, 44 pin Mini Flat package, 44 pin PLCC package.

Note : Z80 is a trademark of Zilog Inc., U.S.A.

2. PIN CONNECTIONS AND PIN FUNCTIONS

The pin connections, pin functions and functions of SIOs are described in this chapter.

2.1.1 Pin connections (DIP, SOP)

The pin connections of the SIOs are as shown in Figure 2.1, Figure 2.3, and Figure2.5.

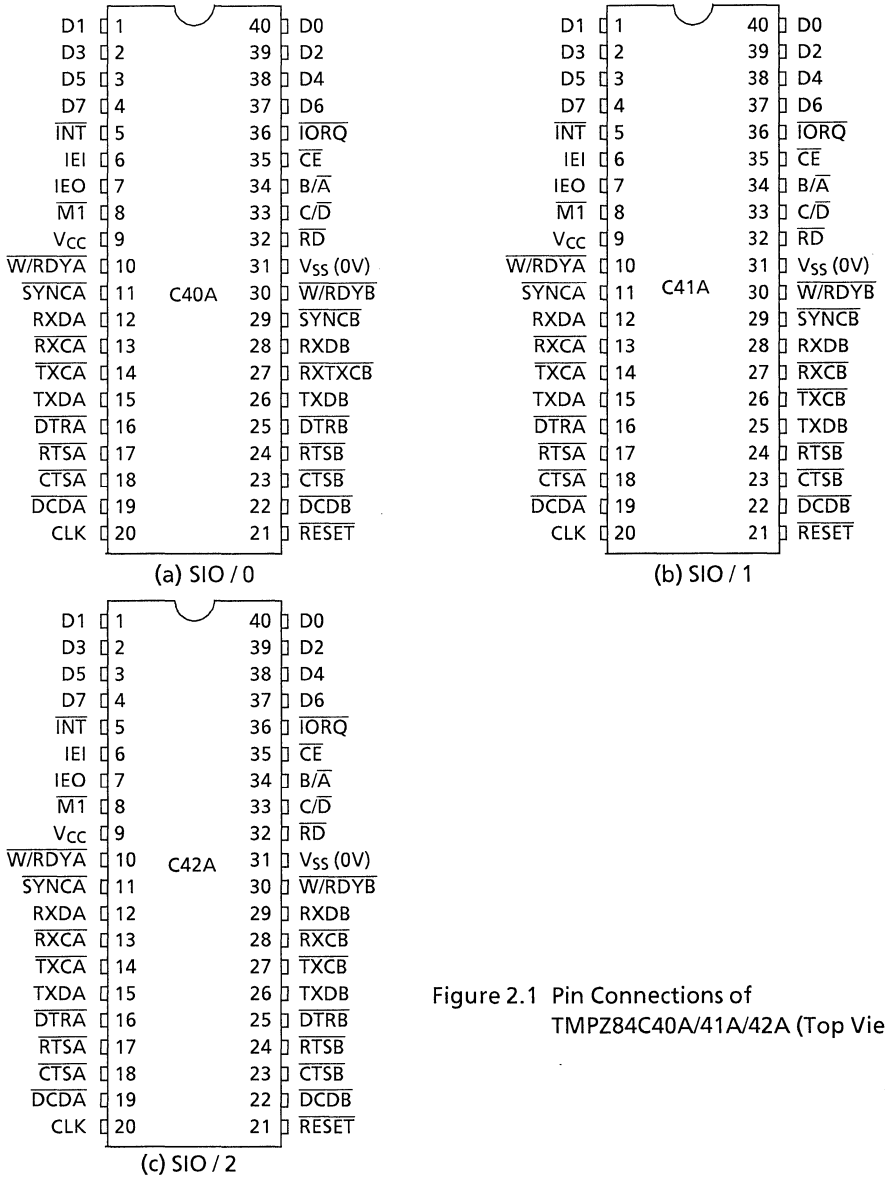


Figure 2.1 Pin Connections of TMPZ84C40A/41A/42A (Top View)

100489

However, it is necessary to choose the terminals in accordance with the purposes, because they are limited in number. The differences in SIO/0, SIO/1 and SIO/2 are shown in Figure 2.2 (a) SIOs version diagram.

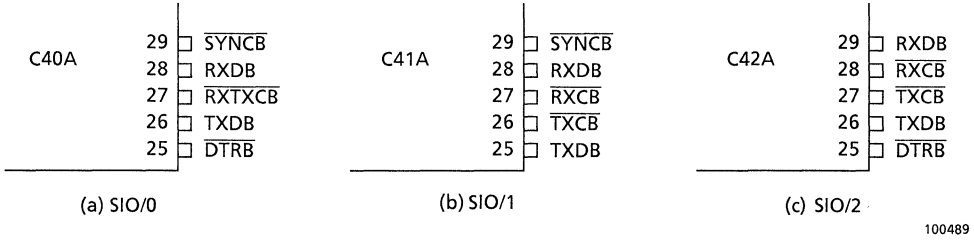
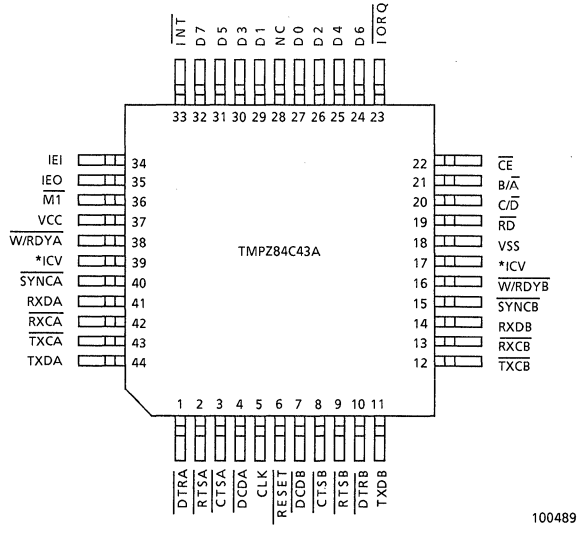


Figure 2.2 (a) SIOs Version (40-pin DIP, 40-pin SOP)

2.1.2 Pin connections (Mini Flat package)



Note 1: NC must be used at open condition.
Note 2: *ICV must be used at open condition or connected with VCC.

Figure 2.3 Pin Connections of TMPZ84C43A (Top View)

2.1.3 How to use TMPZ84C43A as SIO/0 or SIO/1 or SIO/2.

The Figure 2.4 shows six terminals to define TMPZ84C43A as SIO/0 or SIO/1 or SIO/2.

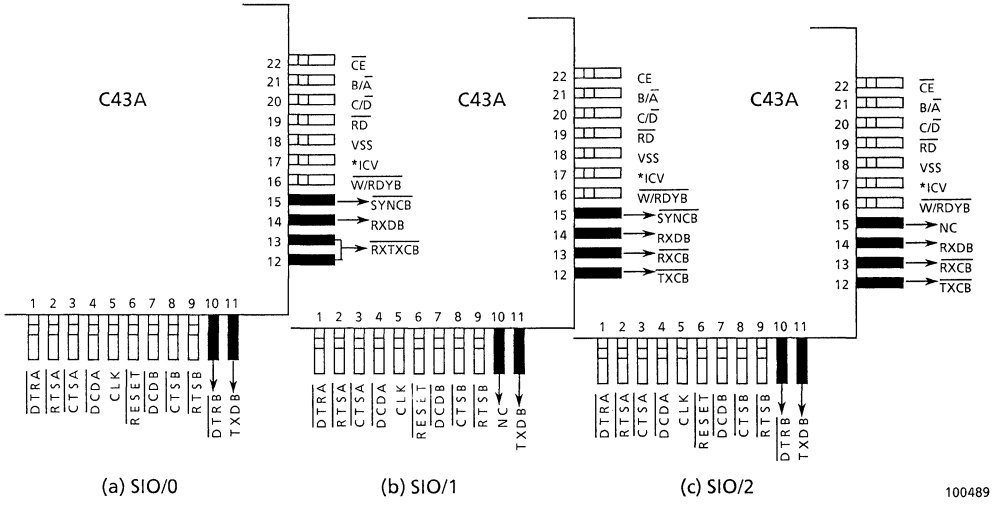
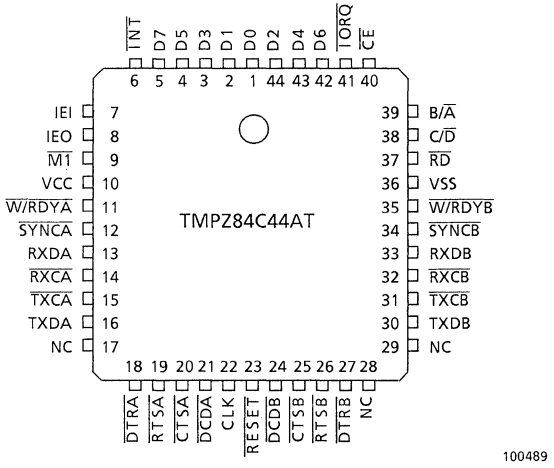


Figure 2.4 How to Use TMPZ84C43A as SIO/0 or SIO/1 or SIO/2

2.1.4 Pin connections (PLCC)



Note 1 : NC must be used at open condition.

Figure 2.5 Pin Connections of TMPZ84C44A (Top View)

2.1.5 How to Use TMPZ84C44A as SIO/0 or SIO/1 or SIO/2.

The Figure 2.6 shows six terminals to define TMPZ84C44A as SIO/0 or SIO/1 or SIO/2.

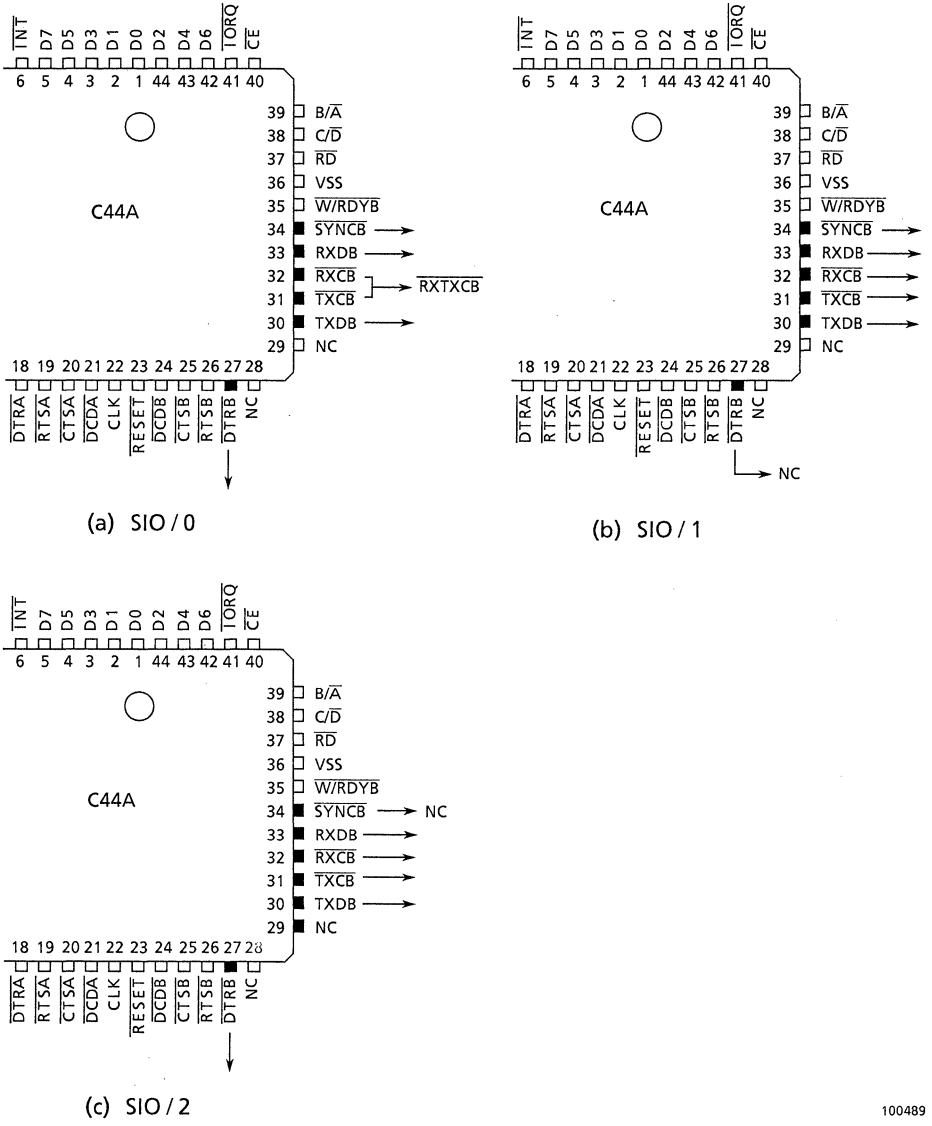


Figure 2.6 How to Use TMPZ84C44A as SIO/0 or SIO/1 or SIO/2

2.2 PIN NAMES AND FUNCTIONS

Table 2.2 Pin Names and Functions (1/2)

Pin Name	Number of Pin	Input/Output 3-state	Function
D0~D7	8	I/O 3-state	8-bit bidirectional data bus.
$\overline{INT}$	1	Output	Interrupt request signal. This is used, in case SIOs request MPU the interrupt. Wired OR connection is possible (because of the open drain).
IEI	1	Input	Interrupt enable input signal.
IEO	1	Output	Interrupt enable output signal. IEI and IEO are used for the daisy-chain structure. When IEI terminal is "1" and IEO terminal is "0", the SIO is being serviced by a MPU interrupt service routine.
$\overline{M1}$	1	Input	Machine cycle 1. When the both of $\overline{M1}$ and $\overline{IORQ}$ are "0", it is the interrupt acknowledge eycle by a MPU to the SIO's interrupt reguest.
$\overline{W/RDYA}$ $\overline{W/RDYB}$	2	Output	Wait/ready signal A, wait/ready signal B. These can be used as wait signal or ready signal according to SIOs-programming. SIO becomes active on "0", when pins are programmed as "WAIT" and are not ready to receive the data for MPU. SIO become active on "0", when pins are programmed as "READY" and ready to receive the data character for DMA.
$\overline{SYNCA}$ *$\overline{SYNCB}$	*2	I/O	Synchronous signal. In case of the asynchronous receive mode, These pins become the same input terminals as $\overline{CTS}$ and $\overline{DCD}$. In case of the external synchronous mode, pins become the input terminals and in case of the internal synchronous mode, become the output terminals.
RxDA RxDB	2	Input	Serial receive data
$\overline{RxCA}$ *$\overline{RxCB}$	*2	Input	Receive clock signal. In asynchronous mode, pins can choose the receive clocks which are 1, 16, 32 and 64 times as large as the data transfer rate according to the program.
TxDA TxDB	2	Output	Serial transmit data
$\overline{DTRA}$ *$\overline{DTRB}$	2	Output	Data terminal ready signal. These pins output the possibility or impossibility of the serial data receive.
$\overline{RTSA}$ $\overline{RTSB}$	2	Output	Transmit request signal. (Request to send) In serial data transmit, become "0".

100489

Table 2.2 Pin Names and Functions (2/2)

Pin Name	Number of Pin	Input/Output 3-state	Function
$\overline{CTSA}$ $\overline{CTSB}$	2	Output	Transmittable signal. (Clear to send) When terminals are "0", SIOs can receive the serial data transmit of the modem which has sent these signals.
$\overline{DCDA}$ $\overline{DCDB}$	2	Input	Data carrier detect signal. When terminals are "0", SIOs can enable the serial data transmit.
CLK	1	Input	Signal-phase clock input. Inputs Z80 standard system clock of single-phase. When CLK terminal is DC state ("1" or "0"), SIOs are in stationary state.
$\overline{IORQ}$	1	Input	I/O request signal In case both of $\overline{IORQ}$ and $\overline{CE}$ are "0", the data or command are transferred between MPU and SIO by the combination of B/$\overline{A}$ and C/$\overline{D}$.
$\overline{CE}$	1	Input	Chip enable signal. When input becomes "0", SIOs are enabled.
B / $\overline{A}$	1	Input	Channel select signal Selects the channel (A/B)
C / $\overline{D}$	1	Input	Command/data select signal. Selects the command and data.
$\overline{RD}$	1	Input	Read signal. In case both of $\overline{CE}$ and $\overline{IORQ}$ are "0", if $\overline{RD}$ is "0", this pin performs the read operation and if $\overline{RD}$ is "1", this pin performs the write operation.
$\overline{RESET}$	1	Input	Reset signal. If $\overline{RESET}$ is turned into "0", the receiver and transmitter become disabled and the serial data become the mark state.
*$\overline{RxTxCB}$	*1	Input	Bonding terminal of $\overline{TxCB}$ and $\overline{RxCB}$.
Vcc	1	Power Supply	+ 5V
Vss	1	Power Supply	0V

100489

Note: The asterisk (*) mark is difference in accordance with the three versions (SIO/0, SIO/1, SIO/2).

3. FUNCTIONAL DESCRIPTION

3.1 BLOCK DIAGRAM

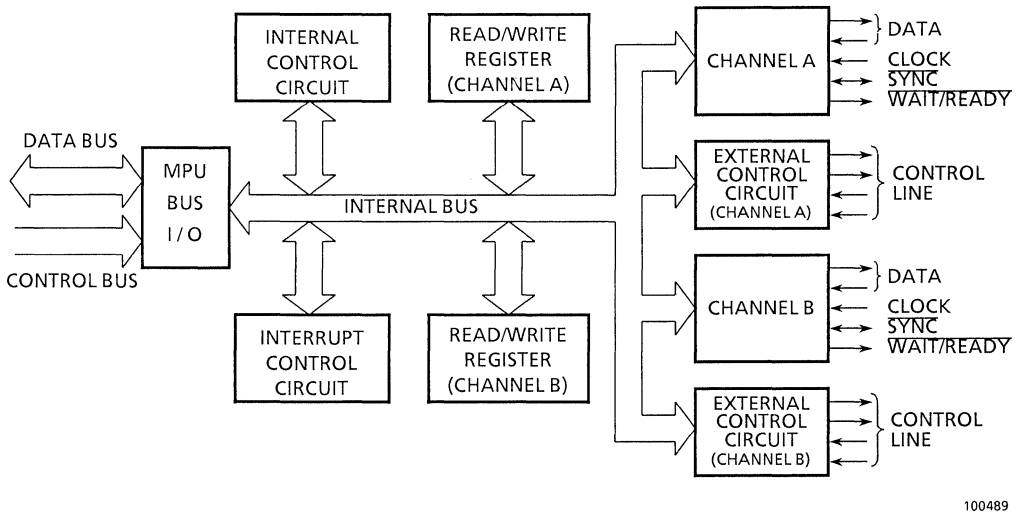


Figure 3.1 Block Diagram

3.2 SYSTEM CONFIGURATION

3.2.1 Architecture

As shown in Figure 3.1, the SIO is composed of MPU bus interface, the internal control circuit, the interrupt control circuit and two full-duplex channels which operate independently. Each channel has the read register, the write register and the external control circuit which is connected to the peripheral LSI or the external device.

Table 3.1 shows the registers in the SIO and their functions. Each channel has eight write registers and three read registers. Refer to 3.4 SIO programming for details.

(1) Communication data path

Figure 3.2 shows the communication path of the transmit/receive data of each channel.

① Receiving

The receiver has an 8-bit receive shift register and a 3-stage 8-bit buffer register in FIFO configuration. This saves time in high-speed data block transfers. The receivers also have the receive error FIFO which holds the status information such as parity and framing errors.

The receive data follow different paths according to the operation mode and character length as shown in Figure 3.2.

Table 3.1 (a) Write Registers

Register		Function
Write register 0	(WR0)	Resets CRC. Sets pointers of registers, and commands.
Write register 1	(WR1)	Sets the interrupt mode.
Write register 2	(WR2)	Sets the vector to be transmitted at interrupt. (Channel B only)
Write register 3	(WR3)	Provides the parameters to control the receiver.
Write register 4	(WR4)	Provides the parameters to control the receiver and transmitter.
Write register 5	(WR5)	Controls the transmitter.
Write register 6	(WR6)	Sets the sync character or the SDLC address field.
Write register 7	(WR7)	Sets the sync character or the SDLC flag.

100489

Table 3.1 (b) Read Registers

Register		Function
Read register 0	(RR0)	Indicates the receive/transmit buffer state and the pin state.
Read register 1	(RR1)	Indicates the error status and the end-of-frame code.
Read register 2	(RR2)	Indicates the interrupt vector contents. (Channel B only)

100489

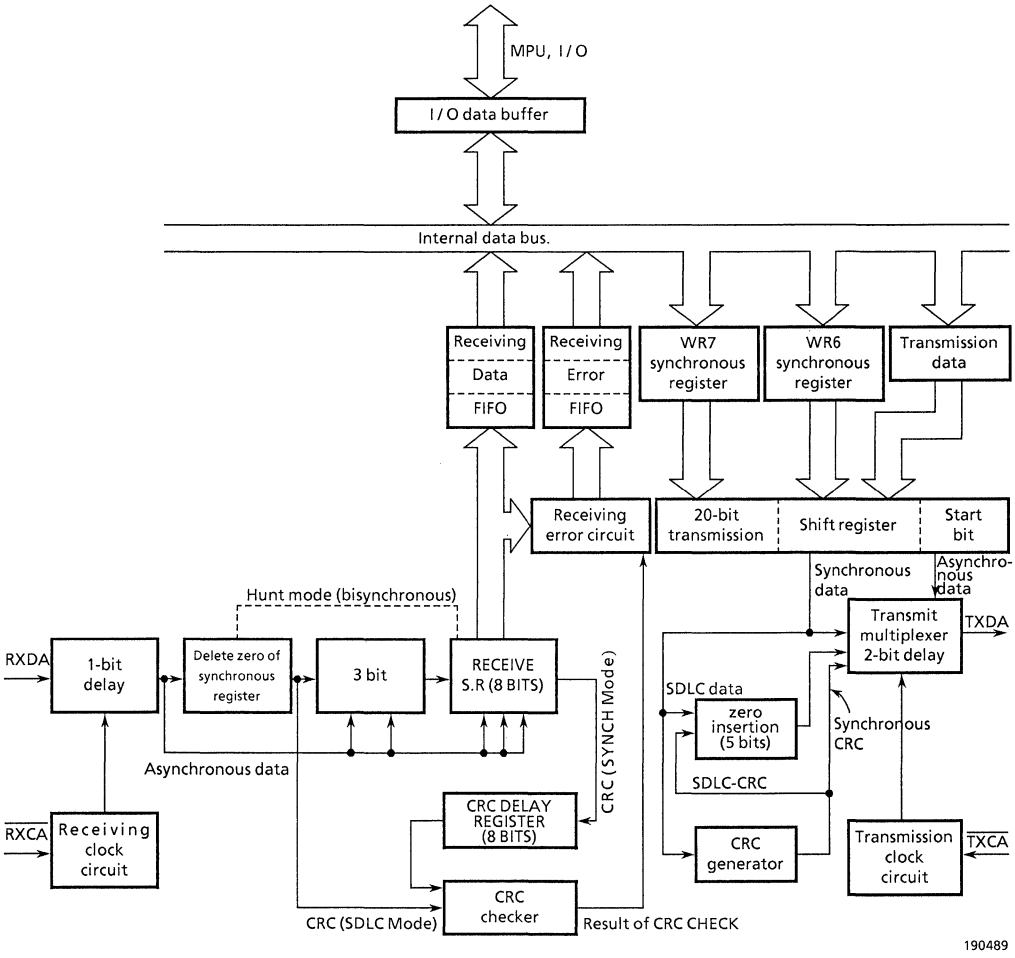


Figure 3.2 Transfer Data path (Channel A)

- Asynchronous mode

  In the asynchronous mode, the receive data enters the 3-bit buffer if the character length is 7 or 8 bits or the 8-bit receive shift register if the character length is 5 or 6 bits.

- Synchronous mode

  In the synchronous mode, the data path depends on the receive processing phase at the time.

The receiver operation starts from the hunt phase. In this mode, the receiver searches the receiver data for the bit pattern which matches the specified sync character. If the SIO is set in the monosync mode, the receiver searches for the bit pattern which matches the sync character set in WR7; if the SIO is set in the bisync mode, the receiver searches for the bit pattern which matches two consecutive sync characters set in WR6 and WR7. When synchronization has been established, the subsequent data enter the 3-bit buffer by bypassing the sync register.

- SDLC mode

In the SDLC mode, the sync register constantly monitors the receive data performing zero deletion as required. When the sync register detects 5 "1" s consecutively in the receive data, the following bit is deleted if it is "0". If it is "1", the bit that follows is checked. If it is "0", it is assumed as a flag, if it is "1", it is assumed an abort sequence (7 consecutive "1"s).

The reformatted data are put in the receive shift register via the 3-bit buffer. When synchronization has been established, the subsequent data follow the same path regardless of the character length.

② Transmission

The transmitter has an 8-bit transmit data register and a 20-bit transmit shift register. The 20-bit transmit shift register holds the data from the WR6, WR7, and transmit data register.

- Asynchronous mode

In the asynchronous mode, the data in the 20-bit transmit shift register are added with the start and stop bits to be sent to the transmit multiplexer.

- Synchronous mode

In the synchronous mode, the WR6 and WR7 hold the sync character. The contents of these registers are sent to the 20-bit transmit register as the sync character at the transmission of data blocks or as the idle sync character if a transmitter underrun occurs in data block transmission.

- SDLC mode

In the SDLC mode, the WR6 holds the station address and the WR7 holds the flag. The flag (WR7) is sent to the 20-bit transmit register at the start and end of each frame. For each of the other data fields, one "0" follows five consecutive "1"s.

(2) I/O functions

To transfer data from/to the MPU, the SIO must be set in the polling, interrupt, or block transfer mode.

- Polling

  To operate the SIO in the polling mode, all interrupts mode must be disabled. In the polling mode, the MPU reads the status bit D0 and D2 in each channel's RR0 to check for reception or transmission.

- Interrupts

  There are 3 types of SIO interrupt: transmit interrupt, receive interrupt, and external/status interrupt. These interrupts can be enabled by program. The receive interrput is further divided into the following three:

  - Interrupt on the first received character
  - Interrupt on all received characters
  - Interrupt on special receive conditions

  Higher priority is given to channel A than channel B. On the same channel, higher priority is given to reception, transmission, and external/status in this order.

  The SIO provides the daisy-chained interrupt priority control feature and the interrupt vector generating feature. Further, it provides the "status affected vector" feature. This feature outputs 4 interrupts depending on the interrupt source.

- Block transfer

  The SIO has the block transfer mode to adapt to the MPU's block transfer and the DMA controller. For block transfer, the $\overline{W/RDY}$ line is used. For the MPU's block transfer, this line is used as the wait line; for the DMA block transfer, it is used as the ready line. The SIO's ready output indicates to the DMA controller that the data is ready to transfer. The SIO's wait output indicates to the MPU that the SIO is not ready for data transfer and therefore requesting the extension of the output cycle.

3.2.2 SIO basic operations

(1) Asynchronous mode

For data transfer in the asynchronous mode, the character length, clock rate, and interrupt mode must be set. These parameters are written in the write registers. Note that WR4 must be set before the other registers are set.

Data transfer does not start until the transmit eneble bit is set. When the auto enable bit is set, the SIO starts transmission upon the $\overline{CTS}$ pin's going "0", allowing the programmer to send a message to the SIO without waiting for the $\overline{CTS}$ signal. Figure 3.3 shows the data format of the asynchronous mode.

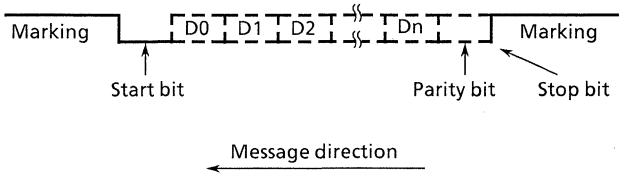


Figure 3.3 Data Format of Asynchronous Mode

① Transmission

Serial data are output from the TxD pin. Its transfer clock rate can be set to one of 1, 1/16, 1/32, and 1/64 times the clock rate to be supplied to the transmit clock input ($\overline{TxC}$). The serial data are output on the falling edge of $\overline{TxC}$.

② Reception

The receive operation in the asynchronous mode starts when the receive enable bit (D0 of WR3) is set. When the receive data input RxD is set to "0" for the duration of at least 1/2 bit time, the SIO interprets it as the start bit, sampling the input data at the middle of the bit time. The sampling is performed on the rising edge of the $\overline{RxC}$ signal.

When the receiver receives the data whose character length is not 8 bits, it converts the data into the one composed of the necessary bits, the parity bit and the unused bit set to "1".

Example : a 6-bit character

"	1	P	D5	D4	D3	D2	D1	D0	"

When the external/status interrupt is enabled and a break state is detected in the receive data, the interrupt is generated and the break/abort status bit (D7 of RR0) is set and the SIO monitors the transmit data until the break state is cleared. The interrupt is also generated when the $\overline{DCD}$ signal is in the inactive state for more than the specified pulse width. The DCD status bit is set to "1".

In the polling mode, the MPU must refer the receive character valid bit (D0 of RR0) to read the data. This bit is automatically reset when the receive buffer is read. In the polling mode, the transmit buffer status must be checked before writing data in the transmitter to avoid overwwrite.

(2) Synchronous mode

There are 3 kinds of character synchronization: monosync, bisync, and external sync. In each of these synchronous modes, the times 1 clock rate is used for both transmission and reception. The receive data is sampled on the rising edge of the receive clock input ($\overline{RxC}$).

The transmit data changes on the falling edge of the transmit clock input.

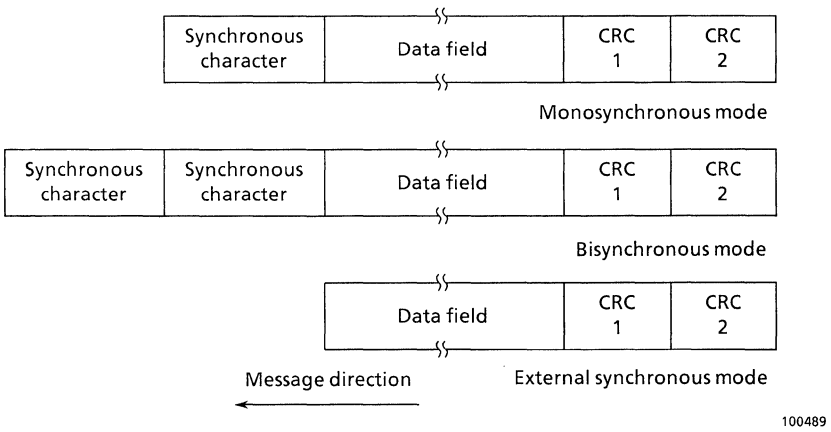


Figure 3.4 Data Format of Synchronous Mode

① Monosync

In this mode, synchronization is established when a match with the sync character (8 bits) set to WR7 is found, enabling data transfer.

② Bisync

In this mode, synchronization is established when a match with 2 consecutive sync characters set to WR6 and WR7 is found, enabling data transfer. In this mode as well as the monosync mode $\overline{\text{SYNC}}$ is active during the receive clock period in which the sync character is being detected.

③ External sync

In this mode, synchronization is performed externally. When synchronization is established, it is indicated by the $\overline{\text{SYNC}}$ pin. The $\overline{\text{SYNC}}$ input must be kept to "0" until the character synchronization is lost. Character assembly starts from the rising edge of the $\overline{\text{RxC}}$ after the falling of the $\overline{\text{SYNC}}$.

After reset, the SIO enters the hunt phase to search for the sync character. If synchronization is lost, the SIO sets the enter-hunt-phase-bit (D4 of WR3) to reenter the hunt phase.

- Transmission

a. Data transfer using interrupt

When the transmit interrupt is enabled, the interrupt is caused upon the transmit buffer's being emptied. For the interrupt processing, other data are written in the transmitter. If these data are not ready for some reason, the transmit underrun condition occurs.

b. Bisync mode

In the bisync mode, if the transmitter runs out of data during transmission, supply characters are inserted. This is done in two methods. In one method, sync characters are inserted. In the other, characters generated so far are transmitted followed by sync characters. Either of these methods can be selected by the reset transmit underrun/EOM command in WR0.

c. End of transmission

Break can be performed by setting bit D4 of WR5. When break is performed, the data in the transmit buffer and the shift register are lost. When the external/status interrupt is enabled, the SIO generates the interrupt depending on the transmitter state and outputs the vector. This mode can be used for block transfer.

- Reception

a. Interrupt on the first received character

This mode is used for ordinary block transfer. In this mode, the SIO generates the interrupt only for the first character; subsequently, it does not generate the interrupt unless special receive conditions are satisfied.

To initialize these settings, command 4 of WR0 (to be enabled by the next receive interrupt) must be set in advance.

b. Interrupt on all received characters

In this mode, the SIO generates the interrupt for all characters coming into the receive buffer. When the status affect vector has been set, a special vector is generated on a special receive condition.

c. Special receive condition interrupt

This interrupt occurs when any of the above interrupts is selected. The special receive conditions include parity error, receive overrun error, framing error, and end-of-frame (SDLC). These error status bits are latched, so that they must be reset after they are read.

They can be reset by command 6 of WR6 (error reset).

(3) SDLC mode

The SIO supports both the SDLC and HDLC protocols. They resemble each other, so that only the SDLC mode is explained here.

Figure 3.5 shows the data format in the SDLC mode. In the SDLC mode, one data block is called a frame and the message in it is put between the open flag and the close flag. The address field in the frame contains the address of a secondary station. Checking this address, the SIO receives or ignores the frame.

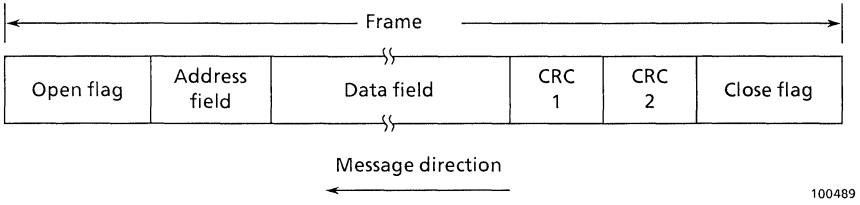


Figure 3.5 Data Format in SDLC Mode

- Transmission

  a. Data transfer using interrupt

  When the transmit interrupt has been set, the interrupt occurs each time the transmit buffer becomes empty. In the SDLC mode, data are sent to the SIO by this interrupt.

  b. Data transfer using wait/ready

  The wait function in the wait/ready capability is used to make the MPU extend the output cycle when the SIO's transmit buffer is not empty. The ready function indicates to the DMA that the SIO's transmit buffer is empty and therefore ready to receive data. If no data has been written in the transmit shift register before transmission, the SIO goes in the underrun state. This capability permits data transfer to the SIO.

  c. Transmit underrun/EOM

  The SIO automatically ends the SDLC frame if there is no data to be transmitted to the transmit data buffer. To implement this, the SIO sends a 2-byte CRC when there is no data to send, then the SIO transmits one or more flags. After reset, the transmit underrun/ EOM status bit is set to prevent the CRC character from being inserted when there is no data to be sent. Using this function, the SIO starts frame transmission. Here, the transmit underrun/EOM reset command must be set in advance between the transmission of the first data and the data end. Thus, the SIO goes in the reset state at the end of each message with the CRC character being sent automatically.

  d. CRC generation

  For CRC calculation, the CRC generator must be reset before transmission (bits D6 and D7 of WR0). CRC calculation starts when the address field is written in the SIO (WR6).

  The transmit CRC enable bit (D0 of WR5) must be set before the address field is written.

e. End of transmission

When the transmitter is disabled during transmission, the data currently transmitted is all transmitted to its end. The subsequent data is put in the marking state. When the transmitter is disabled, characters remain in the buffer. However, the abort sequence is made active when the abort command is written in the command register, deleting all data.

- Reception

As in the transmit mode, several parameters must be preset in the receive mode. The address field is written in WR7 and the flag character in WR7. Receiving the open flag, the receiver compares the contents of the following address field with the address set in WR6 or the global address ("1111 1111"). If the contents of the address field in frame matches either of these address, the SIO starts reception.

a. Interrupt on the first received character

This mode is generally used for the block transfer using the wait/ready capability. In this mode, the SIO generates the interrupt only on the first character. The stauts flag of this interrupt is latched, so that command 4 (to be enabled by the next received character) of WR0 must be preset for re-initialization. When the external/status interrupt is set, an interrupt occurs every time the $\overline{DCD}$ changes. This interrupt also occurs when the special receive condition is satisfied.

b. Interrupt on all received characters

In this mode, the SIO generates an interrupt on all received characters. When the status affect vector has been set, the SIO generates a special vector on the special receive condition interrupt.

c. Special receive condition interrupt

Using the special receive condition, the interrupt on the first received character or the interrupt on all received characters must be selected in advance. The receive overrun status of the special receive condition interrupt is latched. The status bit can be reset by the error reset command (WR0 command).

d. CRC check

The receive CRC check is reset when the open flag at the head of a frame is received. CRC calculation is performed on the subsequent characters up to the close flag. In the SDLC mode, the transmit CRC is inverted, so that a special check sequence is used. The check must end with "0001 1101 0000 1111". Since SIO handles the CRC character as a data, the MPU must discard it after reading it.

e. End of transmission

When the SIO receives the close flag, the end-of-frame-bit is set to indicate that the close flag has been received. When the status affect vector has been set, the special receive condition interrupt occurs and the interrupt vector is output. Any frame can be aborted by abort transmission. When the external/status interrupt has been set, the interrupt occurs and the break/abort bit in RR0 is set.

3.3 SIO STATUS TRANSITION DIAGRAM AND BASIC TIMING

3.3.1 Status transition diagram

Figure 3.6 shows the SIO status transition diagram.

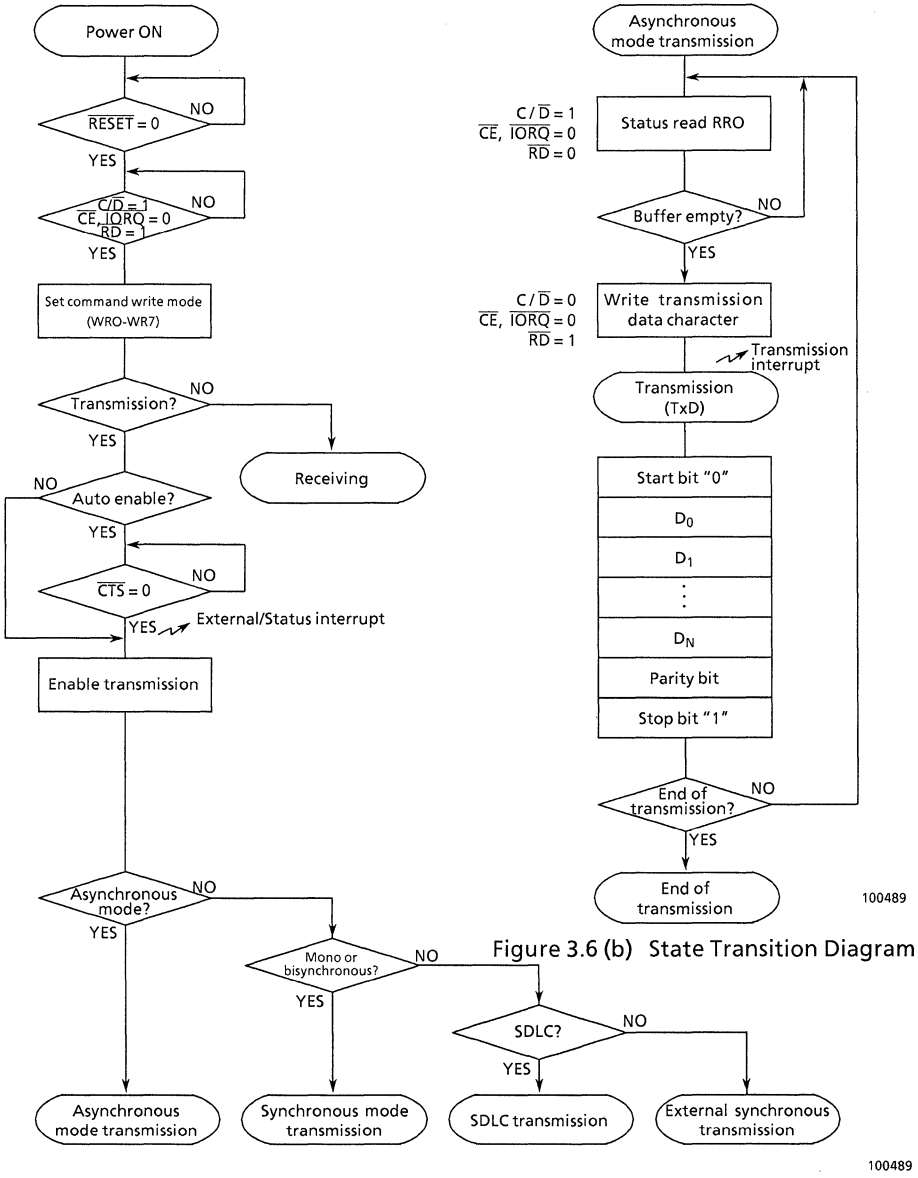


Figure 3.6 (b) State Transition Diagram

Figure 3.6 (a) SIO Status Transition Diagram

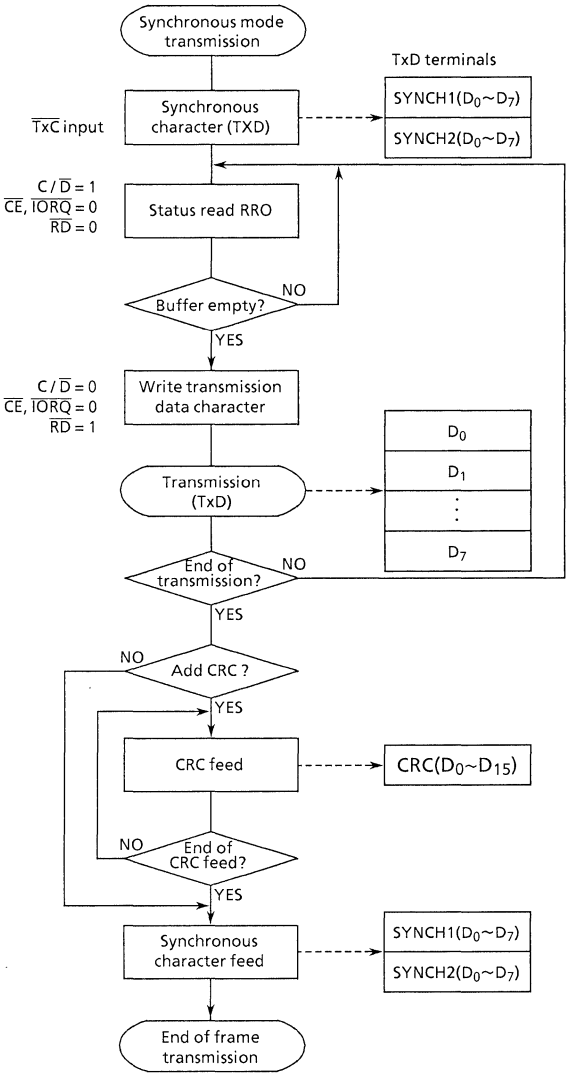
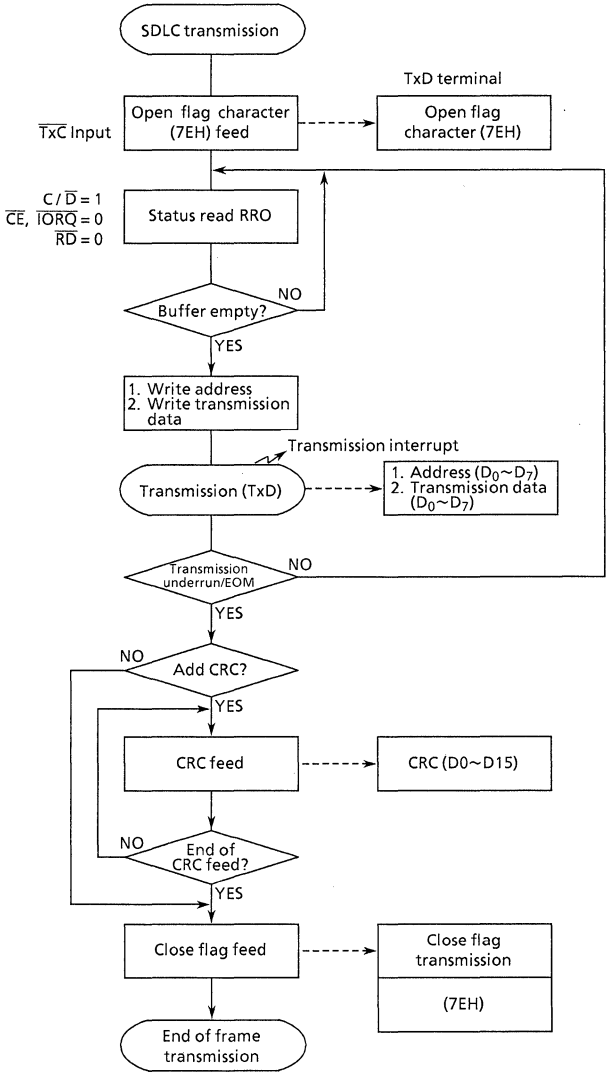
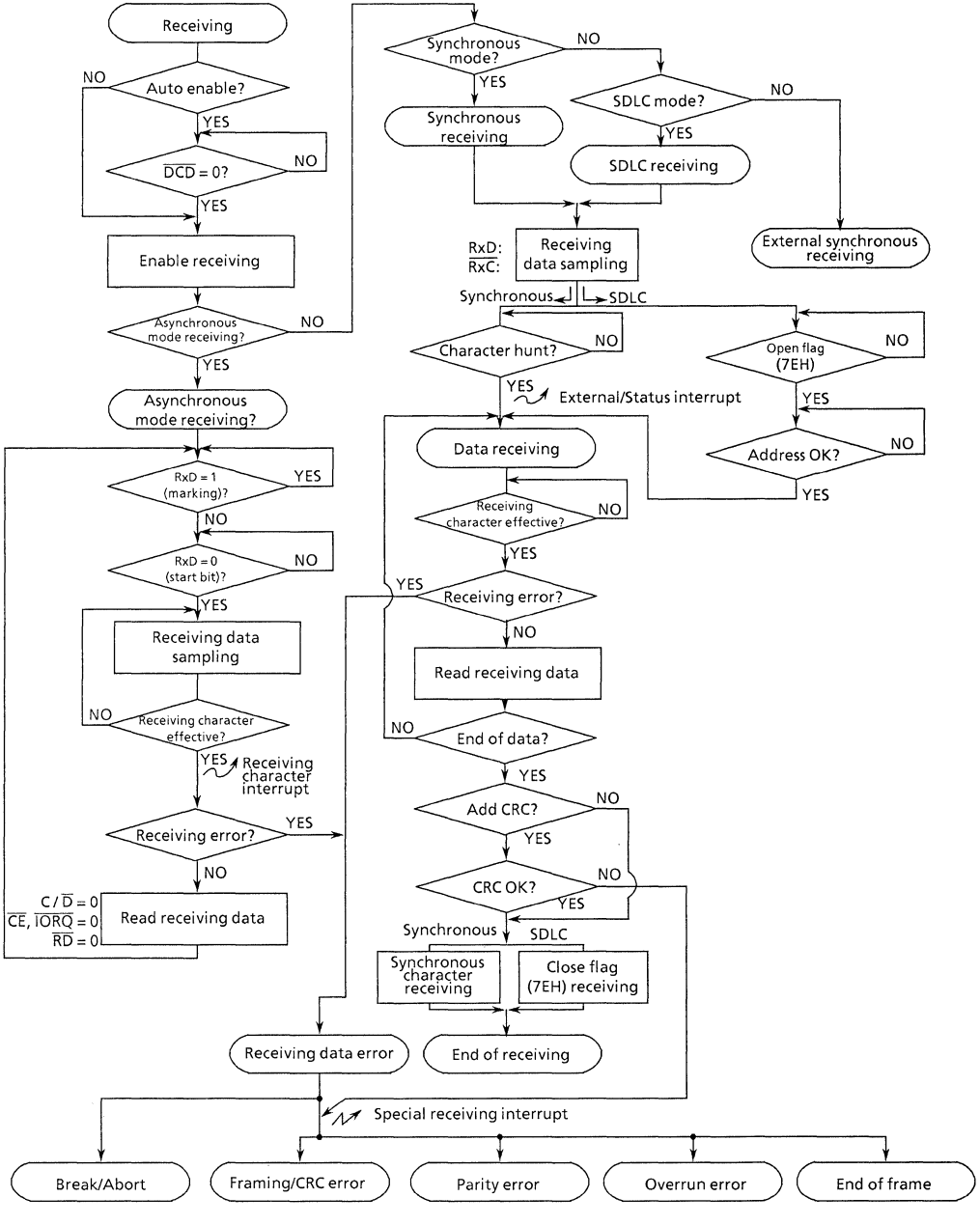


Figure 3.6 (c) State Transition Diagram



100489

Figure 3.6 (d) SIO Status Transition Diagram



100489

Figure 3.6 (e) State Transition Diagram

3.3.2 Basic timing

Figure 3.7 shows the timing in which data or a command is written from the MPU to the SIO. Figure 3.8 shows the timing in which data is read from the SIO to the MPU. Figure 3.9 shows the interrupt acknowledge timing in which the MPU gives an interrupt response to the SIO's interrupt request to set the $\overline{IORQ}$ pin to "0" several clocks after setting the $\overline{M1}$ pin to "0" as the acknowledge signal. To maintain the interrupt serviced state in daisy chain structure, the interrupt request state cannot be changed while $\overline{M1}$ is active.

Figure 3.10 shows the timing in which the return from interrupt is performed.

Figure 3.11 shows how the daisy chain structure works. First, suppose that the SIO is servicing interrupt. When the PIO issues an interrupt request immediately before the first byte "EDH" of the RETI instruction is decoded with $\overline{M1}$ being active, "IEO" of the PIO goes "0". However, when "EDH" is decoded, the PIO's interrupt request is not acknowledged.

Therefore, the PIO's "IEO" returns to "1". When the second byte "4DH" is decoded, the SIO's "IEO" returns to "1". Therefore, the "IEI" and "IEO" of each peripheral LSI at this point of time all go "1", or out of the interrupt serviced state. The PIO keeps the $\overline{INT}$ pin at "0" until this state is set. Then, the interrupt is serviced starting with the peripheral device of the higher priority.

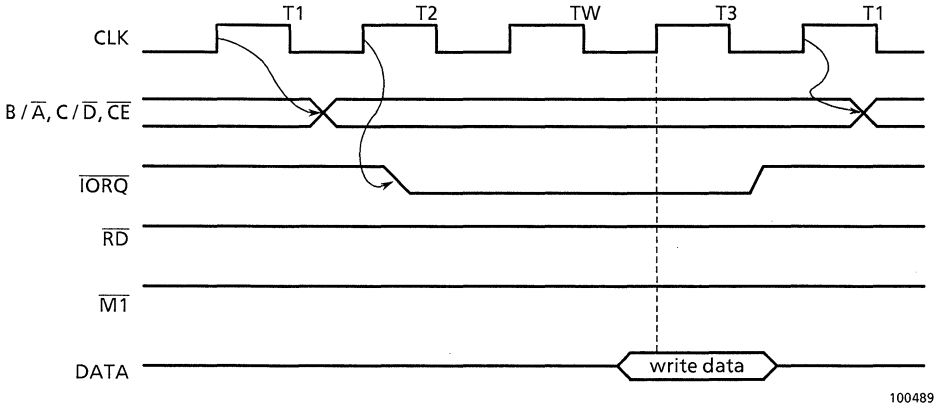


Figure 3.7 Write Timing

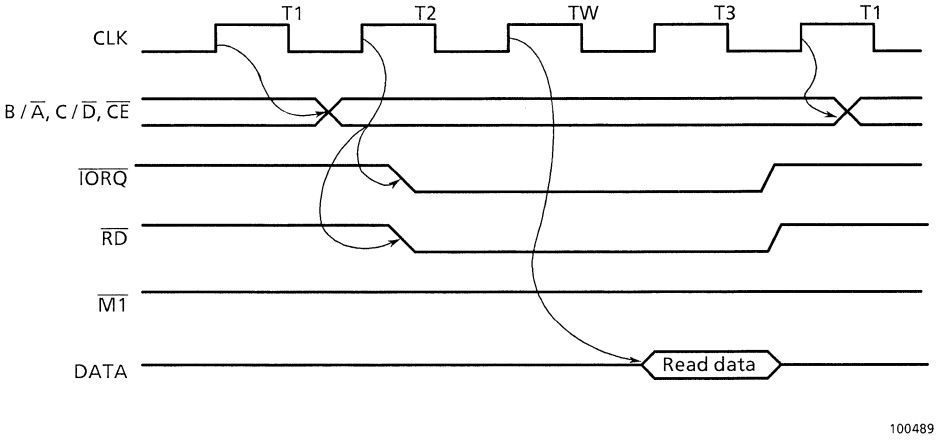


Figure 3.8 Read Timing

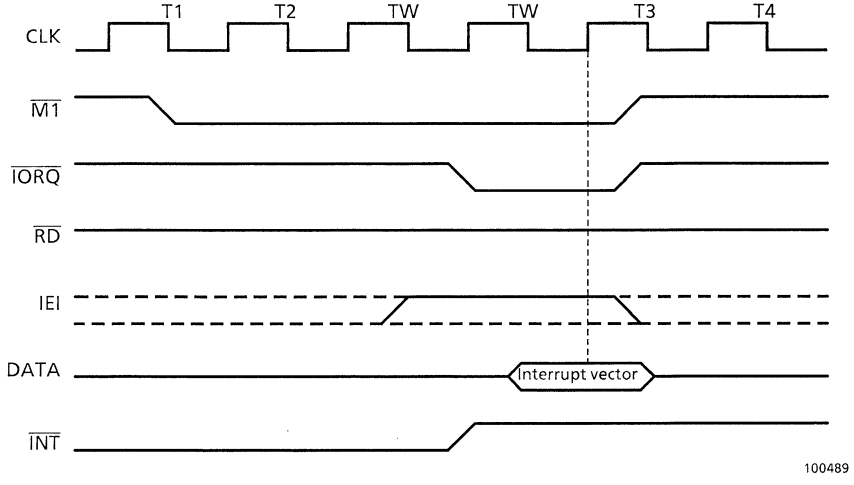


Figure 3.9 Interrupt Acknowledge Timing

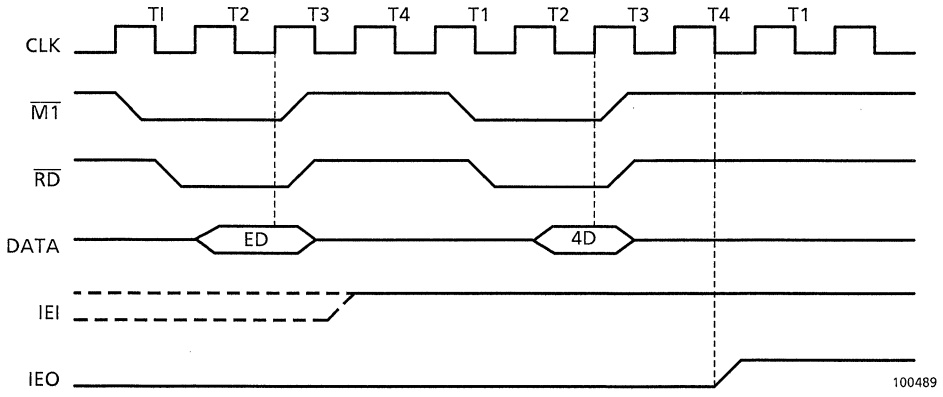
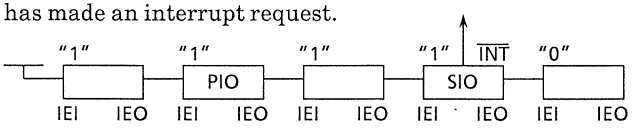
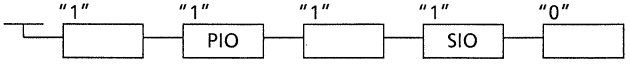


Figure 3.10 Return Timing from Interrupt

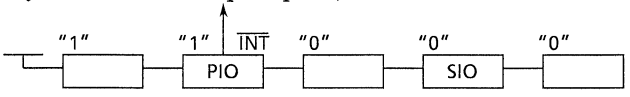
① The SIO has made an interrupt request.



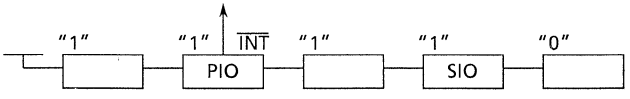
② The SIO is servicing the interrupt.



③ The PIO has made an interrupt request immediately before "EDH" is decoded by the SIO. By the PIO's interrupt request, PIO's IEO is set to "0".



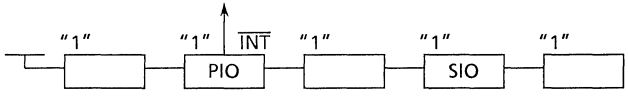
④ Because "EDH" has been decoded, the PIO's interrupt request is not acknowledged. Therefore, PIO's IEO returns to "1".



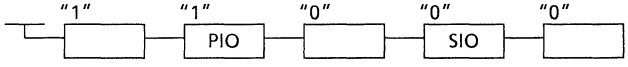
100489

Figure 3.11 Daisy Chain at Execution of RETI Instruction (1/2)

⑤ Because "4DH" has been decoded, the SIO's IEO is set to "1".



⑥ The PIO's interrupt request is acknowledged and the PIO's IEO is set to "0".



100489

Figure 3.11 Daisy Chain at Execution of RETI Instruction (2/2)

3.4 SIO OPERATIONAL PROCEDURE

The following mainly describes the meaning of each bit of the write and read registers. Special attention should be directed to the fact that the parameters of the write register (WR4) should be set before the others.

(1) Write registers

WR0; Write register 0

Table 3.2 Configuration of Write Register 0

D7	D6	D5	D4	D3	D2	D1	D0
CRC reset code		Primary command bit			Register pointer bit		

100489

Bits D0 through D2: Register pointer bits

These bits specify the register on which read/write is performed by the next byte. When read/write is completed, the register pointer points to WR0.

Bits D3 through D5: Basic command bits

- Command 0 (=000) : No operation

  This command only sets the register pointer without making the SIO operate. It is used to invalidate the command in the command chain for the SIO or hold the location at which a command is inserted in the command chain if required.

- Command 1 (=001) : Abort sequence generation

  This command is used to generate the abort sequence (7 or more consecutive "1"s). Note that command 1 is used only in the SDLC.

- Command 2 (=010) : External/status interrupt reset

  Once an external interrupt or a status interrupt has occurred, the status bit of RR0 is latched. This command is issued to enable the RR0's status bit in order to enable the interrupt again.

- Command 3 (=011) : Channel reset

  This command performs generally the same operation as when the $\overline{\text{RESET}}$ pin is set. The difference is that reset is performed only on a single channel. The command for channel A resets the interrupt priority circuit as well.

- Command 4 (=100) : Enable the interrupt at the next character reception.

  This command is used to enable an interrupt when the end of data block has been detected followed by the reception of the next block.

- Command 5 (=101) : Reset transmit interrupt pending

  If the transmit buffer becomes empty in the transmit interrupt enable mode, an interrupt occurs. This command is used to disable the transmit interrupt when there is no data in the transmit buffer.

- Command 6 (=110) : Error reset

  The error (parity or overrun error) caused in block transfer is latched in bits D4 and D5 of RR1. This commands is used to clear these bits.

- Command 7 (=111) : Return from interrupt

  This command performs the same operation as the operation required to execute the RETI instruction on the SIO's data bus. Therefore, non-Z80 CPUs (that is, systems using no RETI instruction) can use the daisy chain in the SIO. This command is available only on channel A.

Bits D6 and D7: CRC reset code

These 2 bits allow the programmer to select between the receive CRC checker reset, the transmit CRC generator reset, and the transmit underrun/EOM reset.

Table 3.3 List of Reset Command Codes

Reset command	D7	D6
No operation	0	0
Reset the receive CRC checker	0	1
Reset the transmit CRC generator	1	0
Reset the transmit underrun/EOM	1	1

100489

WR1; Write register 1

Table 3.4 Configuration of Write Register 1

D7	D6	D5	D4	D3	D2	D1	D0
Enable	Wait/ready Select function	Select receiving/ transmission	Receiving interrupt mode		Status -affect vector	Enable transmission interrupt	Enble external/ status interrupt

100489

Bit D0: External/status interrupt enable

When this bit is set, an interrupt is generated at the start of sync character transmission even if the execution is terminated upon detection of break/abort, the $\overline{DCD}$, $\overline{CTS}$ or $\overline{SYNC}$ signal has changed, or the transmit underrun/EOM latch is set.

Bit D1: Transmit interrupt enable

When this bit is set, a transmit interrupt is generated upon the transmit buffer becoming empty.

Bit D2: Status affect vector

When this bit is set, bits D1 through D3 (V1 through V3) of WR2 is changed. When this bit is not set, the same interrupt vector as the contents of WR2 issued. Note that this bit is available only on channel B.

Bits D3 and D4: Receive interrupt mode

These bits are used to select a receive interrupt mode.

Bits D5 through D7: Selection wait/ready functions

These 3 bits are used to select a $\overline{W/RDY}$ pin function. The wait or the ready function is selected by program and they are not used simultaneously. The meaning of these bits are:

- When D5 is set to "1", it indicates that the $\overline{W/RDY}$ pin responds to the receive buffer; when D5 is reset to "0", it indicates that the pin responds to the transmit buffer.
- When D6 is set to "1", the $\overline{W/RDY}$ pin functions as the $\overline{READY}$ pin; when D6 is reset to "0", the pin functions as the $\overline{WAIT}$ pin.
- When D7 is set to "1", the wait/ready function is enabled; when D7 is reset to "0", the function is disabled.

For example, when D7, D6, and D5 are "1", "1", and "0" respectively, and the transmit buffer is full, the $\overline{READY}$ pin goes "1"; when the transmit buffer is empty, the pin goes "0".

Table 3.5 shows the summary of the above description of bits D3 and D4 and D5 through D7.

Table 3.5 List of Receive Interrupt Mode Codes

Receive interrupt mode	D4	D3
Receive interrupt disable	0	0
Interrupt on first received character or special receive conditon*	0	1
Interrupt on received character or special receive condition*	1	0
Interrupt on received character or special receive condition* (except for parity error)	1	1

100489

* Special receive conditions:
- End of frame (in SDLC mode only)
- Receive overrun error
- Parity error
- Framing error

Table 3.6 Wait/Ready Select Function (D5 through D7)

Pin state			Buffer state	D7	D6	D5
Pin (Function)		Pin output				
DISABLE	$\overline{WAIT}$	Floating	–	0	0	–
	$\overline{READY}$	High	–		1	
ENABLE	$\overline{WAIT}$	Low	The transmit buffer is full and the SIO data port is selected.	1	0	0
		Floating	The transmit buffer is empty.			
	$\overline{READY}$	High	The transmit buffer is full.		1	
		Low	The transmit buffer is empty.			
	$\overline{WAIT}$	Floating	The receive buffer is full.		0	1
		Low	The receive buffer is empty and the SIO data port is selected.			
	$\overline{READY}$	Low	The receive buffer is full.		1	
		High	The receive buffer is empty.			

100489

WR2; Write register 2

Table 3.7 Configuration of Write Register 2

D7	D6	D5	D4	D3	D2	D1	D0
V7	V6	V5	V4	V3	V2	V1	V0

Subject to change under different interrupt conditions if the status-affect vector bit is set.

100489

This write register is the interrupt vector register. When bit D2 of WR1 is not set, the interrupt vector is issued. When bit D2 of WR1 is set, bits D1 through D3 (V1 through V3) are changed depending on the interrupt generation condition. This time, the contents of WR2 remain unchanged. Because WR2 is available only on channel B, WR2 must be programmed even if only channel A of the SIO is used.

Table 3.8 shows the WR2 bit states in the interrupt condition with the status affect vector being set.

Table 3.8 Channel Interrupt Condition Codes

Channel	Interrupt condition	V3	V2	V1
B	Transmit buffer empty	0	0	0
	Change of external/status	0	0	1
	Received character available	0	1	0
	Special receive condition *	0	1	1
A	Transmit buffer empty	1	0	0
	Change of external/status	1	0	1
	Received character available	1	1	0
	Special receive condition *	1	1	1

100489

* Special receive conditions :
- End of frame (in SDLC mode only)
- Receive overrun error
- Parity error
- Framing error

WR3; Write resgister 3

Table 3.9 Configuration of Write Register 3

D7	D6	D5	D4	D3	D2	D1	D0
Receiving bit /character		Auto enable	Enter hunt phase	Enable receiving CRC	Address search mode	Prohibit synchronous character load	Enable receiving

100489

Bit D0: Receive enable

When this bit is set, the receive operation starts. Because this bit is used to start the receive operation, it must be set after the receive-associated programming has been all completed.

Bit D1: Sync character load inhibit

When this bit is set in the sync mode, the sync character is not loaded into the receive buffer.

This bit is used to remove the sync character and idle sync from the received characters.

Bit D2: Address search mode

When this bit is set in the SDLC mode, any message having a programmed address or an address other than the global address (FFH) is not received by WR6. Thefore, the receive interrupt does not occur unless an address match occurs.

Bit D3: Receive CRC enable

When this bit is set, CRC calculation starts at the start of the last data transfer from the receive shift register to the receiver buffer.

Bit D4: Enter hunt Phase

When the establishment of synchronization is required, set this bit to enter the SIO into the hunt phase. The hunt phase is automatically cleared upon establishment of synchronization.

Bit D5: Auto enable

When this bit is set, the transmitter is enabled at the time the $\overline{CTS}$ pin is "0". When the $\overline{DCD}$ pin is "0", the receiver is enabled.

Bits D6 and D7: Receive character length

These bits are used to specify the number of receive bits which make up one character (character length). Table 3.10 shows the number of bits per character.

Table 3.10 Receive Character Length Codes

Bits/character	D7	D6
5	0	0
7	0	1
6	1	0
8	1	1

100489

WR4; Write register 4

Table 3.11 Configuration of Write Register 4

D7	D6	D5	D4	D3	D2	D1	D0
Clock mode		Synchronous mode		Stop bit		Parity Even/$\overline{\text{Odd}}$	Enable

100489

Bit D0: Parity enable

When this bit is set, 1-bit transmit data is added to the number of bits specified by D6 and D7 of WR3 and the data is received in the resulting number of bits. If a character length other than 8 bits is selected, the added parity bit is set to the MSB side to be transferred to the receive data FIFO. When the 8-bit character length is selected, the parity bit is not transferred to the receive data FIFO.

Bit D1: Parity even/odd

This bit is used to determine whether to perform transfer and check in even or odd parity. (Even parity = "1", odd parity = "0")

Bit D2 and D3: Stop bit length

These bits are used to select the stop bit length in the asynchronous mode. In the synchronous mode, both D2 and D3 must be set to "0".

Table 3.12 Stop Bit Length Codes

Stop bit	D3	D2
Sync mode	0	0
1 stop bit/character	0	1
1.5 stop bits/character	1	0
2 stop bits/character	1	1

100489

Bits D4 and D5: Sync mode

These bits are used to select the sync mode.

Table 3.13 Sync Mode Codes

Sync mode	D5	D4
8-bit sync mode	0	0
16-bit sync mode (bisync mode)	0	1
SDLC mode (flag character; 7EH)	1	0
External sync mode	1	1

100489

Bits D6 and D7: Clock mode

These bits are used to select the factor between the transmit/receive clock and the data transfer rate. In the synchronous mode, the ×1 clock mode must be set. In the asynchronous mode, the transmit side and the receive side must have the same factor.

Table 3.14 Clock Mode Codes

Clock mode (data transfer rate)	D7	D6
×1 data transfer rate	0	0
×16 data transfer rate	0	1
×32 data transfer rate	1	0
×64 data transfer rate	1	1

100489

WR5; Write register 5

Table 3.15 Configuration of Write Register 5

D7	D6	D5	D4	D3	D2	D1	D0
DTR	Transmit bit /character		Break transmission	Enable transmission	CRC-16 / SDLC	RTS	Enable CRC transmission

100489

Bit D0: Transmit CRC enable

When this bit is set at the time the transmit data is loaded from the transmit data buffer into the transmit shift register, the CRC calculation is performed on that data. If this bit is not set, the CRC calculation and transmission are not performed in the transmit underrun state in the synchronous or SDLC mode.

Bit D1: Request to send (RTS)

When this bit is set, the $\overline{RTS}$ pin goes "0". When this bit is not set, the $\overline{RTS}$ pin goes "1". In the asynchronous mode, the $\overline{RTS}$ pin goes "1" when the transmit buffer becomes empty. In the synchronous or SDLC mode, this bit state is followed by the $\overline{RTS}$ pin state.

Bit D2: CRC-16/SDLC

When this bit is set, the CRC-16 polynomial ($X^{16}+X^{15}+X^{2}+1$) is selected. When this bit is reset to "0", the CRC-CCITT polynomial ($X^{16}+X^{12}+X^{5}+1$) is selected.

Bit D3: Transmit enable

When this bit is set, the transmitter is enabled. Even if this bit is reset to "0" after the start of transmission, the sync character and the data being transmitted are transmitted to the last.

Bit D4: Transmit break

When this bit is set, transmitting any data forcibly puts the transmit data line (TxD pin) in the space state. When this bit is reset to "0", the TxD pin is put in the marking state.

Bits D5 and D6: Transmit character length

These bits indicate the character length of transmit data.

Table 3.16 Transmit Character Length Codes

Bits/character	D6	D5
Less than 5 bits	0	0
7 bits	0	1
6 bits	1	0
8 bits	1	1

100489

As shown in Table3.16, for the transmission of less than 5 bits (4 bits or 3 bits) per character, D6 and D5 are "0" and "0", which do not indicate how many bits the transmit data consists of. To solve this problem, the data characters must be processed by the format shown in Table 3.17. Note that D indicates data.

Table 3.17 Data Transfer Format with Transmit Data Consisting of Less than 5 bits

Transmit bits/character	D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	0	0	0	D
2	1	1	1	0	0	0	D	D
3	1	1	0	0	0	D	D	D
4	1	0	0	0	D	D	D	D
5	0	0	0	D	D	D	D	D

100489

Bit D7: data terminal ready

This bit indicates the $\overline{DTR}$ pin state. When this bit is set, the $\overline{DTR}$ pin goes "0", when it is reset, the $\overline{DTR}$ pin goes "1".

WR6; Write register 6

Table 3.18 Configuration of Write Register 6

D7	D6	D5	D4	D3	D2	D1	D0
SYNC							
7	6	5	4	3	2	1	0

100489

This register is programmed as follows:

- In the external sync mode : Transmit sync character
- In the monosync mode : Transmit sync character
- In the bisync mode : First sync character
- In the SDLC mode : Slave station address

WR7; Write register 7

Table 3.19 Configuration of Write Register 7

D7	D6	D5	D4	D3	D2	D1	D0
SYNC							
15 (7)	14 (6)	13 (5)	12 (4)	11 (3)	10 (2)	9 (1)	8 (0)

100489

This register is programmed as follows:

- In the monosync mode : Receive sync character
- In the bisync mode : Second sync character
- In the SDLC mode : Flag character (7EH)

This register is not used in the external sync mode.

(2) Read registers

RR0; Read register 0

Table 3.20 Configuration of Read Register 0

D7	D6	D5	D4	D3	D2	D1	D0
Break/ Abort	Transmission underrun /EOM	CTS	Synchro- nize/Hunt	DCD	Transmission buffer empty	Reserve interrupt	Receiving character effective

Used with the external/status interrupt

100489

Bit D0: Receive character available

This bit is set when the receive buffer holds characters of 1 byte or more. This bit is reset when the buffer becomes empty.

Bit D1: Interrupt pending

This bit is set when an interrupt occurs in the SIO regardless of the interrupt condition type.

This bit is available only on channel A.

Bit D2: Transmit buffer empty

This bit is set when the transmit data buffer becomes empty or the SIO is reset. However, in the sync and SDLC modes where the CRC character is being transmitted, bit D2 is reset.

Bit D3: Data carrier detect

This bit indicates the $\overline{DCD}$ pin input state. This bit is latched when the external/status interrupt occurs.

Bit D4: Sync/hunt

The meaning of this bit depends on the operation mode:

(i) Asynchronous mode

Bid D4 indicates the SIO's $\overline{\text{SYNC}}$ pin state. When the $\overline{\text{SYNC}}$ pin state changes, the external/status interrupt occurs.

(ii) External sync mode

When synchronization has been established by the detection of external synchronization, the last bit of the sync character must be set to "0" at the second $\overline{\text{RxC}}$ falling edge from the rising edge of the received $\overline{\text{RxC}}$. That is, to set the $\overline{\text{SYNC}}$ input to "0" by the external circuit after the detection of synchronization, full 2 receive cycle clocks must be awaited.

When the $\overline{\text{SYNC}}$ input goes "0", the sync hunt bit is set. When synchronization is lost or the end of message is detected, the enter hunt phase bit is set.

(iii) Internal sync mode

In the monosync and bisync modes, bit D4 is initialized to "1" by the enter hunt phase command (D4 of WR3). This bit is reset when the SIO detects the sync character.

(iv) SDLC mode

Bit D4 is set when the receiver is disabled or the enter hunt phase command is issued.

Then, when the frame open flag is detected, this bit is reset.

Bit D5: Clear to send ($\overline{\text{CTS}}$)

This bit indicates the opposite of the $\overline{\text{CTS}}$ pin input state.

Bit D6: Transmit underrun/EOM

This bit is set when the SIO is reset (including channel reset). Only the reset transmitter underrun/EOM latch command WR0 bits D7, D6 = "1", "1") can reset this bit. When the transmit underrun state occurs, the external/status interrupt is generated. Bit D5 is also used to control transmission in the sync or SDLC mode.

Bit D7: Break/abort

In the asynchronous mode in reception, this bit indicates the break state detection. When the break state is detected, this bit is set, generating the external/status interrupt. This bit is reset by the external/status interrupt reset command.

After break, the external/status interrupt is generated again. In the SDLC mode, bit D7 is set when the abort sequence is detected, generating the external/status interrupt.

RR1; Reader register 1

Table 3.21 Configuration of Read Register1

D7	D6	D5	D4	D3	D2	D1	D0
End of frame	framing error	Receiving overrun error	Parity error	Fraction			Feed all characters

100489

Bit D0: All sent

In the asynchronous mdoe, this bit is set when all characters are sent from the transmitter or there is no transmit data in the SIO. In the synchronous mode, this bit is always set.

Bits D1 through D3: Fraction codes

Normally, I field is an integral multiple of character length. If it is not, these bits show the number of fraction bits. These codes are effective only for the transmission for which the end of frame bit is set in the SDLC mode.

Example : Figure 3.12 shows examples of fractions in which the number of bits/ character at the end of I field is 8 bits (1) and 4bits (2).

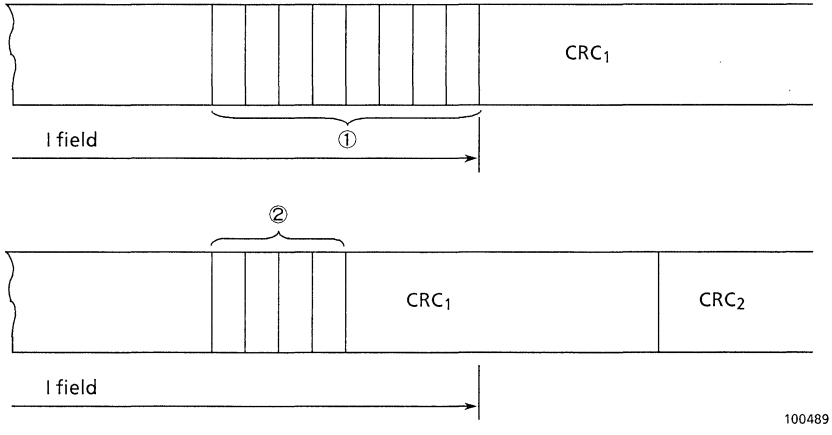


Figure 3.12 Examples of Fraction Bits in I Field

Table 3.22 (a) shows the fraction codes for the receive character whose character length is 8 bits.

Table 3.22 (a) Bit Patterns by Fraction Bits at End of I Field

Number of fraction bits at end of I field		D3	D2	D1
1 byte before	2 bytes before			
0	3	1	0	0
0	4	0	1	0
0	5	1	1	0
0	6	0	0	1
0	7	1	0	1
0	8	0	1	1
1	8	1	1	1
2	8	0	0	0

100489

The same table can also be provided for each character length when the receive character length of I field is other than 8 bits.

Table 3.22 (b) Bit Patterns by Number of Bits/Character (No Fractions)

Bits/character	D3	D2	D1
5 bits/Character	0	0	1
6 bits/Character	0	1	0
7 bits/Character	0	0	0
8 bits/Character	0	1	1

100489

Bit D4: Parity error

This bit is latched when the parity select bit (D0 of WR4) is set and a parity error is detected in the receive data. Latch can be cleared by the error reset command (WR0 bits D5, D4, D3 = "1", "1", "0").

Bit D5: Receive overrun error

The receive data FIFO holds up to 3 characters. When more characters are received without read out by the MPU, the excess character is set to the receive FIFO. When this character is read by the MPU, this receive overrun error is set. Once set, bit D5 latches that state. When the error reset command (command 6 of WR0 bits D3 through D5) is written, this bit is also reset.

Bit D6: CRC/framing error

In the asynchronous mode, this bit is set when a framing error is detected in the received character. Because this bit is not latched, it is always updated.

In the synchronous and SDLC modes, this bit indicates the transmitted CRC check result. This bit is reset when the error reset command (command 6 of WR0 bits D3 through D5) is written.

Bit 7: End of frame

This bit is set when the end flag is detected in the receive data. This bit is reset when the error reset command (command 6 of WR0 bits D3 through D5) is written. This bit is used only in the SDLC mode and is updated when the first character of the next frame is received.

RR2; Read register 2

Table 3.23 Configuration of Read Register 2

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt vector							
V7	V6	V5	V4	V3	V2	V1	V0

Subject to change under different interrupt conditions if the status-affect bit is set.

100489

When the status affect vector bit (D2 of WR1) is set, bits V3 through V1 are changed depending on the interrupt condition at the time. The vector to be read is determined by the interrupt condition having the highest priority at the time of read. When the status affect vector bit is reset, the contents of this register are the same as those of WR2.

3.5.6 Using SIO

The following describes some system examples using the SIO. Figure 3.13 shows an inter-processor communication system. In this example, the MPU on the left side controls the data transfer with the modules on the right side. Both diagrams shown in Figure 3.13 (a) and (b) are communication systems. As shown, the SIO is used to interface with external devices in data communication. The greatest advantage of the SIO is the smaller number of data lines than parallel communication.

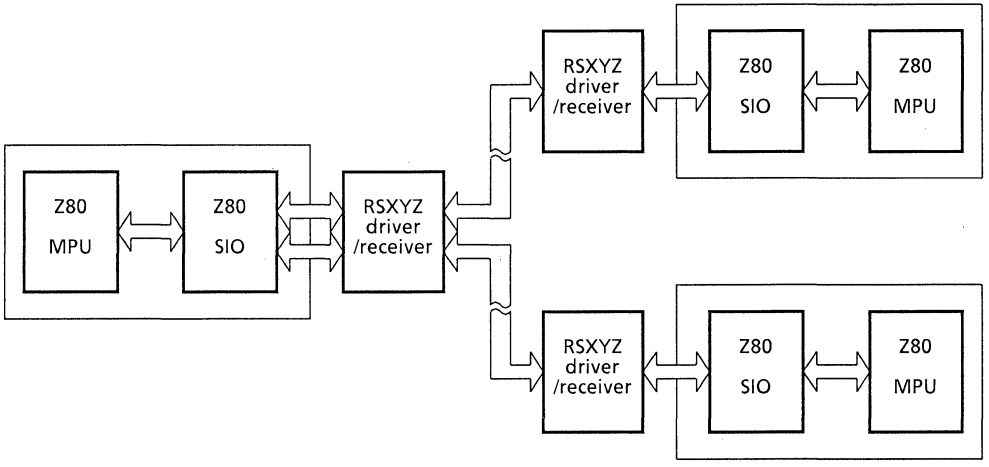


Figure 3.13 (a) Example of Data Communication between Processors

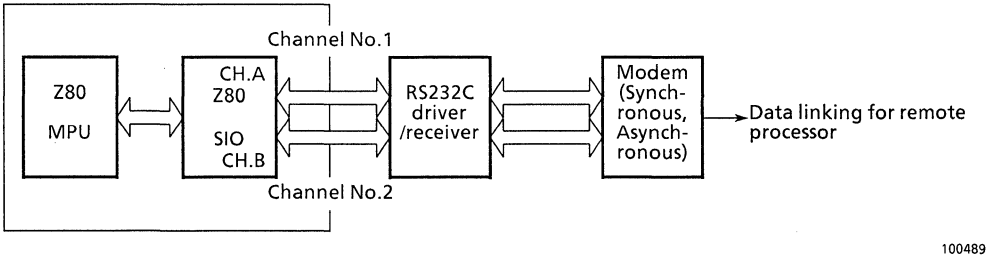


Figure 3.13 (b) Example of Data Communication between Processors

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item		Rating	Unit
V_{CC}	Supply Voltage		0.5 to +7.0	V
V_{IN}	Input Voltage		−0.5 to VCC + 0.5	V
PD	Power Dissipation (6MHz VERSION : TA = 85°C) (8MHz VERSION : TA = 70°C)		250	mW
TSOLDER	Soldering Temperature (10 sec)		260	°C
TSTG	Storage Temperature		−65 to 150	°C
TOPR	Operating Temperature	6MHz VERSION	−40 to 85	°C
		8MHz VERSION	−10 to 70	

050489

4.2 DC ELECTRICAL CHARACTERISTICS

6MHz VERSION : $T_A = -40°C$ to $85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$
8MHz VERSION : $T_A = -10°C$ to $70°C$, $V_{CC} = 5V \pm 5\%$, $V_{SS} = 0V$

SYMBOL	ITEM	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
VILC	Low Clock Input Voltage			−0.3	—	0.6	V
VIHC	High Clock Input Voltage			Vcc − 0.6	—	VCC + 0.3	V
VIL	Input Low Voltage (Except CLK)			−0.5	—	0.8	V
VIH	Input High Voltage (Except CLK)			2.2	—	VCC	V
VOL	Output Low Voltage	IOL = 2.0mA		—	—	0.4	V
VOH1	Output High Voltage (I)	IOH = −1.6mA		2.4	—	—	V
VOH2	Output High Voltage (II)	IOH = −250μA		Vcc − 0.8	—	—	V
ILI	Input Leakage Current	VSS ≦ VIN ≦ VCC		—	—	±10	μA
ILO	Output Leakage Current	VSS + 0.4 ≦ VOUT ≦ VCC		—	—	±10	μA
IL (SY)	$\overline{SYNC}$ Pin Leakage Current	VSS + 0.4 ≦ VIN ≦ VCC		−40	—	10	μA
ICC1	Power Supply Current	VCC = 5V fCLOK = (1) VIHC = VIH = VCC − 0.2V VILC = VIL = 0.2V	(2)	—	4	10	mA
			(3)	—	5	12	
ICC2	Standby Supply Current Except SYNC at "L" output	VCC = 5V VIH = VIHC = VCC − 0.2V VIL = VILC = 0.2V		—	—	10	μA

Note : (1) fCLK = 1/TcC (MIN.), (2) AP-6/AM-6/AF-6/AT-6, (3) AP-8

050489

4.3 AC ELECTRICAL CHARACTERISTICS (1/2)

6MHz VERSION : $T_A = -40°C$ to $85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$
8MHz VERSION : $T_A = -10°C$ to $70°C$, $V_{CC} = 5V \pm 5\%$, $V_{SS} = 0V$

(1/2)

NO.	SYMBOL	PARAMETER	(1) (6MHZ)		(2) (8MHZ)		UNIT
			MIN.	MAX.	MIN.	MAX.	
1	TcC	Clock cycle time	165	DC	125	DC	ns
2	TwCH	Clock pulse width (High)	70	DC	50	DC	ns
3	TfC	Clock fall time	—	15	—	15	ns
4	TrC	Clock rise time	—	15	—	15	ns
5	TwCl	Clock pulse width (Low)	70	DC	50	DC	ns
6	TsCS (C)	Control signal to clock setup time ($\overline{CE}$, C/$\overline{D}$, B/$\overline{A}$)	60	—	60	—	ns
7	TsRD (C)	$\overline{IORQ}\downarrow$, $\overline{RD}\downarrow$ to clock $\uparrow$ setup time	60	—	55	—	ns
8	TdC (DO)	Clock $\uparrow$ to data out delay	—	150	—	100	ns
9	TsDI (C)	Data in to clock $\uparrow$ setup (Write or $\overline{M1}$ cycle)	30	—	30	—	ns
10	TdRD (DOz)	$\overline{RD}\uparrow$ to data out float delay	—	90	—	70	ns
11	TdIO (DOI)	$\overline{IORQ}\downarrow$ to data out delay (INTACK cycle)	—	120	—	85	ns
12	TsM1 (C)	$\overline{M1}$ to clock $\uparrow$ setup time	75	—	50	—	ns
13	TsIEI (IO)	IEI $\downarrow$ to $\overline{IORQ}\downarrow$ setup time (INTACK cycle)	120	—	80	—	ns
14	TdM1 (IEO)	$\overline{M1}\downarrow$ to IEO $\downarrow$ delay (interrupt before $\overline{M1}$)	—	160	—	120	ns
15	TdIEI (IEOr)	IEI $\uparrow$ to IEO $\uparrow$ delay (after ED decode)	—	110	—	110	ns
16	TdIEI (IEOf)	IEI $\downarrow$ to IEO $\downarrow$ delay	—	70	—	70	ns
17	TdC (INT)	Clock $\uparrow$ to $\overline{INT}\downarrow$ delay	—	150	—	100	ns
18	TdIO (W / RWf)	$\overline{IORQ}\downarrow$ or $\overline{CE}\downarrow$ to $\overline{W/RDY}\downarrow$ delay (Wait mode)	—	175	—	130	ns
19	TdC (W / RRf)	Clock $\uparrow$ to $\overline{W/RDY}\downarrow$ delay (Ready mode)	—	100	—	80	ns
20	TdC (W / RWZ)	Clock $\downarrow$ to $\overline{W/RDY}$ float delay (Wait mode)	—	110	—	90	ns
21	Th, Th (CS)	Any unspecified hold when setup is specified	0	—	0	—	ns
22	TwPh	Pulse width (High)	200	—	200	—	ns

050489

AC ELECTRICAL CHARACTERISTICS (2/2)

NO.	SYMBOL	PARAMETER	(1) (6MHZ)		(2) (8MHZ)		UNIT
			MIN.	MAX.	MIN.	MAX.	
23	TwPI	Pulse width (Low)	200	—	200	—	ns
24	TcTxC	$\overline{TxC}$ cycle time	330	∞	250	∞	ns
25	TwTxCl	$\overline{TxC}$ width (Low)	100	∞	80	∞	ns
26	TwTxCh	$\overline{TxC}$ width (High)	100	∞	80	∞	ns
27	TdTxC (TxD)	$\overline{TxC}\downarrow$ to TxD delay (x1 mode)	—	220	—	180	ns
28	TdTxC (W/RRf)	$\overline{TxC}\downarrow$ to $\overline{W/RDY}\downarrow$ delay (Ready mode)	5	9	5	9	CLK Periods*
29	TdTxC (INT)	$\overline{TxC}\downarrow$ to $\overline{INT}\downarrow$ delay	5	9	5	9	CLK Periods*
30	TcRxC	$\overline{RxC}$ cycle time	330	∞	250	∞	ns
31	TwRxCl	$\overline{RxC}$ width (Low)	100	∞	80	∞	ns
32	TwRxCh	$\overline{RxC}$ width (High)	100	∞	80	∞	ns
33	TsRxD (RxC)	RxD to $\overline{RxC}\uparrow$ setup time (x1 mode)	0	—	0	—	ns
34	ThRxD (RxC)	$\overline{RxC}\uparrow$ to RxD hold time (x1 mode)	100	—	80	—	ns
35	TdRxC (W/RRf)	$\overline{RxC}\uparrow$ to $\overline{W/RDY}\downarrow$ delay (Ready mode)	10	13	10	13	CLK Periods*
36	TdRxC (INT)	$\overline{RxC}\uparrow$ to $\overline{INT}\downarrow$ delay	10	13	10	13	CLK Periods*
37	TdRxC (SYNC)	$\overline{RxC}\uparrow$ to $\overline{SYNC}\downarrow$ delay (Output modes)	4	7	4	7	CLK Periods*
38	TsSYNC (RxC)	$\overline{SYNC}\downarrow$ to $\overline{RxC}\uparrow$ setup (External sync modes)	− 100	—	− 100	—	ns

050489

Note : (1) AP-6/AM-6/AF-6/AT-6, (2) AP-8

AC test condition

VIH = 2.4V, VIHC = VCC-0.6V, VIL = 0.4V, VILC = 0.6V

VOH = 2.2V, VOL = 0.8V CL = 100pF

* System Clock

In all modes, the System Clock rate must be at least five times the maximum data rate.

4.4 CAPACITANCE

TA = 25°C

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	Clock Input Capacitance	f = 1MHz All terminals except that to be measured should be earthed.	—	—	7	pF
CIN	Input Capacitance		—	—	5	pF
COUT	Output Capacitance		—	—	10	pF

050489

4.5 TIMING DIAGRAM

Numbers shown in the following figures correspond with those in the 4.3 A.C. Electrical Characteristics Table.

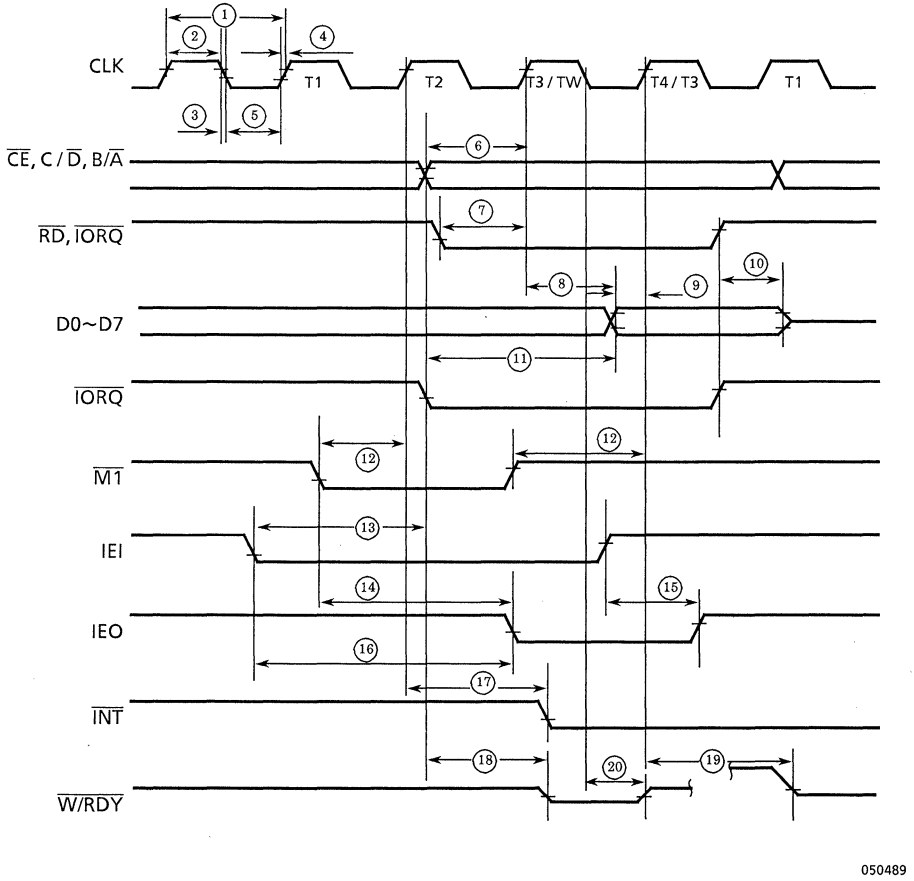


Figure 4.1 (a) Timing Diagram

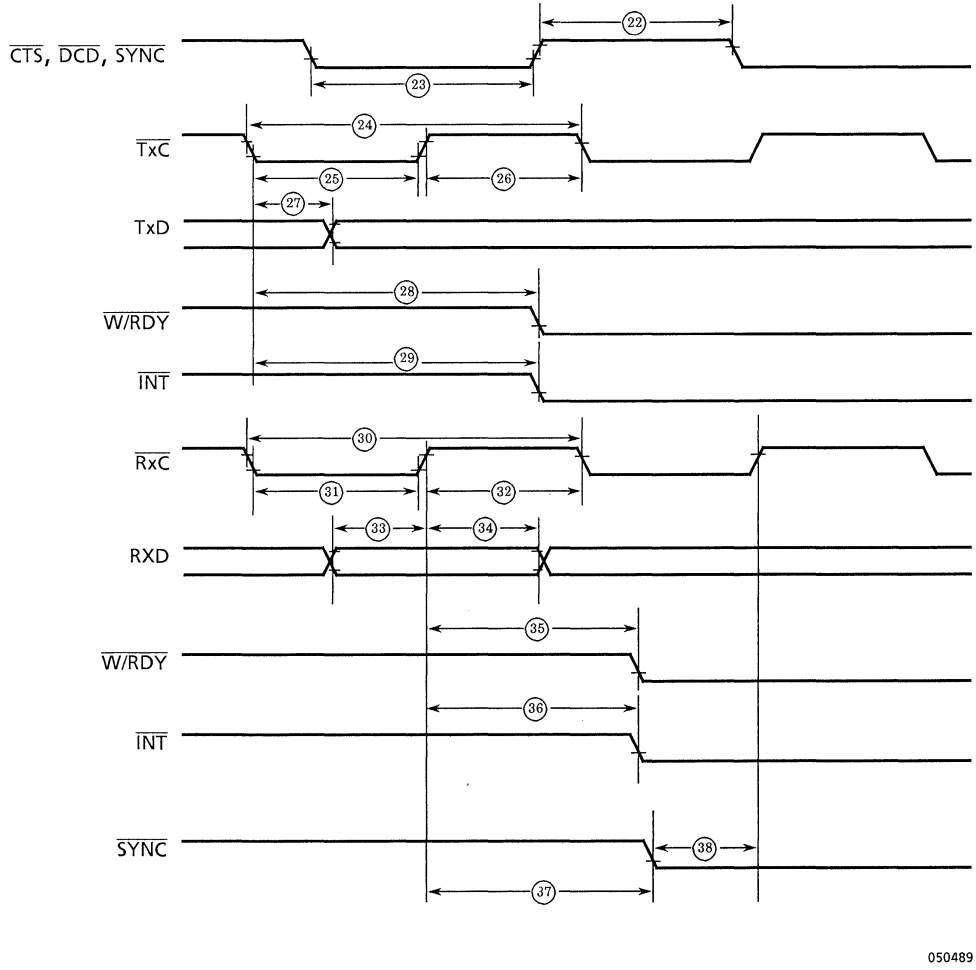


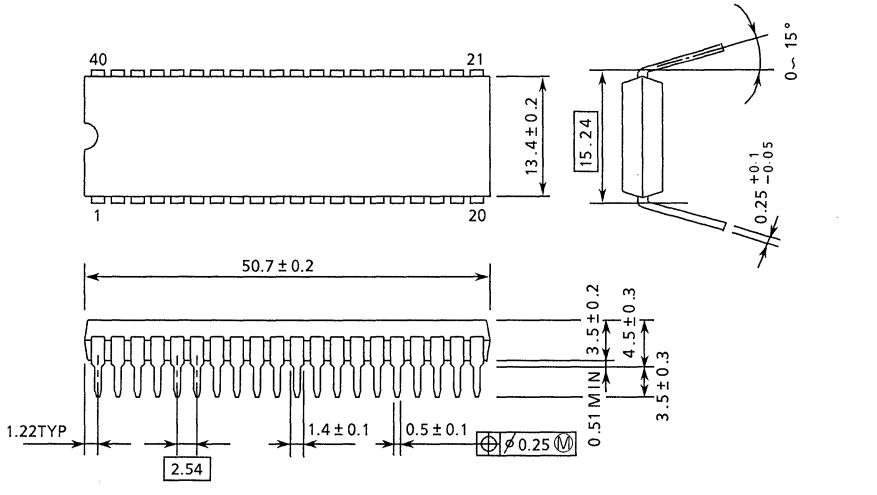
Figure 4.1 (b) Timing Diagram

5. PACKAGE DESCRIPTION

5.1 40-PIN DIP

DIP40-P-600

Unit : mm



270289

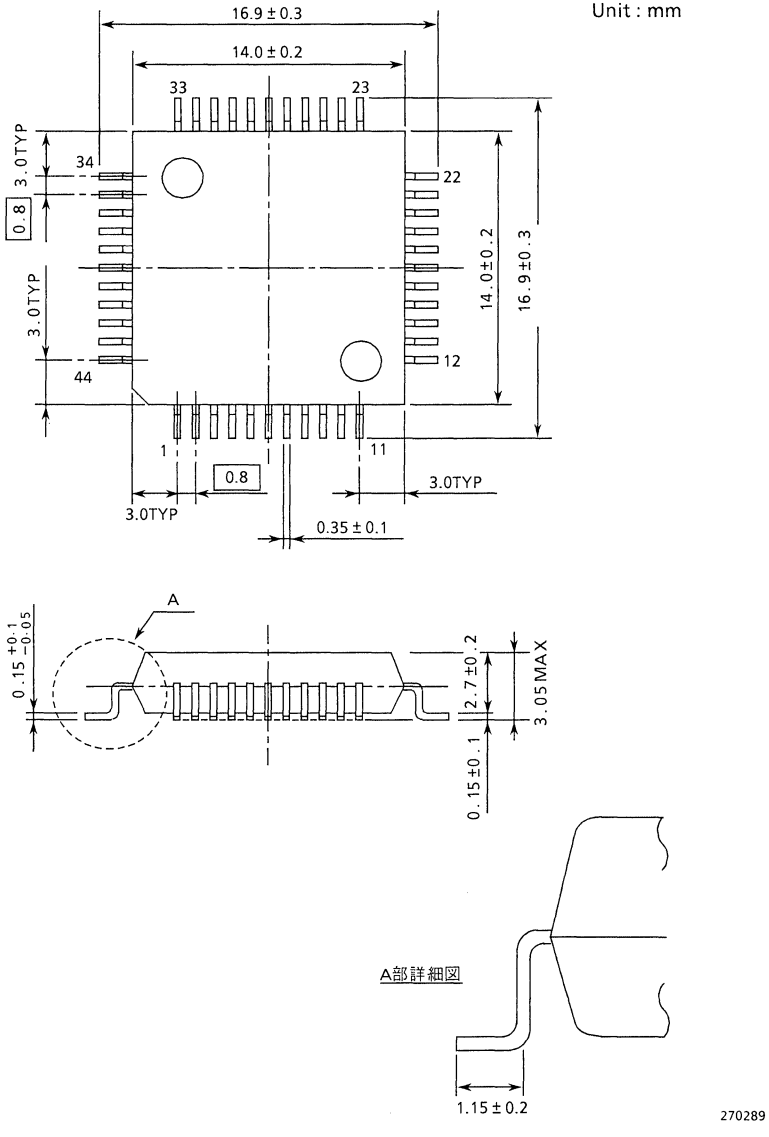
Note 1 : This dimension is measured at the center of bending point of leads.

Note 2 : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.40 leads.

5.2 44-PIN MINI-FLAT PACKAGE

QFP44-P-1414F

Unit : mm

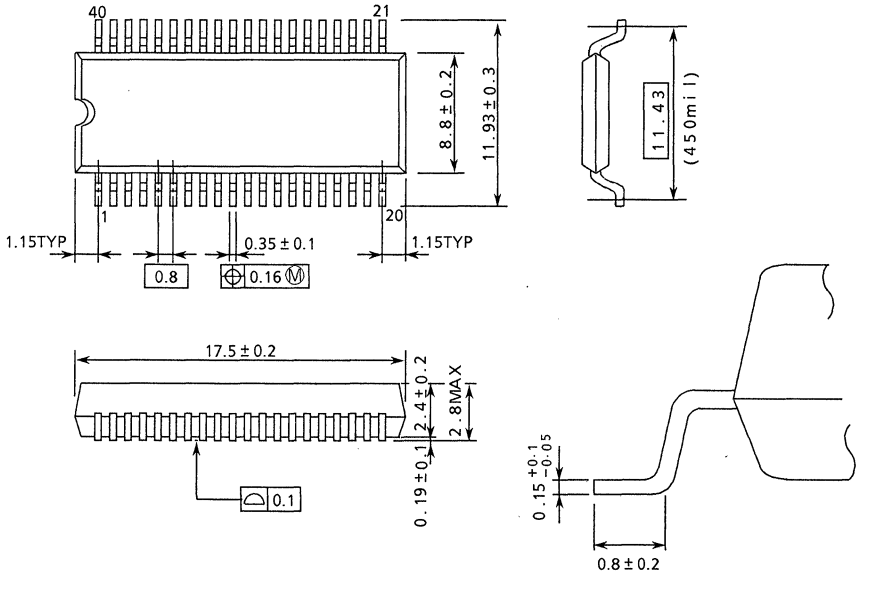


Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

5.3 SOP PACKAGE

SSOP40-P-450

Unit : mm



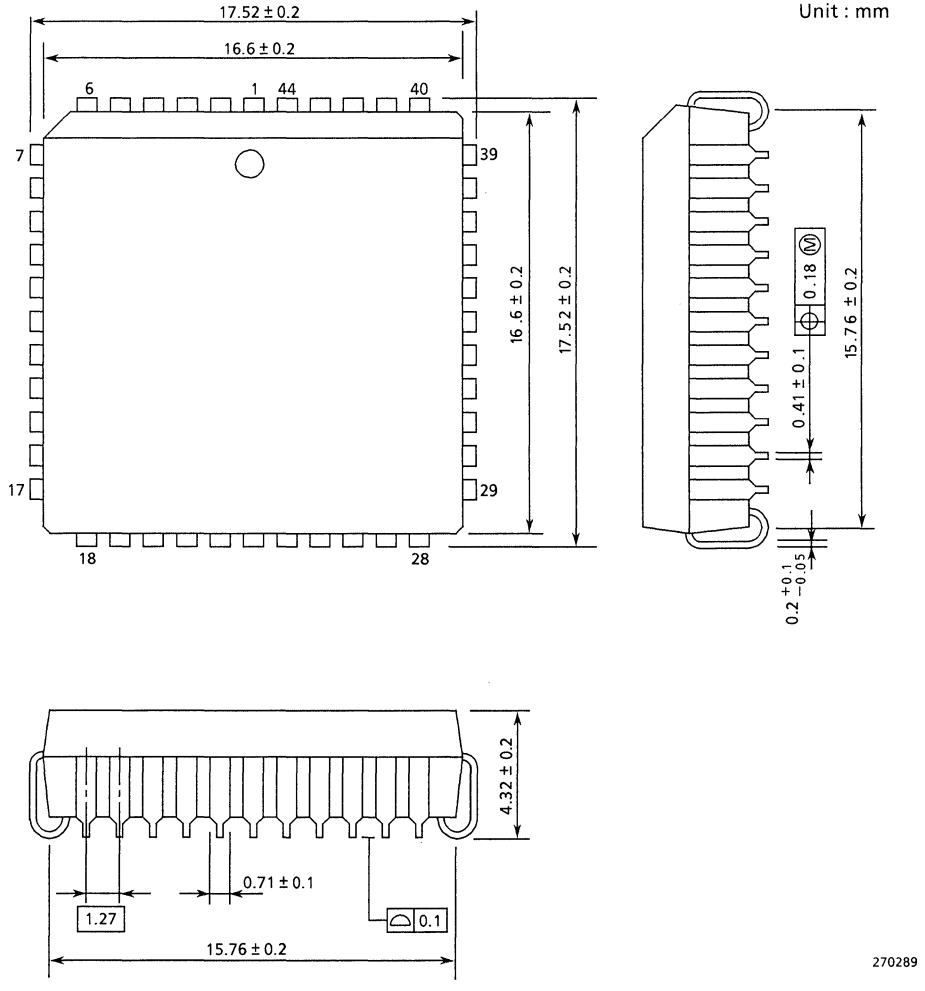
270289

Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

5.4 44-PIN PLCC PACKAGE

QFJ44-P-S650

Unit : mm



6. PRECAUTIONS

In the programming using the SIO, it can be used only for single channel according to registers and bits.

TMPZ84C61AP-6
TLCS-Z80 CGC : Z80 CLOCK GENERATOR/CONTROLLER

1. GENERAL DESCRIPTION AND FEATURES

The TMPZ84C61A is a clock generator/controller (CGC) for the TLCS-Z80 Family (Microprocessor (MPU) and its peripheral LSIs) fabricated with Toshiba's CMOS Silicon Gate Technology. The TMPZ84C61A is provided with two input terminals which are capable of selecting one of the following 3 modes when the TLCS-Z80 MPU executes the HALT instruction.

(1) RUN Mode

The clock (CLK) output operation of the TMPZ84C61A is continued. The TLCS-Z80 MPU is in the HALT state at this time and continues to execute the NOP instruction.

(2) IDLE Mode

The clock (CLK) output by the TMPZ84C61A is stopped. However, only the internal oscillator continues to run.

(3) STOP Mode

The operation of the TMZ84C61A is completely stopped.

In the STOP mode, the operation of the microcomputer system is completely stopped. Therefore, it becomes possible to keep the system at low power consumption.

When "0" is inputted to the $\overline{DIV}$ terminal, the signal obtained by dividing input frequency from the external oscillator is transmitted from the CLK terminal, and when "1" is inputted, the signal of the same frequency, as that of the external oscillator is transmitted from the CLK terminal.

The TMPZ84C61A is enclosed in 16 pin standard DIP.

The principal functions and features of the TMPZ84C61A are as follows.

(1) Upper compatible with TMPZ84C60P

(2) Low power consumption

3mA Typ. (@5V, 6MHz operation)
500μA Typ. (@5V 6MHz) (IDLE Mode)
10μA MAX. (@5V stationary) (STOP Mode)

(3) Single 5V power supply (5V ± 10%)

(4) Extended operating temperature (−40°C to 85°C)

(5) The following 3 modes are selectable :

- RUN Mode
- IDLE Mode
- STOP Mode

(6) Clock output frequency division

(7) With the $\overline{\text{RESOUT}}$ terminal signal, stable reset pulse can be provided to the MPU and its peripheral LSIs.

Compatibility with TMPZ84C60

TMPZ84C61A is used in the same mode as TMPZ84C60 under the following conditions.

(1) Leave the $\overline{\text{RESOUT}}$ terminal open, and connect the $\overline{\text{RESET}}$ terminal of the MPU with that of TMPZ84C61A.

(2) Input "1" to the $\overline{\text{DIV}}$ terminal.

2. PIN CONNECTIONS AND PIN FUNCTIONS

The pin connections and I/O pin names and brief functions of the TMPZ84C61A are shown below.

2.1 PIN CONNECTIONS

The pin connections of the TMPZ84C61A are as shown in Figure 2.1.

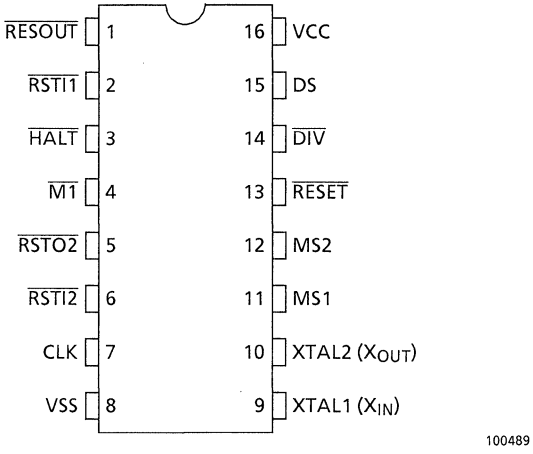


Figure 2.1 Pin Connections

2.2 PIN NAMES AND FUNCTIONS

I/O pin names and functions of the TMPZ84C61A are as shown in Table 2.1.

Table 2.1 Pin Names and Functions

Pin Name	Number of Pin	Input/Output 3-state	Function
$\overline{RSTI1}$	1	Input	Restart signal from clock (CLK) stop state (Level trigger input)
$\overline{HALT}$	1	Input	Halt signal ($\overline{HALT}$) input
$\overline{M1}$	1	Input	Machine cycle 1 ($\overline{M1}$) signal input
$\overline{RSTO2}$	1	Output	Restart signal $\overline{RSTI2}$ output
$\overline{RSTI2}$	1	Input	Restart signal from clock (CLK) stop state (Edge trigger input)
CLK	1	Output	Single-phase clock output. When the HALT instruction in STOP or IDLE Mode is executed, The TMPZ84C61A holds clock output at "0" level.
DS	1	Input	Counter output stage selecting input. Input to set up a warming-up time at time of restart ftom the clock stop state in stop mode.
$\overline{RESET}$	1	Input	Reset signal input. Restart signal from clock (CLK) stop state (Level trigger input)
$\overline{RESOUT}$	1	Output	Reset signal output. Restart signal output to MPU and peripherals.
$\overline{DIV}$	1	Input	Determinds division or non-division of oscillation frequency. Division mode is selected by the input of "0".
MS1, MS2	2	Input	Mode selection input. One of 3 modes (RUN, IDLE,STOP) is selected according to the state of these 2 pins
XTAL1 (X_{IN}), XTAL2 (X_{OUT})	2	Input Output	Crystal oscillator connecting terminal
VCC	1	Power supply	+ 5V
VSS	1	Power supply	0V

100489

3. DESCRIPTION OF OPERATION

The system configuration, functions and basic operation of the TMPZ84C61A Clock Generator are described here.

3.1 BLOCK DIAGRAM

The block diagram of the internal configuration is shown in Figure 3.1.

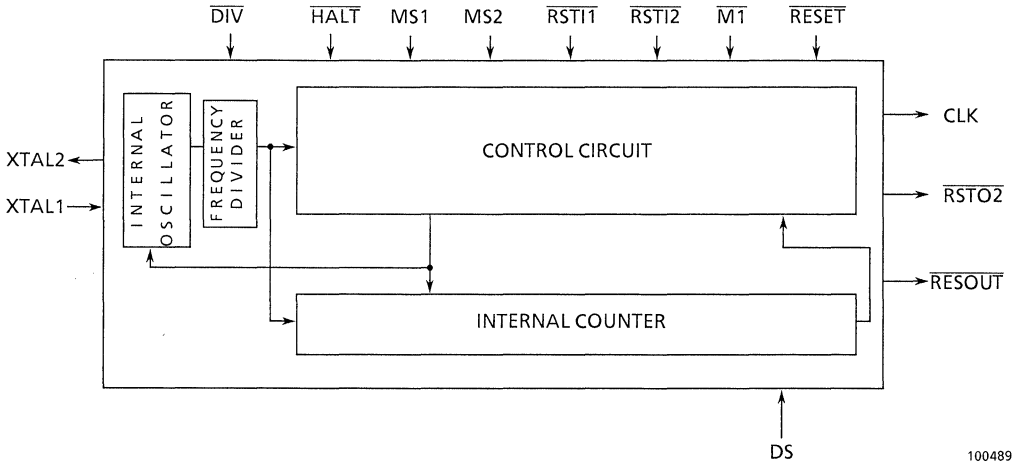


Figure 3.1 Block Diagram

3.2 SYSTEM CONFIGURATION

The internal configuration of TMPZ84C61A is as shown in Figure 3.1.

The waveform that is inputted by the external oscillator is converted into the square-wave for clock by the internal oscillator, Clock is controlled by the control circuit and the counter, and outputted to the outside.

In this section, the following principal components and functions which must be known in using the TMPZ84C61A will be described.

(1) Generation of clock

(2) Operation mode

(3) Warming-up time at restart

3.2.1 Generation of clock

TMPZ84C61A internally supports an oscillation circuit to facilitate the clock output. A desired clock is generated from the CLK terminal by simply connecting the oscillator with the external terminals (XTAL1 and XTAL2). The input of "1" to the $\overline{DIV}$ terminal allows the output of the same frequency as that of the oscillator. By entering "0" to the $\overline{DIV}$ terminal, TMPZ84C61A generates the clock signal with a 1/2 the input frequency. The $\overline{DIV}$ terminal must be kept "Low" for a frequency level of more than 6MHz.

Figure 3.2 shows a typical connection of the oscillator.

(1) A quartz crystal should conform to the follwing characteristics or MR12000-C20 of TokyoDempa (oscillation frequency : 12MHz) :

Output clock frequency (f) : 6MHz

MR12000-C20 quarts crystal oscillator $C_S < 4pF$ $R_S < 25\Omega$

$Cin = Cout = 33pF$

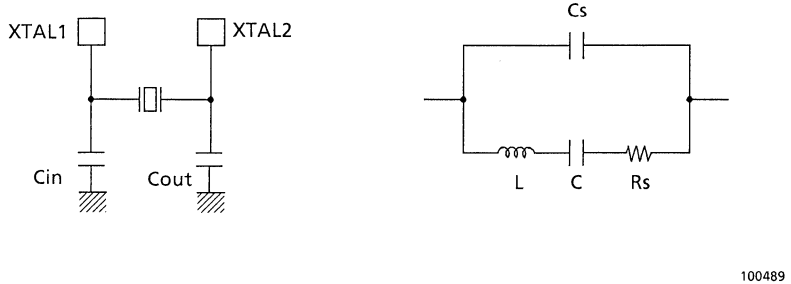


Figure 3.2 Examples of Oscillator Connection

3.2.2 Operation modes

There are 3 kinds of operations modes ; RUN mode, IDLE mode,and STOP mode available for the TMPZ84C61A. One of these modes can be selected by the mode select input (MS1, MS2). These operation modes are effective when the TLCS-Z80 MPU is executing the HALT instruction. When fetching the HALT instruction, MPU outputs "0" to the $\overline{HALT}$ terminal to indicate that MPU enters into the HALT state. After executing the HALT instruction, the TMPZ84C61A performs the operation in one of these mode by this signal.

The operations of these modes in the HALT state are shown in Table 3.1.

Table 3.1 TMPZ84C61A Operation Modes

MS1	MS2	Operation Mode	Content of Operation at Halt State
1	1	RUN Mode	Continues to supply clock to the outside.
0	* Note)	IDLE Mode	The internal oscillator only operates and supply of clock to the outside is stopped. Clock output (CLK) is held at "0" level.
1	0	STOP Mode	The internal operations are all stopped. Clock output (CLK) is held at "0" level.

100489

Note : MS2 can be used for any modes.

Clock is continuously supplied in any mode except the HALT state. MPU is restarted from the clock stop state in IDLE mode and STOP mode by inputting one of $\overline{\text{RSTI1}}$, $\overline{\text{RSTI2}}$ or $\overline{\text{RESET}}$ signal.

MPU is released from the HALT state by the input of $\overline{\text{RESET}}$ signal or acceptance of maskable ($\overline{\text{INT}}$ signal) or non-maskable ($\overline{\text{NMI}}$ signal) interrupt request. Therefore, these signals are normally connected to MPU as follows :

- Connect the $\overline{\text{RESET}}$ signal of the system with TMPZ84C61A.
- Connect the $\overline{\text{RESET}}$ signals of the MPU and peripherals with the $\overline{\text{RESOUT}}$ signal.
- $\overline{\text{INT}}$ signal (maskable interrupt input) and $\overline{\text{RSTI1}}$ signal (restart input) are commonly used.
- $\overline{\text{RETO2}}$ signal (output of restart input signal $\overline{\text{RSTI2}}$) is connected to $\overline{\text{NMI}}$ signal (non-maskable interrupt input).

3.2.3 Warming-up time at restart (STOP mode)

When released from the HALT state by acceptance of interrupt request, MPU will execute the interrupt routine. Therefore, to restart the clock by $\overline{\text{RSTI1}}$ or $\overline{\text{RSTI2}}$ restart signal in STOP mode it is necessary to supply clock to the outside after the oscillation is sufficiently stabilized. The TMPZ84C61A provides a sufficient warming-up time enough to reach stabilized frequency by operating the internal counter. The warming-up is completed and clock output is started at the rising edge of the internal counter output which is divided oscillation frequency. There is the DS input terminal provided for setting this warming-up time, and a time of 2^{17} (DS=0) or 2^{14} (DS=1) division of the externally connected oscillator is provided.

The block diagram of the internal counter unit is shown in Figure 3.3 and the relationship between the logic of the DS terminal and warming-up time is shown in Table 3.2.

Further, in case of the restart by $\overline{\text{RESET}}$ signal, the internal counter does not operate for a quick operation at time of power ON.

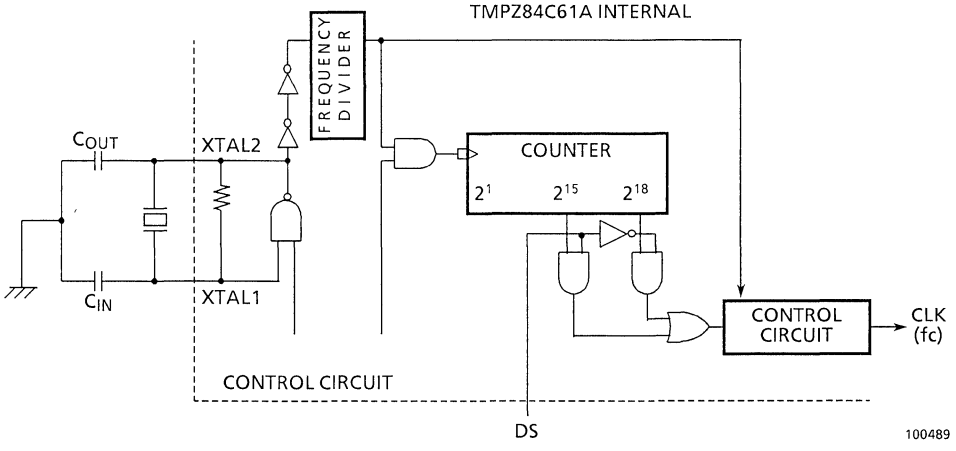


Figure 3.3 Block Diagram of Internal Counter

Table 3.2 Warming-up Time

DS	Counter Output	fc = 6MHz	Unit
0	2^{18}	21.9	ms
1	2^{15}	2.7	

100489

3.3 STATUS CHANGE FLOWCHART AND BASIC TIMING

In this section, the status change and basic timing when the TMPZ84C61A is operating are explained.

3.3.1 Status change flowchart

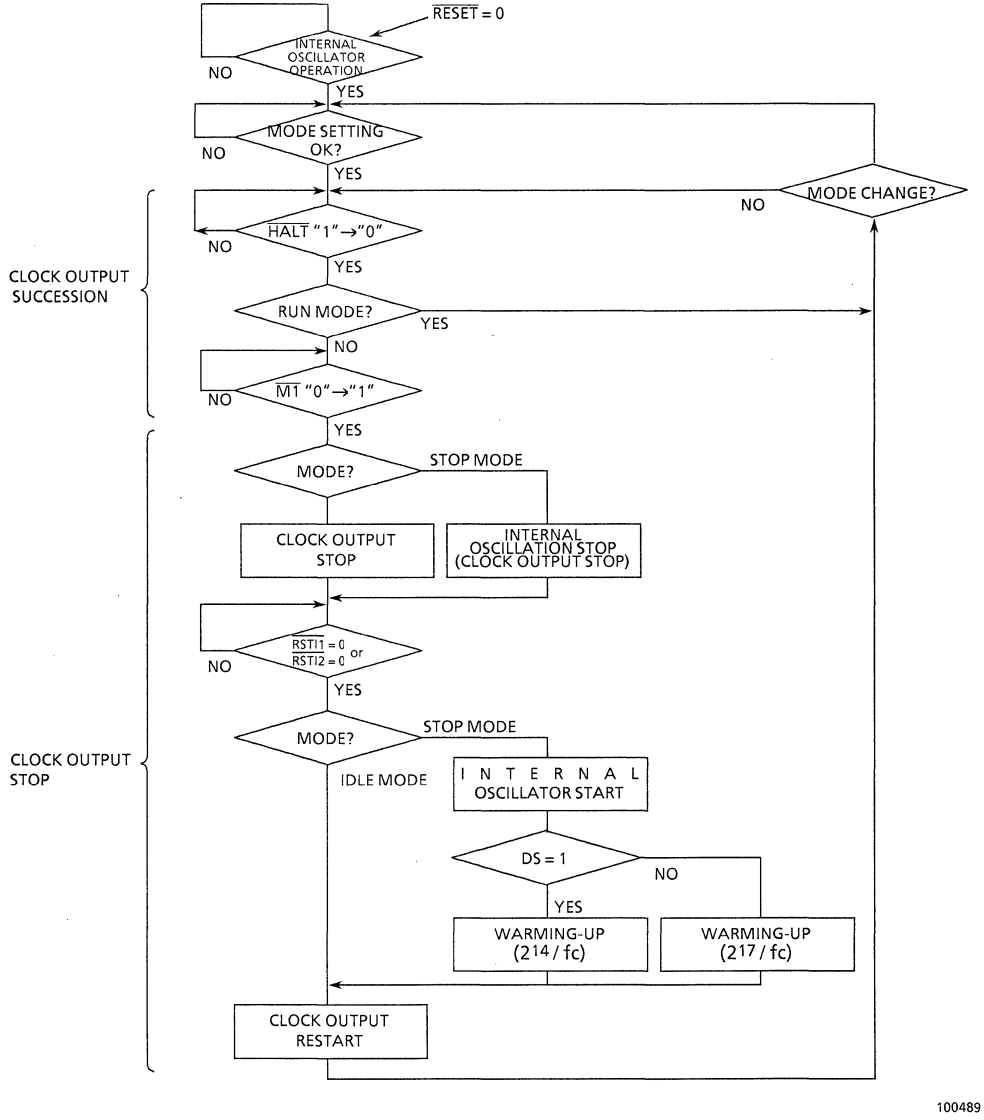


Figure 3.4 Status Change Flowchart

3.3.2 Basic timing

(1) Operation when HALT instruction is executed

The basic timing of each mode when the TLCS-Z80 MPU executes the HALT intruction is explained. Synchronously with the fall of T4 state of the HALT instruction opcode fetch cycle (M1), MPU makes the $\overline{\text{HALT}}$ signal to "0" level. By this signal,the TMPZ84C61A detects that MPU is going to enter into the HALT state.

(a) RUN mode (MS = 1, MS2 = 1)

The basic timing of RUN mode is shown in Figure 3.5.
In the RUN mode, clock is continuously supplied to the outside even when MPU is in the HALT state. This mode is used on a system requiring the memory address refresh.

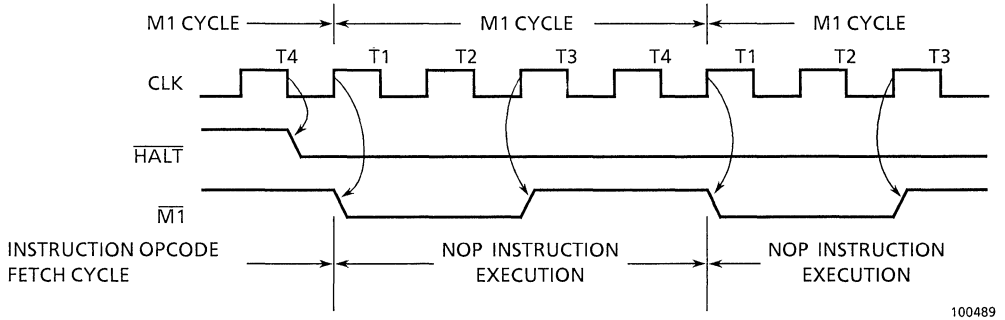


Figure 3.5 RUN Mode Timing

(b) IDLE mode (MS1 = 0) and STOP mode (MS1 = 1, MS2 = 0)

The basic timing in the IDLE and STOP modes is shown in Figure 3.6.
In these modes, clock output is stopped at the "0" level during T4 state by the $\overline{\text{HALT}}$ signal and $\overline{\text{M1}}$ signal following the HALT instruction.
However, in case of the STOP mode the internal oscillator of the TMPZ84C61A is also stopped.

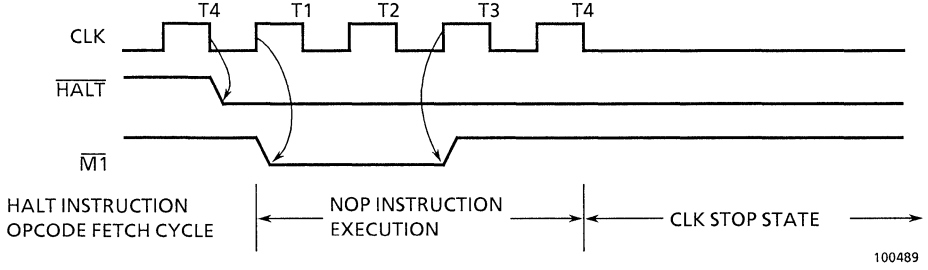


Figure 3.6 IDLE/STOP Mode Timing

(2) Restart from clock stop

The clock is restarted from the stopped state in the IDLE or STOP mode when "0" is inputted to any one of the following signals :

- $\overline{\text{RSTI1}}$ (level trigger input)
- $\overline{\text{RSTI2}}$ (edge trigger input)
- $\overline{\text{RESET}}$ (level trigger input)

(a) Restart in IDLE mode

The restart sequence from the clock output stop state in the IDLE mode is shown in Figure 3.7. In the restart in the IDLE mode, the clock output is restarted in a relatively short delay time as the internal oscillator is in operation even when the clock output is kept stopped.

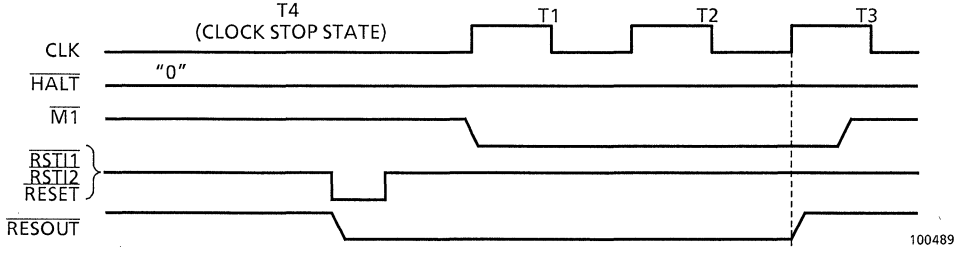


Figure 3.7 Restart Sequence Timing from Clock Stop State (IDLE mode)

(b) Restart in STOP mode

The restart sequence from the clock output stopped state in the STOP mode is shown in Figure 3.8. In restarting the clock output by inputting "0" into $\overline{\text{RSTI1}}$ or $\overline{\text{RSTI2}}$ signal, a warming-up time is automatically provided by the internal counter.

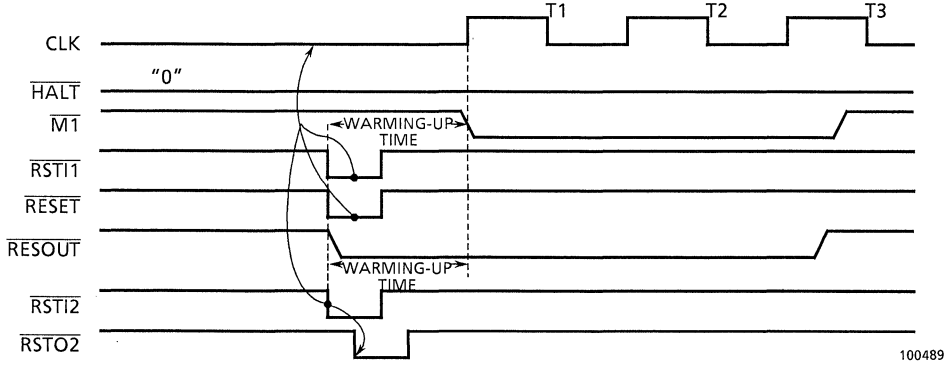


Figure 3.8 Restart Sequence Timing from Clock Stop State (STOP mode)

3.4 METHOD OF USE

Connection of the TMPZ84C61A with MPU when executing the HALT instruction is explained here.

3.4.1 $\overline{\text{RESET}}$ Signal

Figure 3.9 shows the examples of restart timing in the STOP mode when the TLCS-Z80 MPU and TMPZ84C61A use $\overline{\text{RESET}}$ signal commonly.

To reset the TLCS-Z80 MPU, $\overline{\text{RESET}}$ signal must be kept at "0" level for at least 3 clocks. Further, when $\overline{\text{RESET}}$ signal becomes "1" level, MPU is released from the HALT state after the dummy cycle for at least 2T states and executes an instruction starting from address 0000H.

In restarting the clock output in the STOP mode by $\overline{\text{RESET}}$ signal, the internal counter for determining the warming-up time does not operate.

Note that MPU may not be restarted properly due to unstable clock output immediately after the internal oscillator is restarted. Therefore, $\overline{\text{RESET}}$ signal should be kept at "0" level for a sufficient period of time enough to firmly reset MPU by taking stability of crystal oscillation at time of power ON.

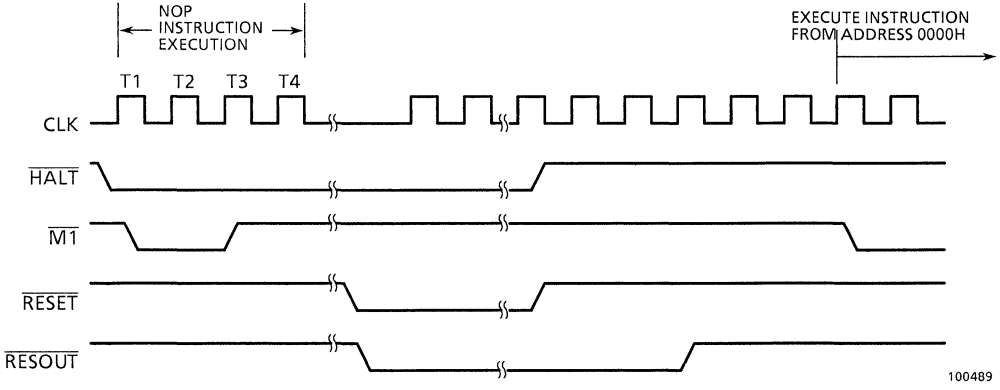


Figure 3.9 Example of Clock Restart Timing by $\overline{\text{RESET}}$ Signal

3.4.2 HALT release by interrupt signal

When the TMPZ84C61A is in the IDLE or STOP mode, the clock output is restarted by $\overline{\text{RSTI1}}$ or $\overline{\text{RSTI2}}$ signal input and MPU starts to run according to that clock input. However, after the clock output, MPU is still in the HALT state and executes the NOP instruction. To release MPU from the HALT state, it is necessary to input the interrupt signal ($\overline{\text{INT}}$ or $\overline{\text{NMI}}$). MPU samples the interrupt signal at the falling edge of the last clock of each instruction (NOP instruction for the HALT state).

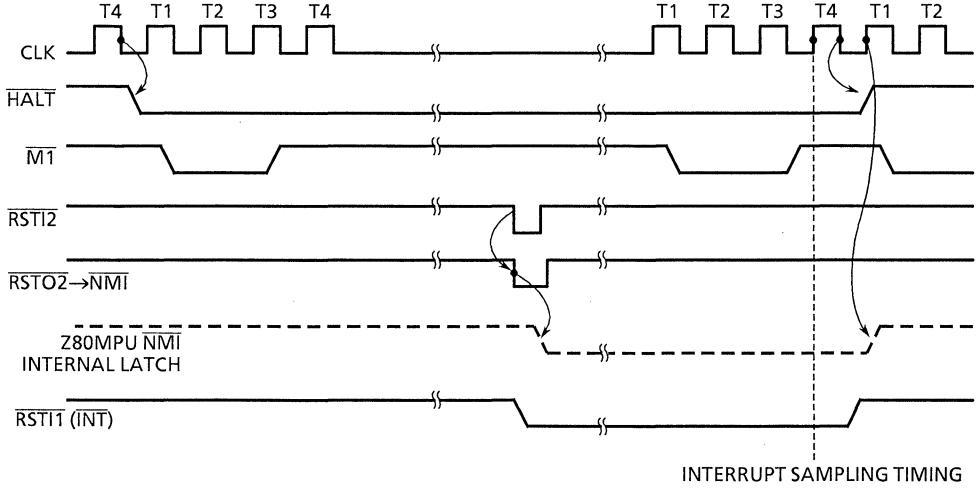
(1) When non-maskable interrupt ($\overline{NMI}$) is used:

The non-maskable interrupt is the edge trigger input, and there is flip-flop (F/F) in MPU. The state of this internal NMI F/F is sampled at the falling edge of the last clock of an instruction. Therefore, if a short low active ("0") pulse has been input before the interrupt signal sampling timing, this interrupt request is accepted.

$\overline{RSTI2}$ input of the TMPZ84C61A is output to $\overline{RSTO2}$ through the internal circuit. It is therefore recommended that the restart signal is generated to the $\overline{RSTI2}$ input terminal and $\overline{RSTO2}$ signal is output into the $\overline{NMI}$ terminal of MPU.

(2) When maskable interrupt ($\overline{INT}$) is used :

For maskable interrupt, the interrupt enable flip-flop (IFF) must be set to "1" before "0" of $\overline{INT}$ input signal is recognized. In the connection of MPU and TMPZ84C61A, this interrupt signal $\overline{INT}$ is jointly used with the restart signal $\overline{RSTI1}$ of the TMPZ84C61A. Shown in Figure 3.10 are examples of the timing when $\overline{RSTO2}$ output signal of the TMPZ84C61A is input to $\overline{NMI}$ of MPU and $\overline{RSTI1}$ signal of the TMPZ84C61A is jointly used with $\overline{INT}$ signal of MPU.



100489

Figure 3.10 Example of Clock Restart Timing by $\overline{RSTI1}$ or $\overline{RSTI2}$ Signals

3.4.3 Example of connection

A connecting example of the TMPZ84C61A and MPU is shown in Figure 3.11.1. This figure shows an example when $\overline{RSTO2}$ output signal of the TMPZ84C61A is input into $\overline{NMI}$ of MPU by jointly using $\overline{RESET}$ signal with MPU and $\overline{RSTI1}$ signal of the TMPZ84C61A and $\overline{INT}$ signal of MPU are jointly used.

Also refer to Figure 3.11.2 for the example of connecting the $\overline{RESOUT}$ signal of TMPZ84C61A with the $\overline{RESET}$ signal of the MPU.

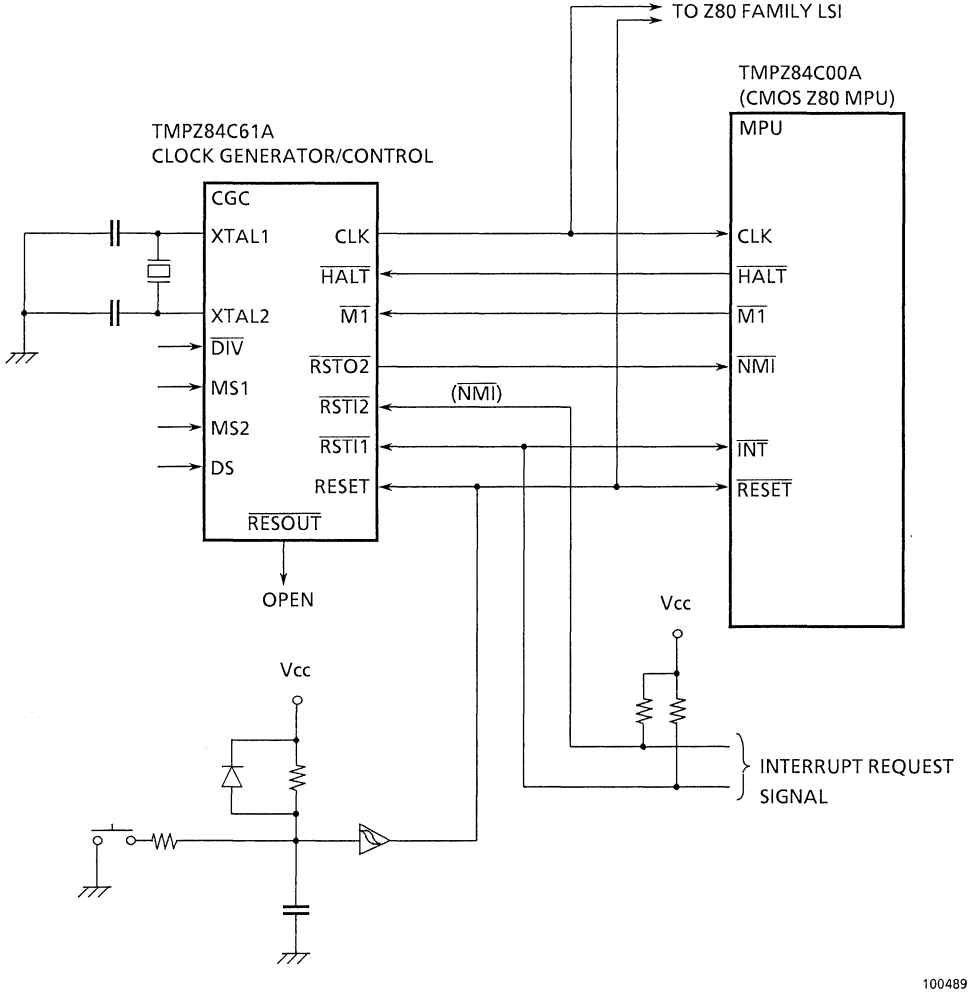


Figure 3.11.1 Connecting Example of the TMPZ84C61A and TLCS-Z80 MPU without using $\overline{\text{RESOUT}}$

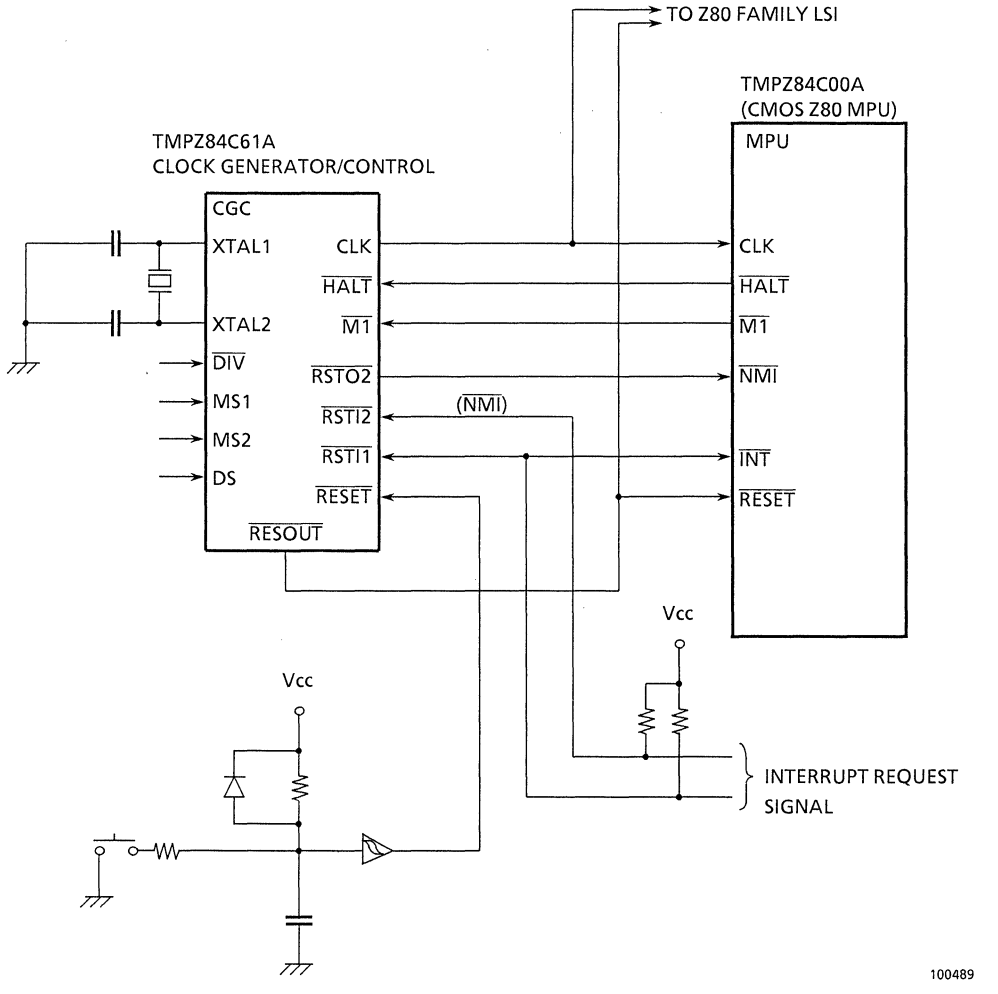


Figure 3.11.2 Conncting Example of the TMPZ84C61A and TLCS-Z80 MPU using $\overline{\text{RESOUT}}$

4. ELECTRICAL CHARACTERISTICS

4.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING	UNIT
V_{CC}	Supply Voltage	− 0.5 to + 7	V
V_{IN}	Input Voltage	− 0.5 to Vcc + 0.5	V
P_D	Power Dissipation (TA = 85°C)	250	mW
T_{SOLDER}	Soldering Temperature (10sec)	260	°C
T_{stg}	Storage Temperature	− 65 to 150	°C
T_{opr}	Operating Temperature	− 40 to 85	°C

100489

4.2 DC ELECTRICAL CHARACTERISTICS

$TOPR = -40°C$ to $85°C$, $VCC = 5V \pm 10\%$, $VSS = 0V$

SYMBOL	ITEM	TEST CONDITIOIN	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage (Except XTAL 1,2)		− 0.5	−	0.8	V
V_{IH}	Input High Voltage (Except XTAL 1,2)		2.2	−	V_{CC}	V
V_{OLC}	Output Low Vlotage (CLK)	I_{OL} = 2.0mA	−	−	0.4	V
V_{OL}	Output Low Vlotage (Except CLK)	I_{OL} = 2.0mA	−	−	0.4	V
V_{OHC}	Output High Vlotage (CLK)	I_{OH} = − 250μA	V_{CC} − 0.6	−	−	V
V_{OH1}	Output High Vlotage (Except CLK)	I_{OH} = − 1.6mA	2.4	−	−	V
V_{OH2}	Output High Vlotage (Except CLK)	I_{OH} = − 250μA	V_{CC} − 0.8	−	−	V
I_{IL}	Input Leak Current	$V_{SS} \leqq V_{IN} \leqq V_{CC}$	−	−	± 1	μA

100489

DIV = "0" (6MHz) (Oscillation frequency: 12MHz)

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
ICC1	Supply Current (Operation) (RUN Mode)	V_{CC} = 5V, fXTAL = 12MHz $V_{IHC} = V_{IH} = V_{CC} - 0.2V$, $V_{ILC} = V_{IL} = 0.2V$	—	3	5	mA
ICC2	Supply Current (STOP Mode)	V_{CC} = 5V, $V_{IHC} = V_{IH} = V_{CC} - 0.2V$, $V_{ILC} = V_{IL} = 0.2V$	—	0.3	10	μA
ICC3	Supply Current (IDLE Mode)	V_{CC} = 5V, fXTAL = 12MHz $V_{IHC} = V_{IH} = V_{CC} - 0.2V$, $V_{ILC} = V_{IL} = 0.2V$	—	0.5	1	mA

100489

4.3 AC ELECTRICAL CHARACTERISTICS

TOPR = −40°C to 85°C, VCC = 5V ± 10%, VSS = 0V, $\overline{DIV}$ = "1", $\overline{DIV}$ = "0"

NO.	SYMBOL	ITEM		TEST CONDI-TION	MIN.	TYP.	MAX.	UNIT
1	TcC	CLK Frequency			165	–	–	ns
2	TwCh	High CLK width			70	–	–	ns
3	TwCl	Low Clk width			70	–	–	ns
4	TrC	CLK rising time			–	–	12	ns
5	TfC	CLK falling time			–	–	12	ns
6	TsHALT (M1r)	$\overline{HALT}$ set-up time			10	–	–	ns
7	TwRSTI1	Low $\overline{RSTI1}$ width			70	–	–	ns
8	TwRSTI2	Low $\overline{RSTI2}$ width			160	–	–	ns
9	TdRSTO2 ($\overline{RSTI2f}$)	$\overline{RSTO2}$ delay time			–	–	60	ns
10	TwRSTO2	Low $\overline{RSTO2}$ width			70	–	–	ns
11	TwRESET	Low $\overline{RESET}$ width		CL = 100pF	70	–	–	ns
12	TRST1S	CLK restart time by $\overline{RSTI1}$ (STOP Mode)	DS = 0		–	(2^{17} + 2.5) TcC	–	ns
			DS = 0		–	(2^{14} + 2.5) TcC	–	ns
13	TRST2S	CLK restart time by $\overline{RSTI2}$ (STOP Mode)	DS = 0		–	(2^{17} + 2.5) TcC	–	ns
			DS = 0		–	(2^{14} + 2.5) TcC	–	ns
14	TRST1I	CLK restart time by $\overline{RSTI1}$ (IDLE mode)			2.5 TcC			ns
15	TRST2I	CLK restart time by $\overline{RSTI2}$ (IDLE mode)			2.5 TcC			ns
16	TRESETI	CLK restart time by $\overline{RESET}$ (IDLE mode)			1 TcC			ns

100489

Note :Test conditions

1) Input high voltage VIH = 2.4V, Input low voltage VIL = 0.4V
2) Testing point
   a VOH = 2.2V, VOL = 0.8V (except CLK output)
   b CLK Output : VOH = VCC − 0.6V, VOL = 0.4V

4.4 CAPACITANCE

TA = 25°C

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
C_{CLOCK}	Clock Input Capacitance (XTAL2)	f = 1MHz All terminals except that to be measured should be earthed.	–	–	15	pF
C_{IN}	Input Capacitance		–	–	5	pF
C_{OUT}	Output Capacitance		–	–	6	pF

100489

4.5 TIMING DIAGRAM

Figure 4.1 to 4.4 show the basic timings of respective operation. Numbers shown in the figures correspond with those in the AC Electrical Characteristics Table in 4.3.

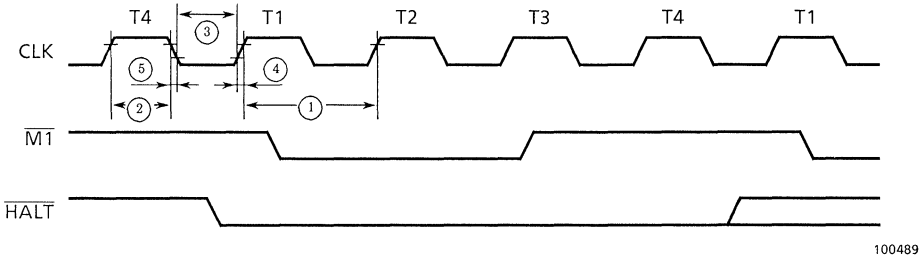


Figure 4.1 Clock Timing

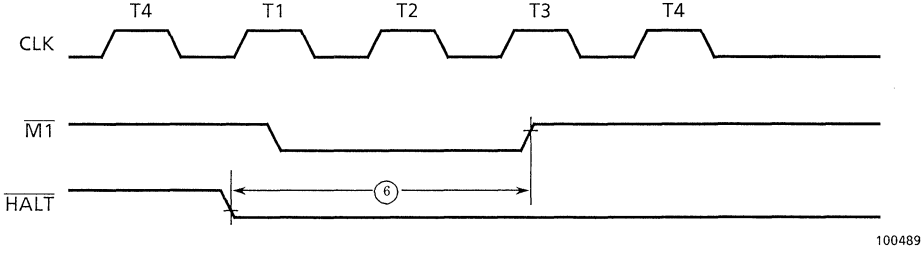


Figure 4.2 Clock Stop Timing (IDLE/STOP Mode)

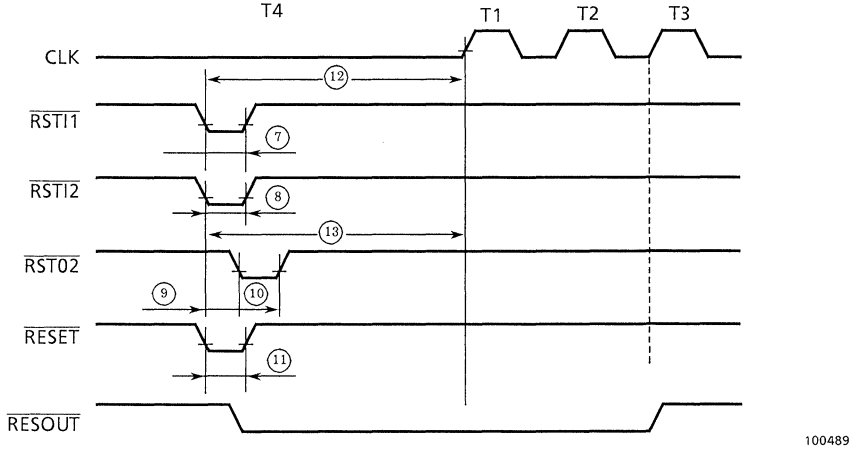


Figure 4.3 Clock Restart Timing (STOP Mode)

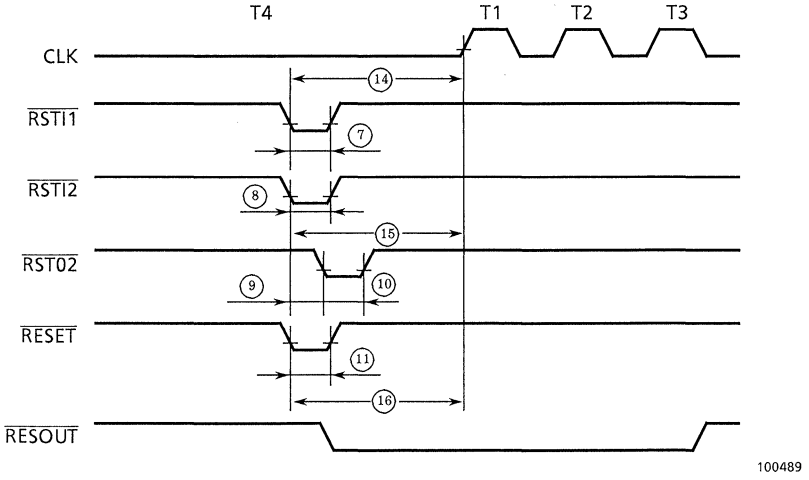
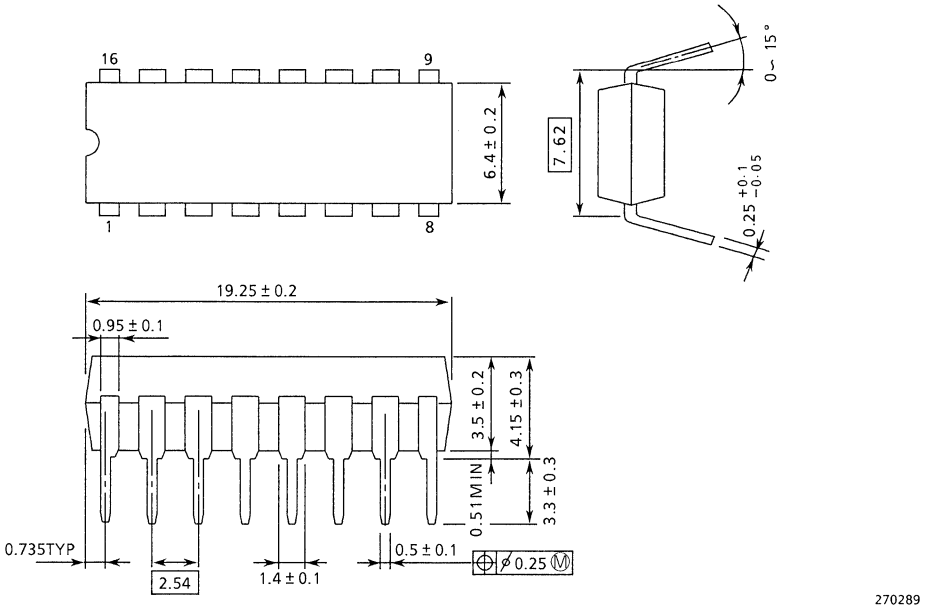


Figure 4.4 Clock Restart Timing (IDLE Mode)

5. OUTLINE DRAWING

DIP16-P-300A

Unit : mm



270289

Note :

1. This dimension is measured at the center of bending point of leads.
2. Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.16 leads.

6. PRECAUTIONS

When the TMPZ84C61A is used, care should be taken to the following points.

(1) In using $\overline{\text{RESET}}$ signal commonly with MPU, hold $\overline{\text{RESET}}$ signal at "0" for a sufficient period of time enough to positively reset MPU. Especially, in case of the restart in the STOP mode using $\overline{\text{RESET}}$ signal, be careful as output of the internal oscillator is not stable.

(2) MPU is not released from the HALT state simply when the clock input is resumed. To release MPU, it is necessary to reset MPU or accept an interrupt request.

(3) Since $\overline{\text{RSTI2}}$ signal of the TMPZ84C61A and $\overline{\text{NMI}}$ signal of MPU are both trailing edge trigger inputs. if both signals are jointly used, $\overline{\text{RSTI2}}$ signal only may be detected and $\overline{\text{NMI}}$ signal may not be detected in some cares. This trouble can be solved by using $\overline{\text{RSTO2}}$ signal, which is the output signal of $\overline{\text{RSTI2}}$ input signal, as the input signal to $\overline{\text{NMI}}$ in order to a sufficient pulse for the detection.

TOSHIBA

PART 1 8-BIT MICROPROCESSOR

CHAPTER 2 TLCS-Z80 ASSP FAMILY

TOSHIBA CORPORATION

CONTENTS

TMPZ84C011A

TMPZ84C013A

TMPZ84C015A

The Guide Of TOSHIBA TLCS-Z80 ASSP Development Tools And Development Methods

TMPZ84C011AF-6
TLCS-Z80 MICROPROCESSOR

1. OVERVIEW AND FEATURES

The TMPZ84C011A is a high-performance CMOS 8-bit microprocessor that contains the functional peripherals such as counter timer circuit (CTC) parallel I/O port, and clock generator/controller (CGC) around the TLCS-Z80 MPU. This microprocessor directly inherits the basic architecture of the TLCS-Z80 MPU, making available the software resources and development tools accumulated so far.

The TMPZ84C011A is based on the new CMOS process and housed in a standard 100-pin mini-flat package, greatly contributing to system minituarization and power saving.

The TMPZ84C011A has many I/O ports, making it available for a wide range of system applications such as the controller, the measuring instrument, and the word processor.

Features:

- Built-in MPU, CTC, and CGC functions of the TLCS-Z80 family.
- High speed operation (6 MHz operation)
- Built-in clock generator (CGC. Clock Generator/Controller)
- Built-in standby capability (with the controller built in)
- Low power consumption
  operation ····· 15 mA typ. (4 MHz) 22mA typ. (6MHz)
  idle ····· 1.0 mA typ. (4 MHz) 1.5mA typ. (6MHz)
  standby····· 500 nA typ.
- Wide operational power voltage range (5V ± 10%, 3 to 6V operation supported)
- Wide operational temperature range (−40°C to +85°C)
- Three selectable operational modes
  Run mode ····· Normal operation
  Idle mode ····· Only clock generation goes on
  Stop mode ····· Clock generation stops; standby state
- Many general-purpose ports (5 ports, 40 bits) which support I/O operation on a bit basis.
- Four channels of timer/counter (the same capability as the TLCS-Z80 CTC).
- Built-in dynamic RAM refresh controller
- Daisy-chain interrupt control pin

- Housed in a compact 100-pin mini-flat package
- Emulation by Z80 ICE (ex. Toshiba Real Time Emulator RTE80) is available.
- The Toshiba evaluator board (BM8024) and adapter board (BM8026) are available.

note : Z80 is a trade mark of Zilog Inc.

2. PIN ASSIGNMENTS AND FUNCTIONS

2.1 PIN ASSIGNMENTS (TOP VIEW)

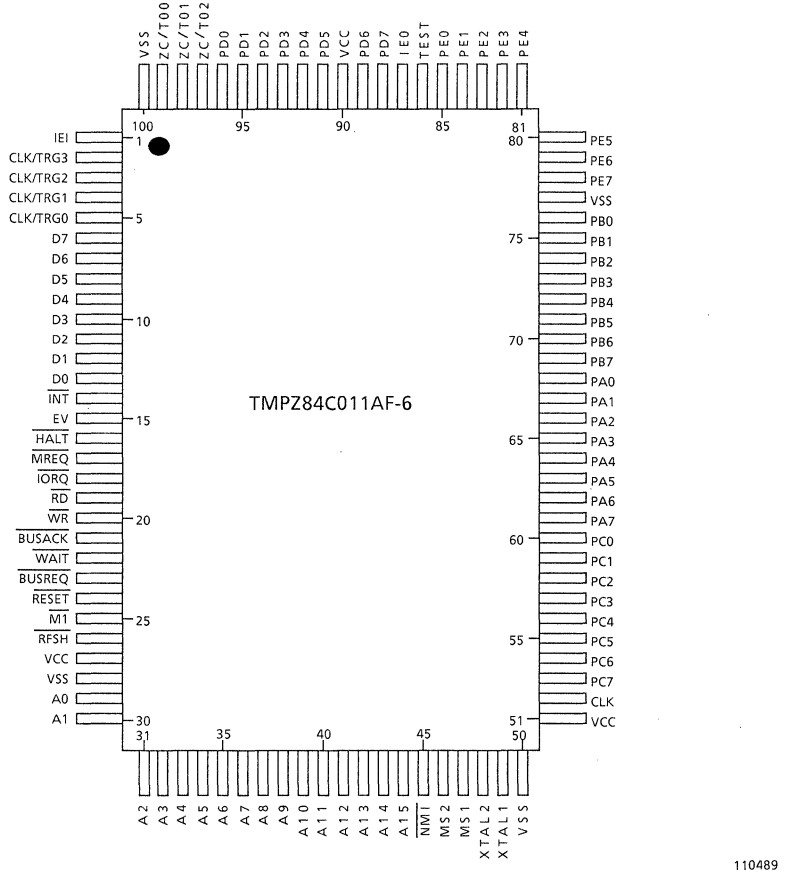


Figure 2.1.1 Pin Assignments

(1/3)

Pin	Number	Type	Function
D0-D7	8 (6-13)	I/O 3-state	The 8-bit bi-directional data bus.
A0-A15	16 (29-44)	Output 3-state	The 16-bit address bus. These pins specify memory and I/O port addresses. During a refresh cycle, the refresh address is output to the low-order 7 bits.
PA0-PA7	8 (61-68)	I/O 3-state	The general-purpose I/O port (port A) which can be set by program for input or output on a bit basis. When it is set for output, it provides latch output.
PB0-PB7	8 (69-76)	I/O 3-state	The general-purpose I/O port (port B) which can be set by program for input or output on a bit basis. When it is set for output, it provides latch output.
PC0-PC7	8 (53-60)	I/O 3-state	The general-purpose I/O port (port C) which can be set by program for input or output on a bit basis. When it is set for output, it provides latch output.
PD0-PD7	8 (88-89) (91-96)	I/O 3-state	The general-purpose I/O port (port D) which can be set by program for input or output on a bit basis. When it is set for output, it provides latch output.
PE0-PE7	8 (78-85)	I/O 3-state	The general-purpose I/O port (port E) which can be set by program for input or output on a bit basis. When it is set for output, it provides latch output.
$\overline{M1}$	1 (25)	Output 3-state	Machine cycle 1 signal pin. In an Op code fetch cycle, this pin goes "0" with the $\overline{MREQ}$ signal. At the execution of a 2-byte Op code, this pin goes "0" for each operation fetch. In a maskable interrupt acknowledge cycle, this pin goes "0" with the $\overline{IORQ}$ signal. When the EV signal is applied, this pin is put in the high-impedance state.
$\overline{RD}$	1 (19)	Output 3-state	The read signal pin. This signal indicates that the MPU is ready for accepting data from memory or an I/O device. The data from the addressed memory or I/O device is gated by this signal onto the MPU data bus. When the $\overline{BUSREQ}$ signal is applied, this pin is put in the high impedance state.
$\overline{WR}$	1 (20)	Output 3-state	The write signal pin. This signal active when the data to be stored in the addressed memory or I/O device is on the data bus. When the $\overline{BUSREQ}$ signal is applied, this pin is put in the high-impedance state.
$\overline{MREQ}$	1 (17)	Output 3-state	The memory request signal pin. When the execution address for memory access is on the address bus, this pin goes "0". During a memory refresh cycle, this pin also goes "0" with the $\overline{RFSH}$ signal.

110489

(2/3)

Pin	Number	Type	Function
$\overline{IORQ}$	1 (18)	Output 3-state	The input/output request signal pin. This pin goes "0" when the address for an I/O operation is on the low-order 8 bits (A0 through A7) for the address bus. The $\overline{IORQ}$ signal is also output with the $\overline{M1}$ signal at interrupt acknowledge to tell an I/O device that it can put the interrupt response vector onto the data bus.
$\overline{WAIT}$	1 (22)	Input	The wait request signal pin. This signal indicates to MPU that the addressed memory or I/O device is not ready for data transfer. As long as this signal is "0", the MPU continues to be in the wait state.
$\overline{BUSREQ}$	1 (23)	Input	The bus request signal pin. The $\overline{BUSREQ}$ signal forces the MPU address bus, data bus, and control signals $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ to get in the high-impedance state. This signal is normally wire-ORed and required an external pullup resistor for these applications.
$\overline{BUSACK}$	1 (21)	Output	The bus acknowledge signal pin. In response to the $\overline{BUSREQ}$ signal, the $\overline{BUSACK}$ signal indicates to the requesting peripheral LSI that the MPU address bus, data bus, and control signals $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ have been put in the high- impedance state.
$\overline{HALT}$	1 (16)	Output 3-state	The halt signal pin. This pin goes "0" when the MPU has executed a HALT instruction and is in the halt state. It is put in the high-impedance state when the EV signals is applied.
$\overline{RFSH}$	1 (26)	Output	The refresh signal pin. When the dynamic memory refresh address is on the low-order 7 bits of the address bus, this signal goes "0". At the same time, the $\overline{MREQ}$ signal also goes active ("0").
EV	1 (15)	Input	The evaluator signal pin. When "1" is applied to this pin, the $\overline{M1}$ and $\overline{HALT}$ pins are put in the high-impedance state. In using the TMPZ84C011A for an evaluator chip, giving the signal with $\overline{BUSREQ}$ electrically disconnects the MPU section and the chip operates following the instructions from other MPUs (such as the MPU of the ICE).
TEST	1 (86)	Input	The test signal pin. Normally, this pin should be "0".
CLK/TRG0 ~ CLK/TRG3	4 (2-5)	Input	The external clock/timer trigger input pins. These 4 CLK/TRG pins correspond to 4 channels. In the counter mode, the down-counter is decremented by 1 and, in the timer mode, the timer is activated by each active edge (a rising of falling edge) which are input at these pins. Whether the active edge is a rising or falling edge can be selected by program.
ZCT/O0 ~ ZCT/02	3 (97-99)	Output	The zero count/time out signal output pin. In either the timer mode or counter mode, pulses are output from these pins when the counter value has reached zero.

110489

(3/3)

Pin	Number	Type	Function
IEI	1 (1)	Input	The CTC interrupt enable input signal pin. This signal indicates whether there is an interrupt by an upper peripheral LSIs in daisy-chaining.
IEO	1 (87)	Output	The CTC interrupt enable output signal pin. This signal controls the interrupts by lower peripheral LSIs in daisy chaining. This signal goes high only when the IEI pin is high and the MPU is not servicing the interrupt from the channel in the CTC.
XTAL1 XTAL2	2 (49) (48)	Input Output	The crystal oscillator connection pins. Connect a resonator whose oscillation frequency is double the system clock (CLK output) frequency.
MS1 MS2	2 (47) (46)	Input	The mode select input pins. The states of these 2 pins determine one of the 3 modes (Run, Idle, and Stop).
CLK	1 (52)	Output	The single-phase clock output pin. When the HALT instruction is executed in the Stop or Idle mode, the MPU and the CTC stop operating with the clock (CLK) output held at "0". This output is used for the clock to other peripheral LSIs
$\overline{\text{RESET}}$	1 (24)	Input	The reset signal input pin. This signal is commonly given to the MPU, CTC, I/O and CGC. It is also used for the restart signal from the halt state.
$\overline{\text{INT}}$	1 (14)	Input	The maskable interrupt request signal pin. The interrupt is caused by the CTC or peripheral LSIs. The interrupt is accepted when the interrupt enable flip-flop (IFF) is set to "1" by software. The $\overline{\text{INT}}$ pin is normally wire-ORed, requiring to attach a pullup resistor externally. This signal is also used for the restart signal from the halt state.
$\overline{\text{NMI}}$	1 (45)	Input	The non-maskable interrupt request signal pin. This interrupt request has a higher priority than a maskable interrupt and independent of the interrupt enable flip-flop (IFF) state.
VCC	3 (27, 51, 90)		The power supply pin (+ 5 V).
VSS	4 (28, 50, 77, 100)		The ground pin (0 V).

110489

3. OPERATIONAL DESCRIPTIONS

3.1 ENTIRE BLOCK DIAGRAM AND OPERATION OF EACH BLOCK

3.1.1 Entire Block Diagram

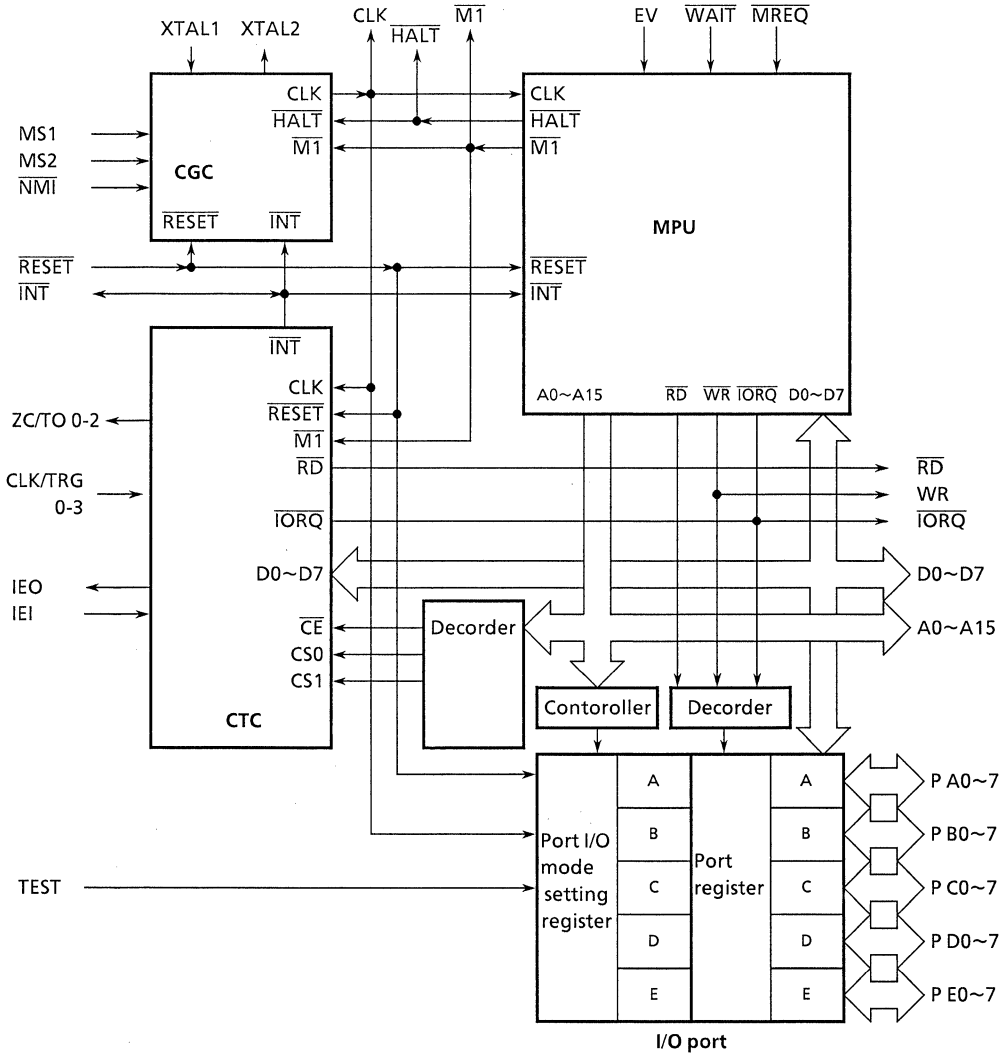


Figure 3.1.1 Entire Block Diagram

3.1.2 Operation of Each Block

The TMPZ84C011A largely consists of a processor (MPU), a counter/timer circuit (CTC), an input/output port section (I/O), and a clock generator/controller (CGC).

- The MPU has all pins of the Toshiba TLCS-Z80 MPU, and its function is fully compatible with TLCS-Z80 microprocessor.
- The CTC provides the capabilities of the Toshiba TLCS-Z80 CTC (TMPZ84C30A) and has the pins required to perform the necessary operations as a TLCS-Z80 peripheral LSI. The four independent timer channels are I/O-addressed internally. It also has the IEI and IEO pins required for the daisy-chain interrupt to provide daisy-chaining with other peripheral LSIs.
- The I/O consists of 5 general-purpose I/O ports, each made up of 8 bits. Each port can be specified by program for input or output on a bit basis. The 5 ports are assigned with I/O addresses, which are specified by program to make the ports perform I/O operations. When these ports are used for output ports, the output is latched, so that the data once output is held as it is until changed.
- The CGC provided the 3 kinds of operation modes to control the entire TMPZ84C011A chip; the Run, Idle, and Stop modes. The CGC provides the same function as the Toshiba TLCS-Z80 CGC (TMPZ84C60) and has the pin to select the Run, Idle, or Stop mode.
  - In the Run mode, the clock generated by the CGC is supplied to the TMPZ84C011A and peripheral LSIs to perform the normal programmed microcomputer operations.
  - In the Idle mode, clock oscillation is going on but the clock is not supplied to the TMPZ84C011A and peripheral LSIs, thereby saving the system power consumption. This mode shortens the time required for system restart.
  - In the Stop mode, clock oscillation is not performed and the system operation can be stopped completely. In this mode, the system can be restarted with the internal data retained with an extremely low power consumption level unique to the CMOS technology.Note that these modes can be set only when the MPU has executed a HALT instruction.
- Additionally, the TMPZ84C011A has also the EV pin which is used to put the MPU in the high-impedance state for electrical disconnection, thus providing an evaluator chip. That is, the MPU in the TMPZ84C011A is electrically disconnected by the EV pin and the $\overline{\text{BUSREQ}}$ pin to implement the emulation by the signal from the in-circuit emulator (ICE).

3.2 MPU OPERATIONS

This subsection describes the system configuration, functions and the basic operations of the MPU of the TMPZ84C011A.

3.2.1 Block Diagram

Figure 3.2.1 shows the block diagram of the MPU.

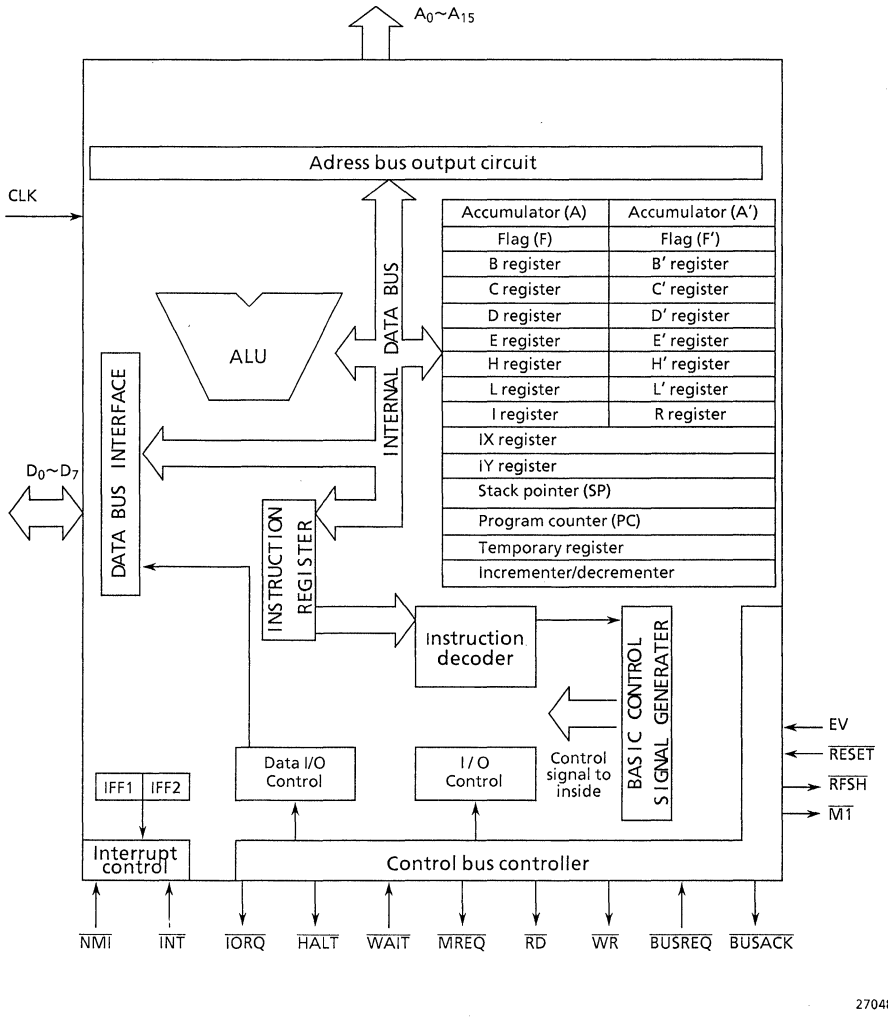


Figure 3.2.1 MPU Block Diagram

3.2.2 MPU System Configuration

The MPU has the configuration shown in Fig. 3.2.1. The address signal is put on the address bus via the address buffer. The data bus is controlled for input or output by the data bus interface. Both the address and data buses are put in the high- impedance state by the $\overline{\text{BUSREQ}}$ signal input to make them available for other peripheral LSIs. The Opcode read from memory via the data bus is written to the instruction register. This Opcode is decoded by the instruction decoder. According to the result of the decoding, control signals are sent to the relevant devices. Receiving these control signals, the ALU performs various arithmetic operations. The register array temporarily holds the information required to perform a given operation.

The following describes the MPU's internal registers and functions:

[1] Internal Register Groups

The configuration of the internal register groups is as follows:

(1) Main registers

A, F, B, C, D, E, H, L

(2) Alternate registers

A', F', B', C', D', E', H', L'

(3) Special registers

I, R, IX, IY, SP, PC

Figure 3.2.3 shows the configuration of the internal register groups. The register groups, each being of a static RAM, consists of eighteen 8 bit registers and four 16-bit registers. The following describes the function of each register:

(1) Main registers (A, F, B, C, D, E, H, L)

(a) Accumulator (A)

The accumulator is an 8-bit register used for arithmetic and data transfer operations.

(b) Flag register (F) (see Figure 3.2.2)

The flag register is an 8-bit register to hold the result of each arithmetic operation. Actually, the 6 of the 8 bits are set ("1")/reset ("0") according to the condition specified by an instruction.

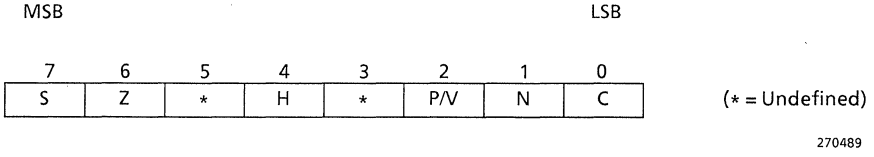


Figure 3.2.2 Flag Register Configration

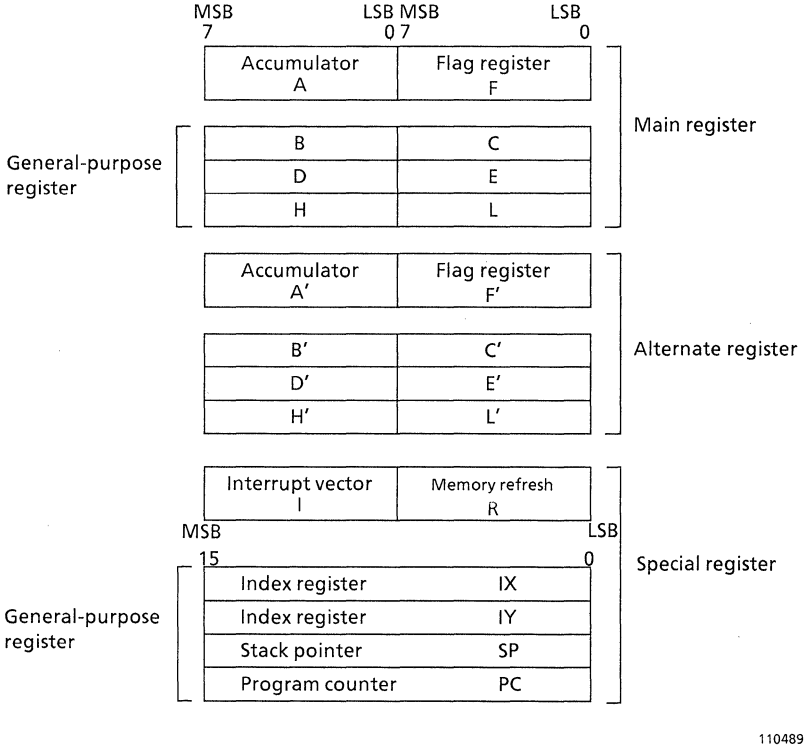


Figure 3.2.3 Flag Register Configuration

The following 4 bits are directly available to the programmer for setting the jump, call and return instruction conditions:

- Sign flag (S)

  When the result of an operation is negative, the S flag is set to "1". Actually, the content of bit 7 of accumulator is stored in this flag.

- Zero flag (Z)

  When all bits turn out to be "0" after operation, the Z flag is set to "1". Otherwise, it is set to "0". With a block search instruction (CPI, CPIR, CPD or CPDR), the Z flag is set to "1" if the source data and the accumulator data match. With a block I/O instruction (INI, IND, OUTI or OUTD), the Z flag is set to "1" if the content of the B register used as the byte counter is "0" at the end of comparison.

- Parity overflow flag (P/V)

  This flag has two functions. One is the parity flag (P) that indicates the result of a logic operation (AND A,B etc.). The P flag is set to "1" if the parity is even as a result of the operation on signed values by two's complement. It is set to "0" is the parity is odd. With a block search instruction (CPI, CPIR, CPD or CPDR) and a block transfer instruction (LDI or LDD), the P flag indicates the state of the byte counter (register pair BC). It is set to "1" if the byte counter is not "0" and to "0" when the byte counter becomes "0" (at the end of comparison or data transfer). The content of the interrupt enable flip-flop (IFF) is saved to the P flag when the contents of the R register or I register are transferred to the accumulator.

  The other use of the P/V flag is the overflow flag (V) that indicates whether an overflow has occurred or not as a result of an arithmetic operation. The V flag is set to "1" when the value in the accumulator gets out of a range of the maximum value $+127$ and the minimum value -128 and therefore cannot be correctly represented as a two's complement notation.

  Whether the P/V flag operates as the P flag or V flag is determined by the type of the instruction executed.

- Carry flag (C)

  The C flag is set to "1" if a carry occurs from bit 7 of the accumulator or a borrow occurs as a result of an operation.

The following two flags are not available to the programmer for the test and set ("1")/reset ("0") purposes. They are internally used by the MPU for BCD arithmetic operations.

- Half carry flag (H)

  The H flag is used for holding the carry or borrow from the low-order 4 bits of a BCD operation result. When a DAA instruction (decimal adjust) is executed, the MPU automatically uses the H flag to adjust the result of a decimal addition or subtraction.

- Add/subtract flag (N)

  In BCD operation, algorithm is different between addition and subtraction. The N flag indicates whether the executed operation is addition or subtraction.
  For how the flags change depending on the instruction, see 3.2.4 "TMPZ84C011A Instruction Set".

(c) General-purpose registers (B, C, D, E, H, L)

General-purpose registers consist of 8 bits each. They are used as 16-bit register pairs (BC, DE, HL) as well as separate 8-bit registers to supplement the accumulator . The B register and the register pair BC are used as a counter when a block I/O, block transfer, or search instruction is executed. The register pair HL has various memory addressing features as compared with the register pairs BC and DE.

(2) Alternate registers (A', F', B', C', D', E', H', L')

The configuration of the alternate registers is exactly the same as that of the main registers. There is no instruction that handles the alternate registers directly. The data in the alternate registers are processed by moving them into the main registers by means of exchange instructions as shown below:

EX AF, AF' (A↔A', F↔F')
EXX (B↔B', C↔C', D↔D', E↔E', H↔H', L↔L')

When a high-speed interrupt response has been requested within the system, these instructions can be used to quickly move the data from the accumulator, flag registers, and general-purpose registers on the main or alternate side into the corresponding registers. This eliminates the need for transferring the register contents to/from the external stack during execution of the interrupt handling routine, thereby shortening the interrupt servicing time greatly.

(3) Special registers (I, R, IX, IY, SP, PC)

(a) Interrupt page address register (I)

The TMPZ84C011A provides two kinds of interrupts: maskable interrupt ($\overline{INT}$) and non-maskable interrupt ($\overline{NMI}$). The maskable interrupt provides three modes (0, 1, and 2) in which the interrupt is handled. These modes can be selected by instructions IM0, IM1, and IM2 respectively. In mode 2, any memory location can be called indirectly depending on the interrupt. For this purpose, the I register stores the high- order 8 bits of the indirect address. The low-order 8 bits are supplied from the interrupting peripheral LSI. This scheme permits calling the interrupt handling routine from any memory location in an extremely short access time. For the details of interrupts, see (IV) "Interrupt Capability".

(b) Memory refresh register (R)

The R register is used as the memory refresh counter when the dynamic RAM is used for memory. This perimits using of the dynamic memory in the same manner as the static memory. This 8- bit register is automatically incremented for each instruction fetch. While the MPU decoder executes the fetched instruction, the contents of the R register are synchronized with the refresh signal to place the low-order 7 bits on the address bus. This operation is all performed by the MPU and, therefore, need not perform a special processing by program. The MPU operation is not delayed by this operation. During refresh, the contents of the I register are placed on the high-order 8 bits of the address bus.

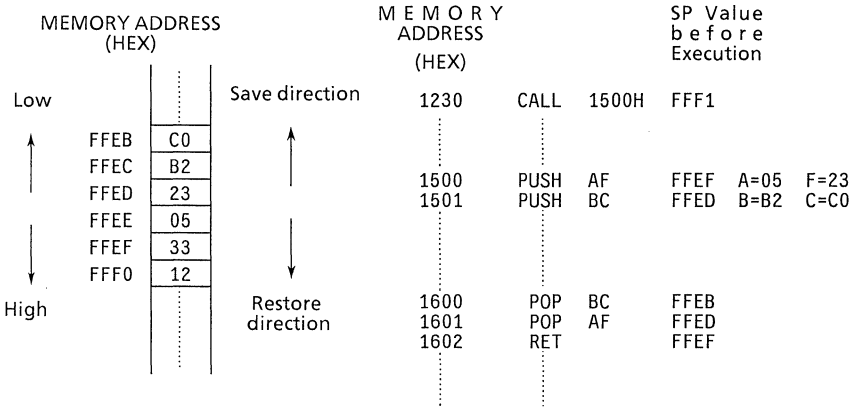
(c) Index registers (IX, IY)

The two independent index registers IX and IY holds the 16-bit base address when used in the index addressing mode. In this addressing mode, the memory address obtained by adding the contents of an index register to the displacement value (for example, LD IX + 40H) is specified. This mode is convenient for using data tables. Also these registers can be used separately for memory addressing and data retaining registers.

(d) Stack pointer (SP)

The stack pointer is a 16-bit register to provide the start address information in the stack area in the external RAM. The content of the stack pointer is decremented at the execution of a call instruction or PUSH instruction or interrupt handling and is incremented at the execution of a return instruction or POP instruction. At the execution of a call instruction or interrupt handling, the current content of the program counter is saved into the stack. At the execution of a return instruction, the content is restored from the stack to the program counter. These operations are all pereformed by the MPU automatically. However, the other registers are not saved or restored automatically. For the storing of these registers, alternate register exchange instruction (EX and EXX) or a PUSH and a POP instructions must be used. When a PUSH instruction is executed, the contents of the specified register are saved into the stack. When a POP instruction is executed, the contents of the stack are moved to the specified register.

These data are restored on a last-in, first-out basis. Use of the stack permits processing of multiple-level interrupts, very deep subroutine nestings, and various data manipulation very easily. The stack pointer is not initialized in the hardware approach. Therefore, it is required to specify initialization (to the highest address in the stack) in the initialization program by allocating the stack area in RAM in during programming.



The above example shows the stack pointer and stack operations in which the instructions starting with the CALL at address 1230H and ending at with the RET at address 1602H have been executed. However, it is assumed that there is no instruction or interrupt other than shown above that uses the above stack during the execution. When the value of the stack pointer before executing the CALL instruction at address 1230H indicates address FFF1H, address 1233H is stored at addresses FFF0H and FFEFH because the CALL instruction consists of 3 bytes, then the stack pointer is decremented. Similarly, the data are saved or restored sequentially according to the instructions. These stack and stack pointer operations are all performed automatically.

(e) Program counter (PC)

The program counter holds, in 16 bits, the memory address of the instruction to be executed next. The MPU fetches the instruction from the memory location indicated by the program counter. When the content of the program counter is put on the address bus, the program counter is incremented automatically. However, it is not incremented with a jump instruction, a call instruction, or interrupt processing. Instead, the specified new address set on it. With a return instruction, the content restored from the stack is set on the program counter. These operations are all performed automatically and therefore, transparent to the programmer.

[2] Halt capability

When a HALT instruction has been executed, the MPU is put in the halt state. The halt capability can be used to halt the MPU against the external interrupts, thereby reducing the power dissipation. In the halt state the states of MPU's internal registers are all retained. The halt state is cleared by reset or when an interrupt is accepted. For the details of halt operation, see [3] "Basic Timing".

(1) Halt operation

When a HALT instruction has been executed, the MPU sets the $\overline{\text{HALT}}$ signal to "0" to tell the outside that the MPU is going to get in the halt state. Actually, the MPU in the halt state continues executing NOPs if there is the system clock input. However, the program counter is not incremented. This keeps the refresh signal generated when the dynamic memory is used. During halt, the MPU's internal states are retained. The TMPZ84C011A contains the clock generator/controller, easily implementing the clock input control for these halt operations.

(2) Clearing the halt state

The halt state is cleared by accepting an interrupt (the $\overline{INT}$ or $\overline{NMI}$ signal input) or by reset (the $\overline{RESET}$ signal input). When an interrupt is accepted, the halt state is cleared and the interrupt handling routine is executed. However, a maskable interrupt (INT) cannot be accepted unless the interrupt enable flip-flop (IFF) is set.

Note that when the halt state is cleared by the $\overline{RESET}$ signal, the MPU is reset and the program counter is set to "0".

[3] RESET Signal

Holding the $\overline{RESET}$ pin at the low level ("0") under the following conditions, the MPU's internal states are reset:

(1) The power voltage level is within the operational voltage range.

(2) System clock stabilization.

(3) Holding the $\overline{RESET}$ signal at the low level ("0") for at least 3 full clock cycles. When the $\overline{RESET}$ signal goes high ("1"), the MPU starts executing instructions from address 0000H after at least 2T state dummy cycles.
When reset, the MPU performs the following processing:

- Program counter

  0000H is set.

- Interrupt

  The interrupt enable flip-flop (IFF) is reset to "0" to disable the maskable interrupt. For the maskable interrupt processing, mode 0 is specified.

- Control output

  All control outputs are made inactive ("1"). Therefore, the halt state is also cleared.

- Interrupt page address register (I register)

  The content of the I register becomes 00H.

- Refresh register (R register)

  The content of the R register becomes 00H.

The registers other than above and the external memory do not change. Therefore, they must be initialized as required.

[4] Interrupt Capability

The interrupt capability is used to pause the execution of the currently executed program upon request from a peripheral LSI, executing the requested processing before. Normally, this interrupt processing routine contains the data exchange and transfer of status and control information between the MPU and the peripheral LSI. When this routine has been completed, the MPU returns to the state before the interrupt was accepted.

The TMPZ84C011A provides the non-maskable interrupt (NMI) and maskable interrupt (INT) capabilities which are detected by the $\overline{NMI}$ and $\overline{INT}$ interrupt request signals, respectively. A non- maskable interrupt, when caused by a peripheral LSI, is accepted unconditionally. This interrupt is used to support critical functions such as the protection of the system from unpredictable happening including power outages. A maskable interrupt can be enabled or disabled by program. For example, if the timer is used and, therefore, an interrupt is not desired, the system can be programmed to disable the interrupt. Table 3.2.1 lists the processing by interrupt cause.

(1) Interrupt enable/disable

A non-maskable interrupt cannot be disabled by program, while a maskable interrupt can be enabled or disabled by program. The MPU has the interrupt enable flip-flop (IFF). A maskable interrupt can be enabled or disabled by setting this flip-flop to "1" (set) or "0" (reset) through an EI instruction (enable) or a DI instruction (disable) in program. Actually, the IFF consists of two flip-flops IFF1 and IFF2. IFF1 is used to select between the enable and disable of a maskable interrupt. IFF2 holds the state of IFF1 effective before a nonmaskable interrupt has been accepted. Both IFF1 and IFF2 are reset to "0" when any of the following conditions occurs, disabling an interrupt:

Actually, the waiting maskable interrupt request is accepted after the execution of the instruction that follows the EI instruction. This delay by one instruction is caused by accepting an interrupt after completion of the execution of a return instruction if the instruction following the EI instruction is a return instruction. In the following operations, the contents of IFF1 and IFF2 are the same.

- MPU reset
- Execution of DI instruction
- Acceptance of maskable interrupt

Both IFF1 and IFF2 are set to "1" when the following condition occurs, enabling an interrupt:

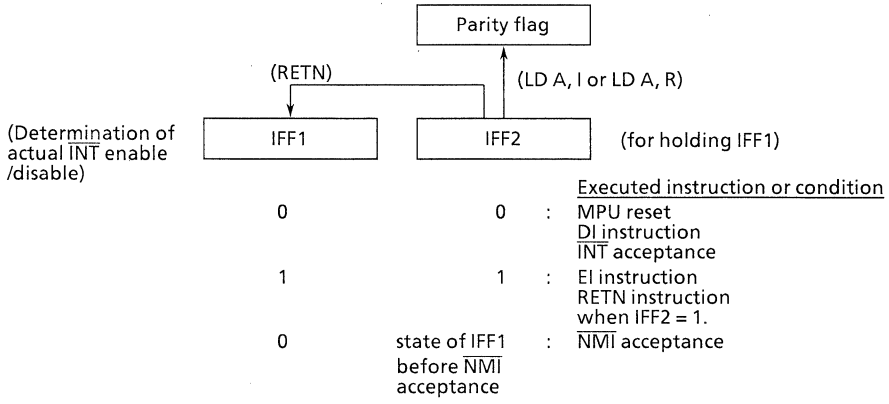
- Execution of EI instruction

Table 3.2.1 Processing by Interrupt Cause

Interrupt Source	Priority	Programmed condition		Vector address	Interrupt return instruction
Non-maskable interrupt (the falling edge of $\overline{NMI}$)	1	None		Address 66H	RETN
Maskable interrupt ($\overline{INT}$ becomes "0" at instruction's last clock)	2	IFF = 1	Mode 0	Instruction from peripheral LSI. Normally, CALL or RST instruction.	(Note) RETI
			Mode 1	Address 38H.	
			Mode 2	The address indicated by the data table (memory) at the address specified by I register (high-order 8 bits) and data from peripheral LSI (low-order 8 bits, LSB = "0").	

110489

Note : Mode 0 applies when the instruction from peripheral LSI is CALL or RST instruction.



110489

Figure 3.2.4 Interrupt Enable Flip-Flop (IFF)

When a non-maskable interrupt has been accepted, IFF1 is reset to "0" (interrupt disable) until an EI or RETI instruction is executed to prevent a next interrupt accepting. For this purpose, the state (interrupt enable/disable) of IFF1 immediately before non-maskable interrupt acceptance must be stored. This state is copied into IFF2 upon acceptance of a non-maskable interrupt. The content of IFF2 is copied into the parity flag at the execution of the following instructions, so that the copied data can be tested or stored:

- The load instruction (LD A, I) to load from the I register to the accumulator.
- The load instruction (LD A, R) to load from the R register to the accumulator.

When the return instruction from the non-maskable interrupt (RETN) is executed, the content of IFF2 is copied back to IFF1. If an operation which changes the contents of IFF2 (due to the execution of EI or DI instruction, for example) has not been performed during interrupt handling. IFF1 automatically returns to the state which was effective immediately before the interrupt acceptance. Table 3.2.1 lists the states of IFF1 and IFF2 after the execution of interrupt- related instructions.

Table 3.2.1 State of IFF1 and IFF2

Operation sequence	IFF1	IFF2	Remarks
MPU reset	0	0	
EI	1	1	
NMI acceptance	0	1	
LD A, I	*	*	Parity flag←IFF2
RETN	1	1	IFF1←IFF2
LD A, R	*	*	Parity flag←IFF2
INT acceptance	0	0	
RETI	*	*	
EI	1	1	
NMI acceptance	0	1	
DI	0	0	
RETN	*	*	

Note : * =no change. 110489

(2) Interrupt processing

With a non-maskable interrupt, the internal NMI flip-flop is set to "1" at the falling edge of the interrupt signal, $\overline{NMI}$. The state of this flip-flop is sampled at the rising edge of the last clock of each instruction to accept an interrupt. A maskable interrupt is accepted if the interrupt signal $\overline{INT}$ is low ("0") at the rising edge of the last clock of each instruction and the interrupt enabled state (IFF = 1 and $\overline{BUSREQ}$ signal = inactive ("1")) is on. Described below is the processing to be performed after a non-maskable interrupt and a maskable interrupt are accepted:

(a) Non-maskable interrupt (NMI)

When a non-maskable interrupt has been accepted, the MPU performs the following processing:

① The internal NMI flip-flop is reset "0".

② IFF1 is reset to "0", disabling the maskable interrupt.
The contents of the IFF1 immediately before the interrupt acceptance are copied into the IFF2.

③ The current contents of the program counter are saved into the stack.

④ The instructions starting from non-maskable interrupt vector address 66H are executed.

The non-maskable interrupt processing program terminates by executing the RETN instruction. This return instruction performs the followings:

① The contents of the current IFF2 are copied into IFF1.

② The contents of the program counter are restored from the stack.

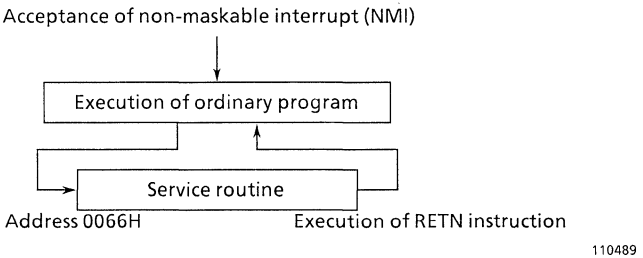


Figure 3.2.5 Non-Maskable Interrupt Processing

(b) Maskable interrupt (INT)

When a maskable interrupt has been accepted, the MPU performs the following processings:

① Both IFF1 and IFF2 are reset to "0", disabling the maskable interrupts.

② The current contents of the program counter are saved into the stack.

③ A maskable interrupt is serviced in one of the three modes 0, 1 and 2. A mode is selected by executing the instruction IM0, IM1 or IM2 before the interrupt is serviced. The instructions are executed starting from the vector address corresponding to the selected mode.

- Mode 0

In mode 0, the interrupting peripheral LSI puts a restart instruction (RST) or a call instruction (CALL) on the data bus and the MPU executes the interrupt service routine according to that instruction.

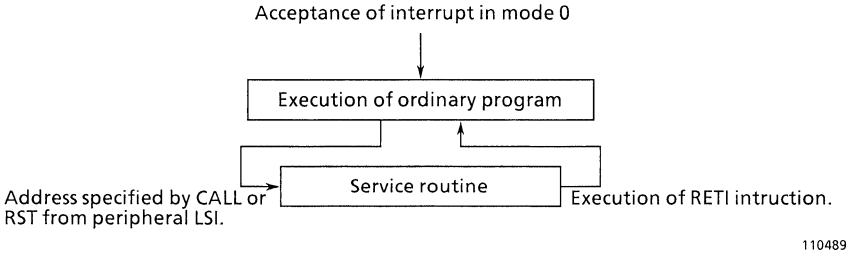


Figure 3.2.6 Interrupt Processing in Mode 0

- Mode 1

When an interrupt is accepted in mode 1, restart is performed from address 0038H. Therefore, the service routine for this interrupt is programmed from the address 0038H.

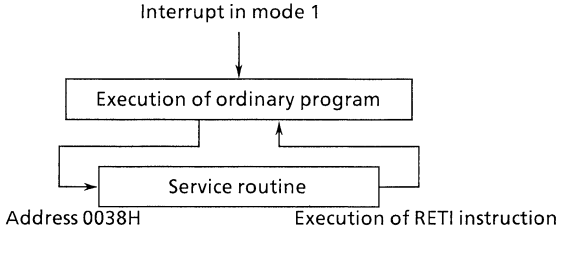


Figure 3.2.7 Interrupt Processing in Mode 1

- Mode 2

The interrupt processing in mode 2 requires a 16-bit pointer consisting of the high-order 8 bits of the I register and the low-order 8 bits (with the LSB = "0") of the data captured from the interrupting CTC or TLCS-Z80 family peripheral LSI. Therefore, the necessary value must be loaded in the I register beforehand. This pointer is used to specify the memory address in the table. The contents of the specified addresses and the next address provide the start address of the service routine. Therefore, use of this mode requires to present table of the service routine's start address (16 bits) by program at convenient location. This location can be anywhere in memory. The LSB of the table pointer is set to "0" because a 2-byte data is needed to specify the service routine start address just in 16 bits and start that address from an even-number address. In the table, the start address begins with the low- order byte followed by the high-order byte as shown in Figure 3.2.8

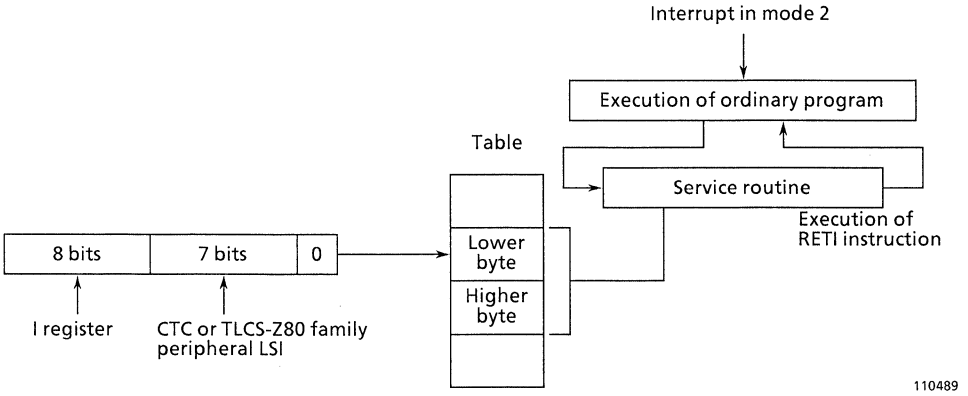


Figure 3.2.8 Interrupt Processing in Mode 2

Mode 2 is used in the daisy chain interrupt processing using the CTC and/or TLCS-Z80 family LSI. The CTC and TLCS-Z80 family perpheral LSIs all contain the interrupt priority controller of daisy chain structure. In this interrupt structure, the interrupt request signals are put in the series order and given priorities for processing when two or more maskable interrupt requests occur at a time. Only the interrupt vector from the peripheral LSI having the highest priority is put on the data bus. By receiving the vector interrupt in mode 2, the processing for that peripheral LSI can be performed. When an interrupt comes from a peripheral LSI having a priority higher than that of the current peripheral LSI during the execution of its interrupt processing routine, the new interrupt can be enabled by the EI instruction to form an interrupt nesting.

The maskable interrupt processing program terminates by executing an RETI instruction. This return instruction performs the following processing:

- Restores the content of the program counter from the stack.
- Notifies the requesting peripheral LSI of the termination of interrupt processing.

3.2.3 MPU Status Transition Diagram and Basic Timing

The following describes the MPU status transition and the basic timing of each MPU operation.

[1] Instruction Cycle

Each TMPZ84C011A instruction is executed by combining the basic operations of memory read/write, input/output, bus request/acknowledge, and interrupt. These basic operations are performed in synchronization with the system clock (the CLK signal).

One clock period is called a state (T). The smallest unit of each basic operation is called a machine cycle (M). Each instruction consists of 1 to 6 machine cycles and each machine cycle, 3 to 6 clock states basically. However, the number of clock states in a machine cycle can be increased by the $\overline{\text{WAIT}}$ signal described later on. Figure 3.2.9 shows an example of the basic timing of a 3-machine-cycle instruction.

The first machine cycle (M1) of each instruction is the cycle in which the Opcode of the instruction to be executed next is read (this is called the Opcode fetch cycle). The Opcode fetch cycle basically consists of 4 to 6 clock states. In the machine cycle that follows the Opcode fetch cycle, data is transferred between the MPU and the memory or peripheral LSIs. This operation basically consists of 3 to 5 clock states.

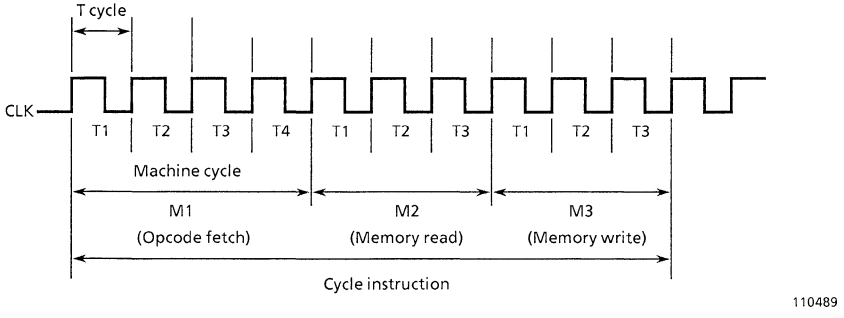


Figure 3.2.9 Example of MPU Basic Timing (3-Machine-Cycle Instruction)

[2] Status Transition Diagram

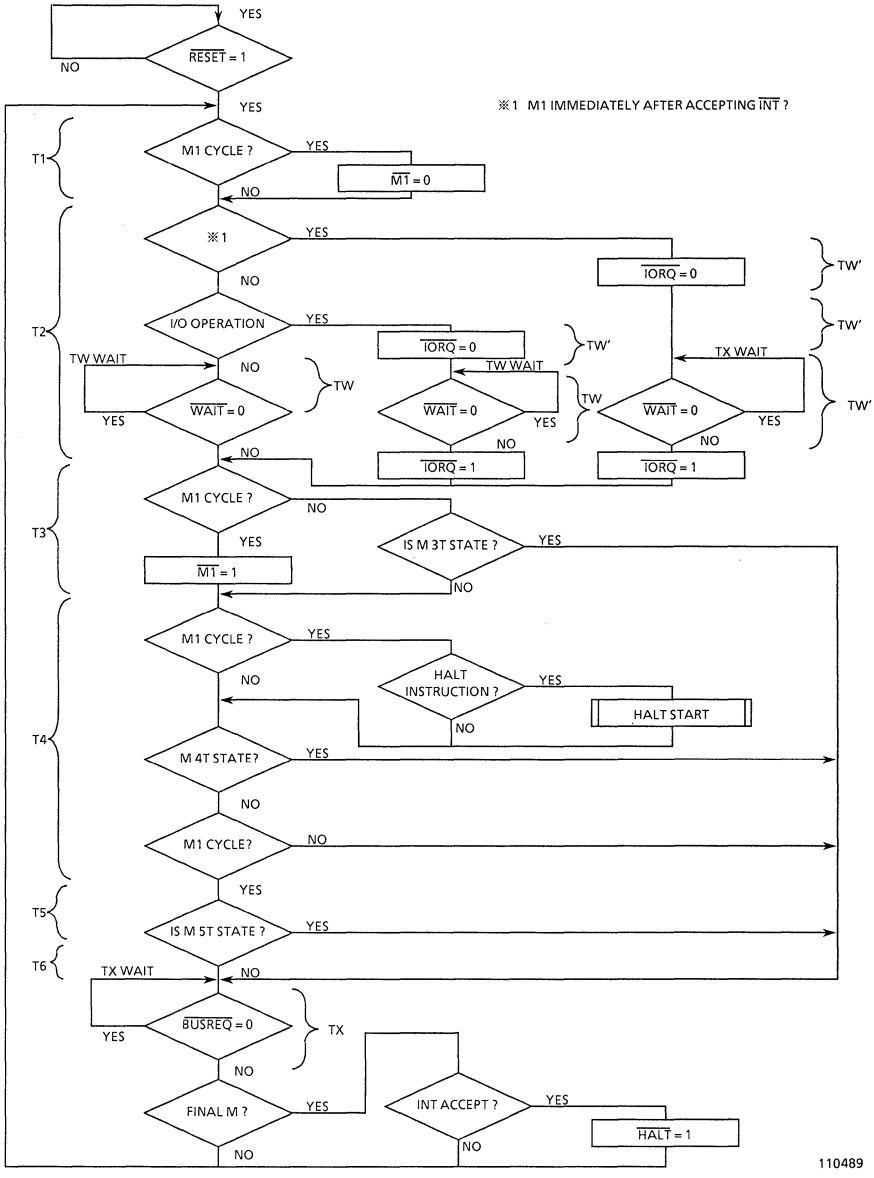


Figure 3.2.10 Status Transition Diagram

[3] Basic Timings

(1) Opcode fetch cycle (M1)

In the Opcode fetch cycle, MPU reads an Opcode from among the machine-language codes in memory. This is also called the M1 cycle for it is the first machine cycle to execute each instruction.

Figure 3.2.11 shows the basic timing of a basic Opcode fetch cycle. In clock state T1, the content of the program counter is put on the address bus. The $\overline{M1}$ signal goes "0", telling the outside the MPU that this is the Opcode fetch cycle. At the same time, $\overline{MREQ}$ and $\overline{RD}$ signals go "0". When the $\overline{MREQ}$ signal goes "0", the address signal has already been stabilized. Therefore, this signal can be used for the memory chip enable signal. The $\overline{RD}$ signal indicates that the MPU is ready to accept the data from memory. By these signals, the MPU accesses memory to put the Opcode in the instruction register. The MPU samples the $\overline{WAIT}$ signal at the falling edge of clock state T2. If the $\overline{WAIT}$ signal is "0" at the falling edges of clock state T2 and the following wait state (T_W), the next state becomes clock state T_W. Figure 3.2.12 shows the delay state of the Opcode fetch cycle caused by the $\overline{WAIT}$ signal.

The data (Opcode) on the data bus is fetched at the rising edge of clock state T3 then, at the rising edges of the same state, the $\overline{MREQ}$, $\overline{RD}$, and $\overline{M1}$ signals go "1". In clock state T3, a memory refresh address is put on the low-order 7 bits of the address bus and the $\overline{RFSH}$ signal goes "0" and the $\overline{MREQ}$ signal goes "0" again. This signal indicates that the memory refresh cycle is on. At this time, the contents of the I register are on the high-order 8 bits of the address bus and the 7 bits of the R register contents are on the low-order 7 bits of the address bus. By using the $\overline{RFSH}$ and $\overline{MREQ}$ signals, memory refresh is performed in clock states T3 and T4. However, the $\overline{RD}$ signal remains "1" because the contents of the memory refresh address are not put on the data bus.

In clock state T4, the $\overline{MREQ}$ signal returns to "1". The refresh address is kept output until the rising edge of the clock state T1 of the next machine cycle, during which the $\overline{RFSH}$ signal is also "0". The cycle delay state caused by setting the $\overline{WAIT}$ signal to "0" is the same with the memory read/write, input/output, and maskable interrupt acknowledge cycles. The following description the diagram of the cycle delay state caused by the $\overline{WAIT}$ signal's going "0" is omitted.

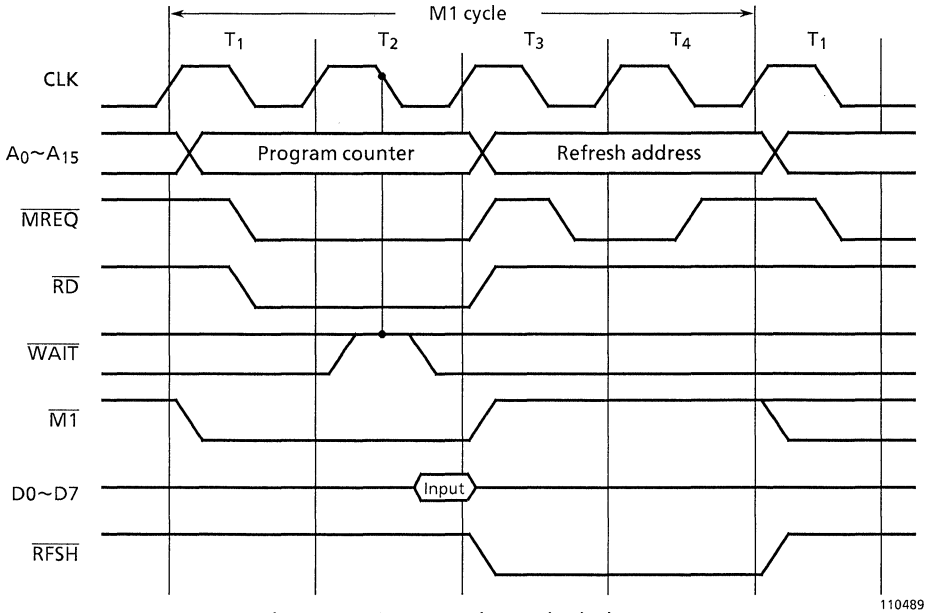


Figure 3.2.11 Opcode Fetch Timing

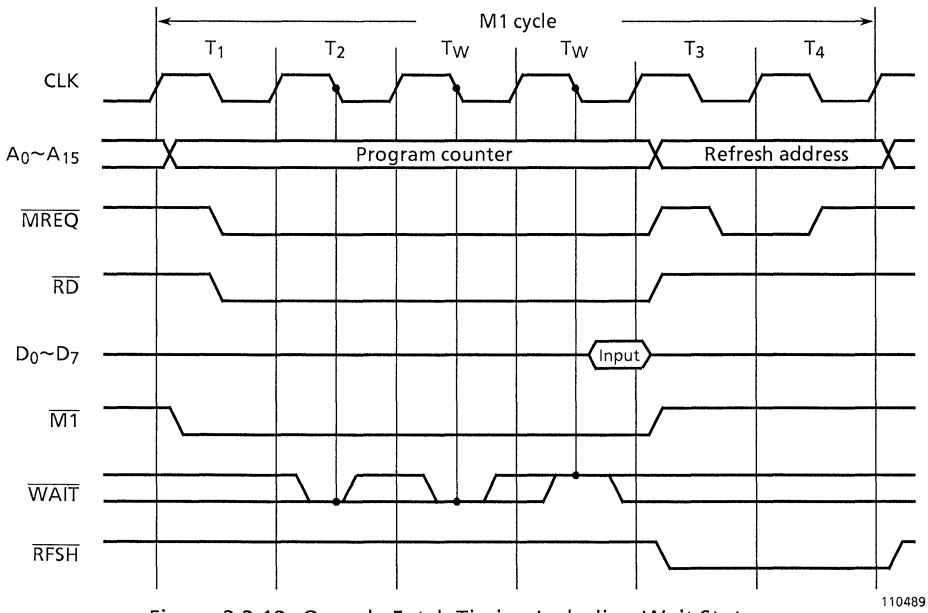


Figure 3.2.12 Opcode Fetch Timing Including Wait State

(2) Memory read/write operations

Figure 3.2.13 shows the basic timing of memory read/write operations (except for the Opcode fetch cycle) in the same diagram for convenience.

In each case, the memory address signal to read/write data on the address bus is output in clock state T1. The operation in which the $\overline{WAIT}$ signal is sampled in clock state T2 and the following T_W state is the same as the Opcode fetch cycle.

In memory read, memory data is put on the data bus by the address, $\overline{MREQ}$, and $\overline{RD}$ signals. The MPU reads this data.

In memory write, the memory address signals is put on the address bus then the $\overline{MREQ}$ signal is set to "0" to put the write data onto the data bus. When the data bus has been stabilized, the $\overline{WR}$ signal is output in clock state T2. The $\overline{WR}$ signal can be used for the memory write signal.

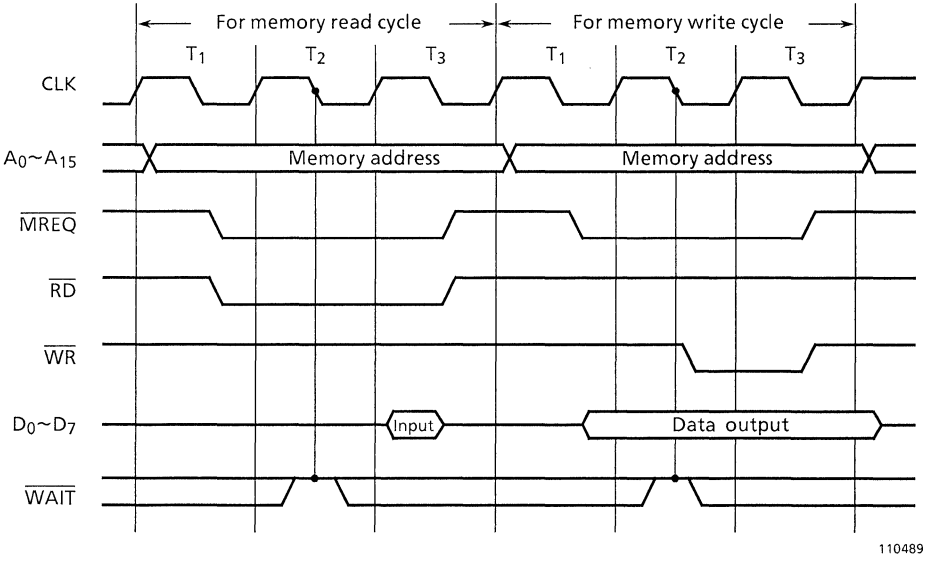


Figure 3.2.13 Memory Read / Write Cycle Timing

(3) Input/output operations

Figure 3.2.14 shows the basic timing of input/output operations. The feature of the I/O operation timing is that, regardless of the state of the $\overline{WAIT}$ signal in clock state T2, the I/O cycle automatically gets in the wait state (T_W*) after clock T2. The $\overline{WAIT}$ signal is sampled at the falling edge of T_W*. If the $\overline{WAIT}$ signal is "0" at the falling edges of T_W* and the following clock state, the I/O cycle moves into clock state T_W. Clock state T_W* is inserted because the $\overline{IORQ}$ signal goes "0" in clock state T2, so that it is too late to sample the $\overline{WAIT}$ signal after decoding the I/O port address. In each of input and output operations, the I/O port address is put on the low-order 8 bits of the address bus in clock state T1. On the high-order 8 bits, the contents of the accumulator or B register are output. In clock state T2, the $\overline{IORQ}$ signal goes "0" instead of the $\overline{MREQ}$ signal. The $\overline{IORQ}$ signal can be used as the chip enable signal for a peripheral LSI.

In an input operation, the contents of the input port are read onto the data bus by the address, $\overline{IORQ}$, and $\overline{RD}$ signals. The MPU reads this data.

In an output operation, the output port address and the output data are respectively put on the address bus and data bus in clock state T1; the $\overline{IORQ}$ and $\overline{WR}$ signals goes "0" in clock state T2. The $\overline{WR}$ signal can be used as the output port write signal.

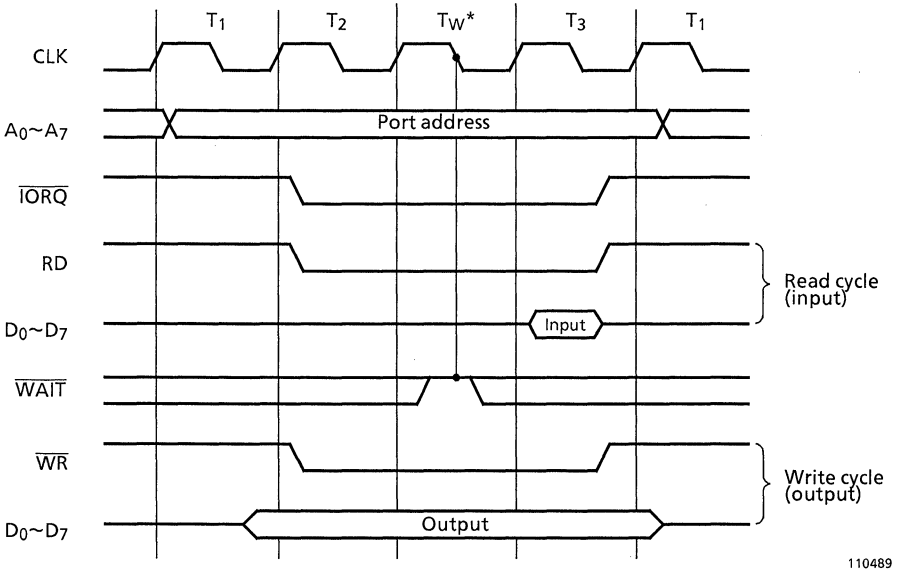


Figure 3.2.14 I/O OPperation Timing

(4) Bus request and bus acknowledge operations

Figure 3.2.15 shows the basic timings of bus request and bus acknowledge operations.

The 3-state address bus (A0 through A15), data bus (D0 through D7), $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ signals controlled by the MPU can be put in the high-impedance state (floating) to electrically disconnect them from the MPU. This operation, by sampling the $\overline{BUSREQ}$ signal at the rising edge of the last clock of each machine cycle, starts with the rising edge of the next clock if this signal is found "0". Subsequently, these buses are controlled by peripheral LSIs. For example, data can be directly transferred between memory and these peripheral LSIs. This state is cleared if the $\overline{BUSREQ}$ signal is found "1" by sampling it at the rising edge of each subsequent clock state (T_X), executing the next machine cycle. During the floating state, the $\overline{BUSACK}$ signal is "0" to tell the peripheral LSIs of it.

In this state, however, no memory refresh is performed and, therefore, the $\overline{RFSH}$ signal is set to "1". Hence, to maintain this state for a long time with a system using dynamic memory, memory refresh must be performed by the external controller.

Note that, in the floating state, neither maskable (INT) nor non-maskable (NMI) interrupts can be accepted.

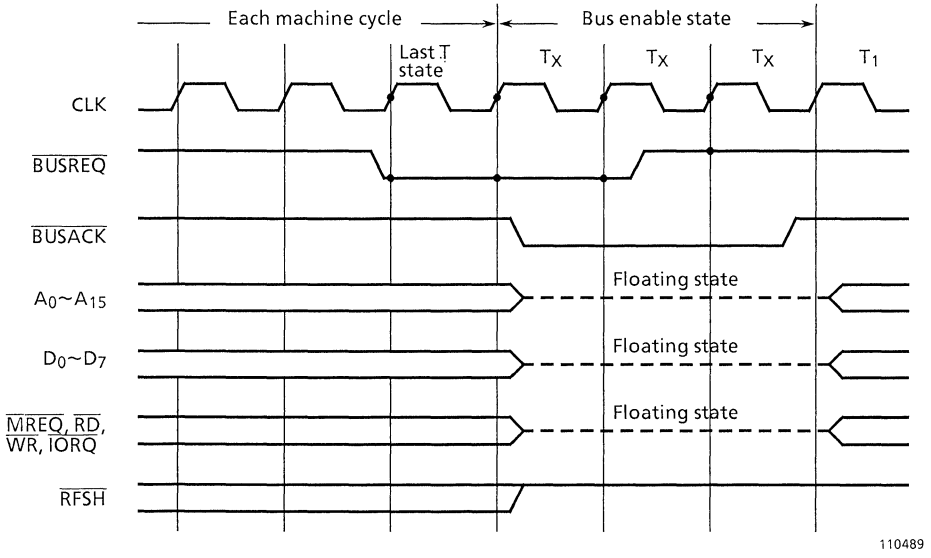


Figure 3.2.15 Bus Request and Bus Acknowledge Timing

(5) Maskable interrupt acknowledge operation

Figure 3.2.16 shows the basic timing of the maskable interrupt acknowledge.

The MPU samples the maskable interrupt request signal ($\overline{INT}$) at the rising edge of the last clock of each instruction execution. If the $\overline{INT}$ signal is found "0", a maskable interrupt is accepted except in the following cases:

- The interrupt enable flip-flop is reset to "0".
- The $\overline{BUSREQ}$ signal is "0".

When a maskable interrupt has been accepted, a special Opcode fetch cycle is generated. In this cycle, 2 clock states of wait state (T_W*) is automatically inserted after the clock state T2. The $\overline{WAIT}$ signal is sampled at the falling edge of the second clock state T_W* and the following clock state T_W and, if the $\overline{WAIT}$ signal is found "0", the instruction cycle gets in the next clock state T_W. In this Opcode fetch cycle, the $\overline{IORQ}$ signal goes "0" in the first T_W* state instead of the $\overline{MREQ}$ signal while in a normal Opcode fetch cycle the $\overline{MREQ}$ signal goes "0" in clock state T1. This tells the maskable interrupt requesting LSI that it can put the 8-bit interrupt vector on the data bus, then reads this data to perform interrupt processing. Therefore, the contents of the program counter put on the address bus are not used. Unlike an ordinary I/O operation, the $\overline{RD}$ signal does not go "0".

In the clock state T3, the memory refresh address signal is put on the address bus for memory refresh like normal Opcode fetch cycle and the $\overline{RFSH}$ signal goes "0". In the subsequent machine cycles (M2 and M3), the current contents of the program counter are saved into the stack. In the machine cycles M4 and M5, the contents of the I register (the high-order 8 bits) and the contents of the address indicated by the address of the vector (the low-order 8 bits) from the CTC or the peripheral LSIs connected externally are put in the program counter.

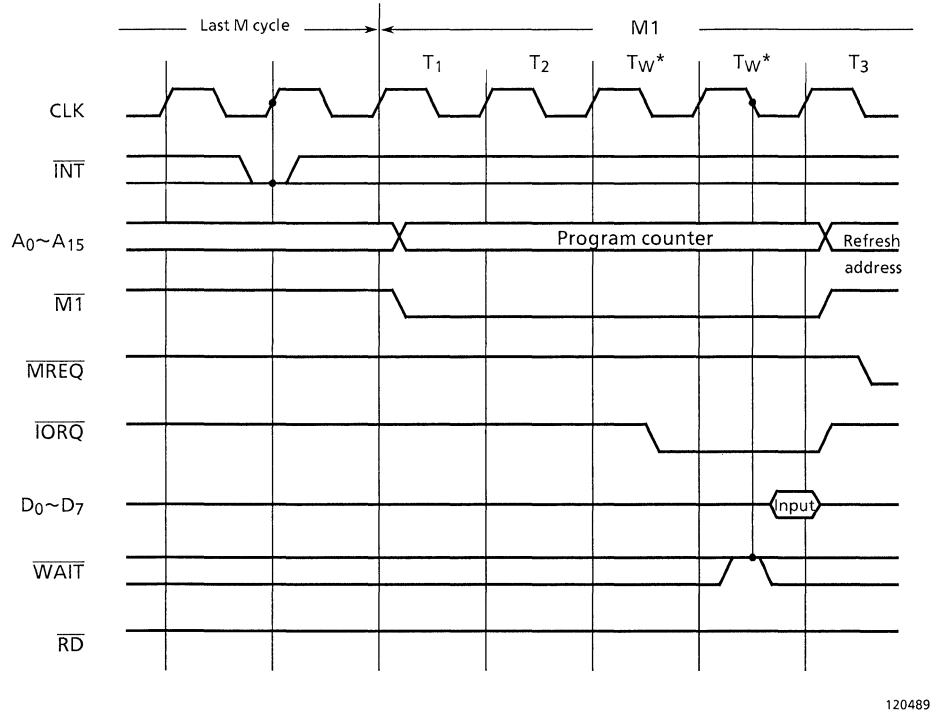


Figure 3.2.16 Maskable Interrupt Acknowledge Timing

(6) Non-maskable interrupt acknowledge operation

Figure 3.2.17 shows the basic timing of non-maskable interrupt acknowledge.

When the non-maskable interrupt request signal ($\overline{NMI}$) goes low, the internal non-maskable flip-flop is set to "1". The $\overline{NMI}$ signal is detected in any timing of each instruction. However, the internal NMI flip-flop is sampled at the rising edge of the last clock of each instruction. Therefore, the $\overline{NMI}$ signal should go low until the last clock state of an instruction

The Opcode fetch cycle for non-maskable interrupt acknowledge is generally the same as the ordinary Opcode fetch cycle. However, the Opcode on the data bus at the time is ignored. The current contents of the program counter are saved into the stack in the subsequent machine cycles (M2 and M3). In the following machine cycle, the operation jumps to address 0066H, the non-maskable interrupt vector address. The machine cycles after these depend on the contents of the fetched Opcode.

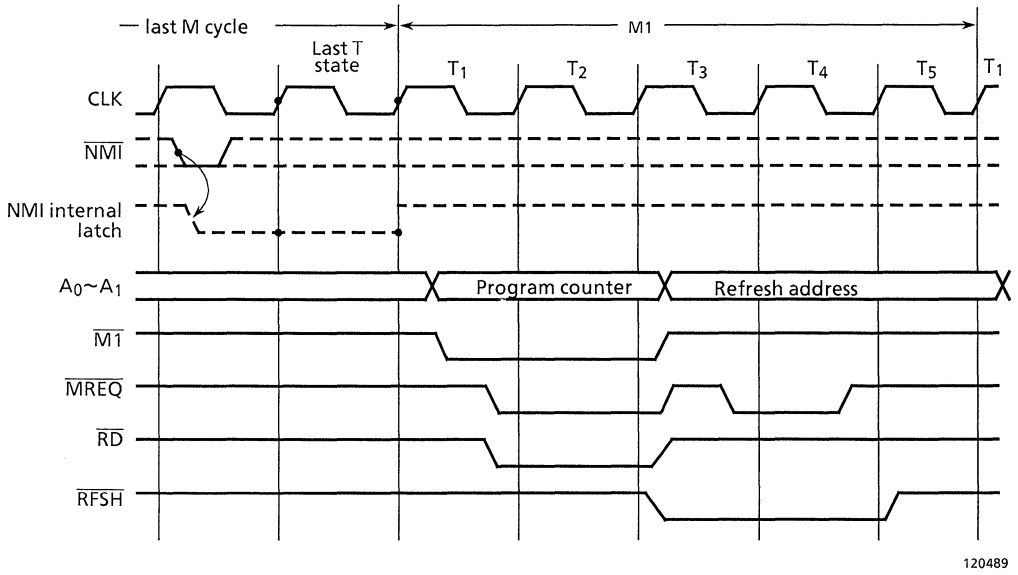


Figure 3.2.17 Non-Maskable Interrupt Acknowledge Timing

(7) Halt operation

Having fetched a HALT instruction in the Opcode fetch cycle, the MPU sets the $\overline{\text{HALT}}$ signal to "0" in synchronization with the falling edge of clock state T4 to tell peripheral LSIs of the fetch and stops operating. if the system clock is kept supplied in the halt state, the MPU continues executing NOP instructions. This is done to output refresh signals when the memory is used. The NOP instruction execution cycle is the same as the ordinary Opcode fetch cycle in which the data on the data bus are ignored.

The halt state is cleared when an interrupt is accepted or the $\overline{\text{RESET}}$ signal is set to "0" to reset the MPU. Figure 3.2.18 shows the halt state clear operation by interrupt acknowledge. An interrupt is sampled at the rising edge of the last clock (clock state T4) of the NOP instruction. A maskable interrupt can be accepted when the $\overline{\text{INT}}$ signal is "0". A non-maskable interrupt is accepted when the internal NMI flip-flop which is set at the falling edge of the $\overline{\text{NMI}}$ signal is set at "1". However, it is required that the interrupt enable flip-flop is set to "1" for a maskable interrupt to be accepted. The interrupt processing for the type of the accepted interrupt gets started from the following cycle.

However, when the supply of the system clock from the CGC has been stopped by the power down operation, it is required to restart the supply of the system clock and input the $\overline{\text{INT}}$ signal until the execution of one instruction is completed or the $\overline{\text{RESET}}$ signal at least 3 clocks are output. Figure 3.2.9 shows the timing of clearing the halt state caused by power down.

For the reset opration, see (8) "Reset operation" below. Note that the $\overline{\text{INT}}$ and $\overline{\text{NMI}}$ signals are shown on the same diagram in Figures 3.2.18 and 3.2.19 for convenience.

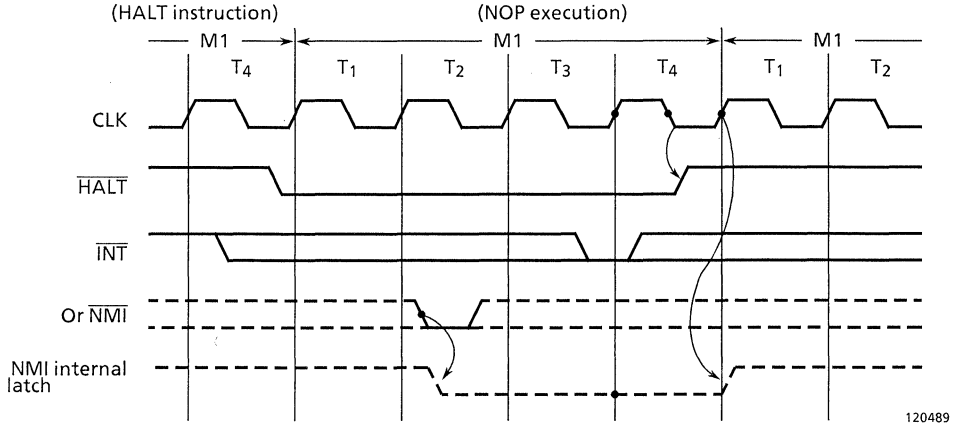


Figure 3.2.18 Timing of Clearing Halt State Caused by Interrupt Acknowledge

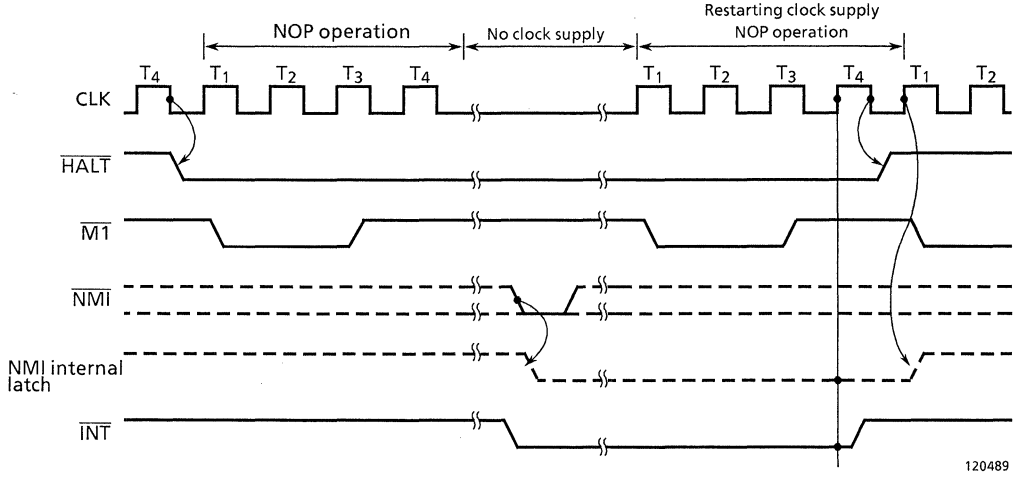


Figure 3.2.19 Timing of Clearing Halt State Caused by Power Down

(8) Reset opration

Figure 3.2.20 shows the basic timing of reset operation.

To reset the MPU, the $\overline{\text{RESET}}$ signal must be kept at "0" for at least 3 clocks. When the $\overline{\text{RESET}}$ signal goes "1", instruction execution starts from address 0000H after a dummy cycle of at least 2 clock states.

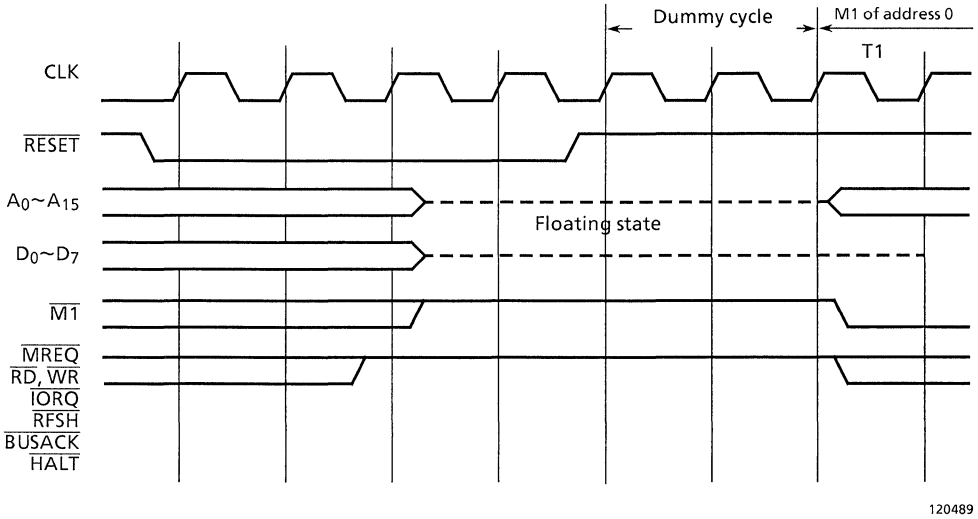


Figure 3.2.20 Reset Timing

To clear the power down state by the $\overline{\text{RESET}}$ signal, the $\overline{\text{RESET}}$ signal must be input until 3 clocks or more are supplied by restarting the supply of the system clock from the CGC.

(9) Evaluation operation

Each of the MPU signals (A0 through A15, D0 through D7, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$, $\overline{HALT}$ and $\overline{M1}$) can be put in the high-impedance state by EV and $\overline{BUSREQ}$ signals to electrically disconnect them from the system.

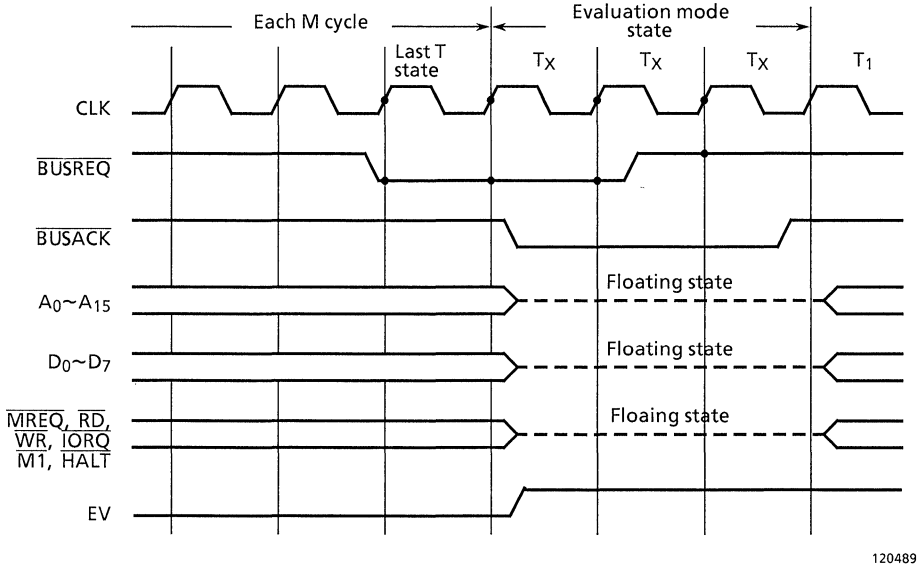


Figure 3.2.21 Evaluation Timing

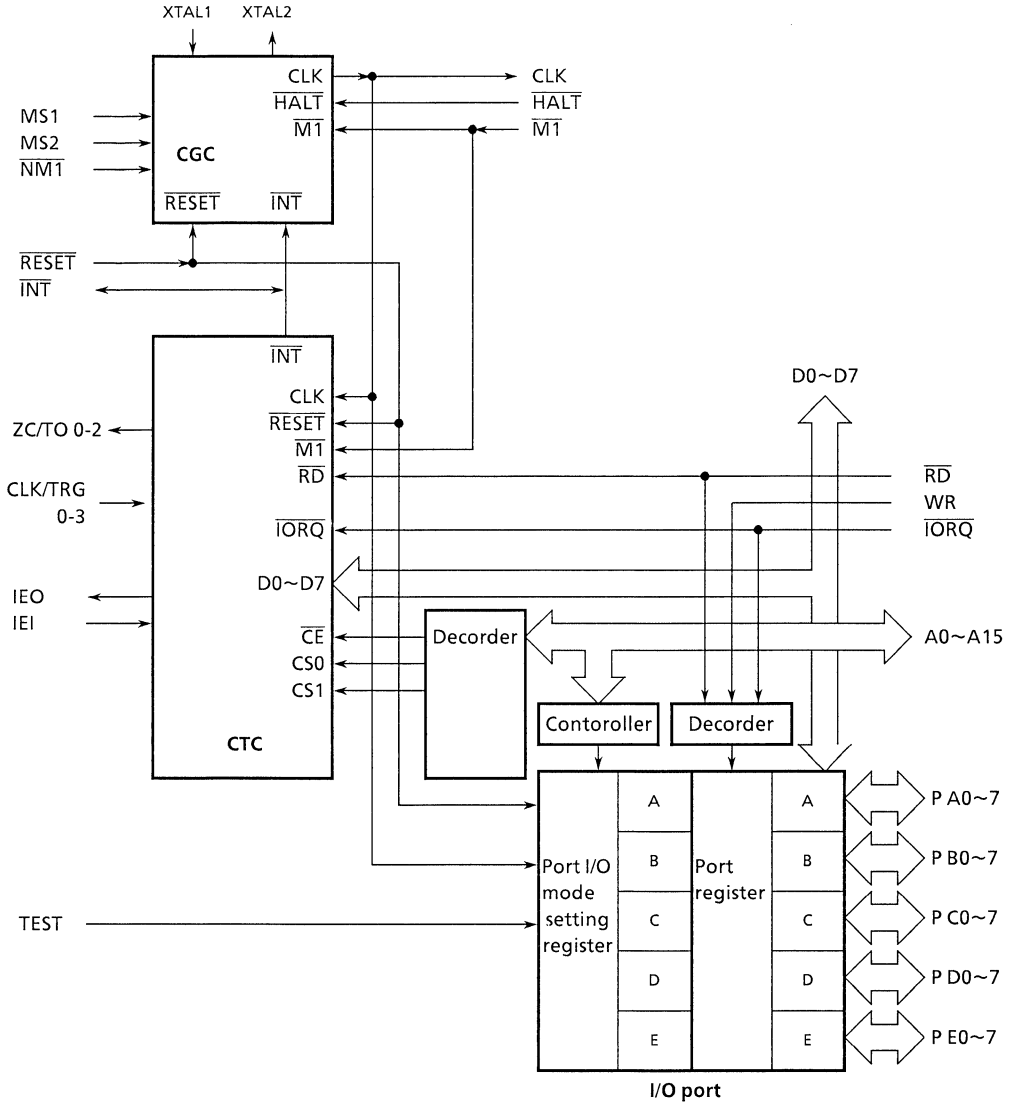


Figure 3.2.22 Block Diagram of the TMPZ84C011A Functioning As Evaluator

3.2.4 TMPZ84C011A Instruction Set

This subsection lists the TMPZ84C011A instruction codes and their functions. The table below lists the symbols and abbreviations used to describe the instruction set. The symbols which require special attention are described in the locations in which they appear.

- Symbols (1/2)

Classification	Symbol	Meaning	
Register	r, g	Register	B, C, D, E, H, L, A,
	t	Register pair	BC, DE, HL
		Stack pointer	SP
	q	Register pair	BC, DE, HL, AF
	p	Register pair	BC, DE
		Index register	IX
		Stack pointer	SP
	s	Register pair	BC, DE
		Index register	IY
		Stack pointer	SP
	t_H	Higher register of register pair (B, D, H)	
		Higher 8 bits of stack pointer (SP)	
	q_H	Higher register of register pair (B, D, H, A)	
	IX_H	Higher 8 bits of index register IX	
	IY_H	Higher 8 bits of index register IY	
	PC_H	Higher 8 bits of program counter (PC)	
	t_L	Lower register of register pair (C, E, L)	
		Lower 8 bits of stack pointer (SP)	
	q_L	Lower register of register pair (C, E, L, F)	
	IX_L	Lower 8 bits of index register IX	
	IY_L	Lower 8 bits of index register IY	
	PC_L	Lower 8 bits of program counter (PC)	
	rb	Bit b (0-7) of register (B, C, D, E, H, L, A)	

120489

- Symbols (2/2)

Classification	Symbol	Meaning
Memory	mn	Memory address represented in 16 bits. m indicates higher 8 bits and n, lower 8 bits.
	$(HL)_b$	Bit b (0-7) of the contents of the memory address indicated by register pair HL.
	$(IX+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IX.
	$(IY+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IY.
Flag change symbol	0	Reset to "0" by operation.
	1	Set to "1" by operation.
	-	No change
	*	Affected by operation
	X	Undefined
	P	Handled as parity flag. P = 0: odd parity P = 1: even parity
	V	Handled as overflow flag. V = 0: No overflow V = 1: Overflow
Operator	←	Transfer
	↔	Exchange
	+	Add
	−	Subtract
	∧	Logical and between bits.
	∨	Logical or between bits.
	⊕	Exclusive or between bits
Others	IFF	Interrupt enable flip-flop
	CY	Carry flag
	Z	Zero flag

120489

TMPZ84C011A Instruction Set (1/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT DATA LOAD	LD r,g	01 rrr ggg	40+r×8+g	r←g	–	–	X	–	X	–	–	–	1	4
	LD r,n	00 rrr 110 nn nnn nnn	06+r×8 n	r←n	–	–	X	–	X	–	–	–	2	7
	LD r,(HL)	01 rrr 110	46+r×8	r←(HL)	–	–	X	–	X	–	–	–	2	7
	LD r,(IX+d)	11 011 101 01 rrr 110 dd ddd ddd	DD 46+r×8 d	r←(IX+d)	–	–	X	–	X	–	–	–	5	19
	LD r,(IY+d)	11 111 101 01 rrr 110 dd ddd ddd	FD 46+r×8 d	r←(IY+d)	–	–	X	–	X	–	–	–	5	19
	LD (HL),r	01 110 rrr	70+r	(HL)←r	–	–	X	–	X	–	–	–	2	7
	LD (IX+d),r	11 011 101 01 110 rrr dd ddd ddd	DD 70+r d	(IX+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),r	11 111 101 01 110 rrr dd ddd ddd	FD 70+r d	(IY+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (HL),n	00 110 110 nn nnn nnn	36 n	(HL)←n	–	–	X	–	X	–	–	–	3	10
	LD (IX+d), n	11 011 101 00 110 110 dd ddd ddd nn nnn nnn	DD 36 d n	(IX+d)←n	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),n	11 111 101 00 110 110 dd ddd ddd nn nnn nnn	FD 36 d n	(IY+d)←n	–	–	X	–	X	–	–	–	5	19
	LD A,(BC)	00 001 010	0A	A←(BC)	–	–	X	–	X	–	–	–	2	7
	LD A,(DE)	00 011 010	1A	A←(DE)	–	–	X	–	X	–	–	–	2	7
	LD A,(mn)	00 111 010 nn nnn nnn mm mmm mmm	3A n m	A←(mn)	–	–	X	–	X	–	–	–	4	13
	LD (BC),A	00 000 010	02	(BC)←A	–	–	X	–	X	–	–	–	2	7
	LD (DE),A	00 010 010	12	(DE)←A	–	–	X	–	X	–	–	–	2	7
	LD (mn),A	00 110 010 nn nnn nnn mm mmm mmm	32 n m	(mn)←A	–	–	X	–	X	–	–	–	4	13
	LD A,I	11 101 101 01 010 111	ED 57	A←I	*	*	X	0	X	IFF	0	–	2	9
	LD A,R	11 101 101 01 011 111	ED 5F	A←R	*	*	X	0	X	IFF	0	–	2	9
	LD I,A	11 101 101 01 000 111	ED 47	I←A	–	–	X	–	X	–	–	–	2	9
	LD R,A	11 101 101 01 001 111	ED 4F	R←A	–	–	X	–	X	–	–	–	2	9
16-BIT DATA LOAD	LD t,mn	00 tt0 001 nn nnn nnn mm mmm mmm	01+t×10 n m	t←mn	–	–	X	–	X	–	–	–	3	10
	LD IX,mn	11 011 101 00 100 001 nn nnn nnn mm mmm mmm	DD 21 n m	IX←mn	–	–	X	–	X	–	–	–	4	14

r	rrr
g	ggg
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

Note : r,g means any of the registers A, B, C, D, E, H, L.
IFF in "Flag" column indicates that the content of the interrupt enable flip-flop is copied into the P/V flag.

120489

TMPZ84C011A Instruction Set (2/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z	H	P/V	N	C	No. OF CY- CLES	No. OF STA- TES
16-BIT DATA LOAD	LD IY,mn	11 111 101 00 100 001 nn nnn nnn mm mmm mmm	FD 21 n m	IY←mn	–	–	X	–	X	–	–	–	4	14
	LD HL,(mn)	00 101 010 nn nnn nnn mm mmm mmm	2A n m	H←(mn+1) L←(mn)	–	–	X	–	X	–	–	–	5	16
	LD t,(mn)	11 101 101 01 tt1 011 nn nnn nnn mm mmm mmm	ED 4B+t×10 n m	t_H←(mn+1) t_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IX,(mn)	11 011 101 00 101 010 nn nnn nnn mm mmm mmm	DD 2A n m	IX_H←(mn+1) IX_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IY,(mn)	11 111 101 00 101 010 nn nnn nnn mm mmm mmm	FD 2A n m	IY_H←(mn+1) IY_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD (mn),HL	00 100 010 nn nnn nnn mm mmm mmm	22 n m	(mn+1)←H (mn)←L	–	–	X	–	X	–	–	–	5	16
	LD (mn),t	11 101 101 01 tt0 011 nn nnn nnn mm mmm mmm	ED 43+t×10 n m	(mn+1)←t_H (mn)←t_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IX	11 011 101 00 100 010 nn nnn nnn mm mmm mmm	DD 22 n m	(mn+1)←IX_H (mn)←IX_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IY	11 111 101 00 100 010 nn nnn nnn mm mmm mmm	FD 22 n m	(mn+1)←IY_H (mn)←IY_L	–	–	X	–	X	–	–	–	6	20
	LD SP,HL	11 111 001	F9	SP←HL	–	–	X	–	X	–	–	–	1	6
	LD SP,IX	11 011 101 11 111 001	DD F9	SP←IX	–	–	X	–	X	–	–	–	2	10
	LD SP,IY	11 111 101 11 111 001	FD F9	SP←IY	–	–	X	–	X	–	–	–	2	10
	PUSH q	11 qq0 101	C5+q×10	(SP−2)←q_L,(SP−1)←q_H, SP←SP−2	–	–	X	–	X	–	–	–	3	11
	PUSH IX	11 011 101 11 100 101	DD E5	(SP−2)←IX_L,(SP−1)←IX_H SP←SP−2	–	–	X	–	X	–	–	–	4	15
	PUSH IY	11 111 101 11 100 101	FD E5	(SP−2)←IY_L,(SP−1)←IY_H SP←SP−2	–	–	X	–	X	–	–	–	4	15
	POP q	11 qq0 001	C1+q×10	q_H←(SP+1),q_L←(SP), SP←SP+2	–	–	X	–	X	–	–	–	3	10
	POP IX	11 011 101 11 100 001	DD E1	IX_H←(SP+1),IX_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
	POP IY	11 111 101 11 100 001	FD E1	IY_H←(SP+1),IY_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
*1	EX DE,HL	11 101 011	EB	DE↔HL	–	–	X	–	X	–	–	–	1	4
	EX AF,AF'	00 001 000	08	AF↔AF'	–	–	X	–	X	–	–	–	1	4
	EXX	11 011 001	D9	BC↔BC',DE↔DE',HL↔HL'	–	–	X	–	X	–	–	–	1	4

t	tt
BC	00
DE	01
HL	10
SP	11

q	qq
BC	00
DE	01
HL	10
AF	11

Note : t is any of the register pairs BC, DE, HL, SP.
q is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively. (Ex) $BC_L = C$, $AF_H = A$.

*1 : EXCHANGE

120489

TMPZ84C011A Instruction Set (3/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES	
EXCHANGE	EX (SP),HL	11 100 011	E3	H↔(SP+1),L↔(SP)	–	–	X	–	X	–	–	–	5	19	
	EX (SP),IX	11 011 101 11 100 011	DD E3	IX_H↔(SP+1) IX_L↔(SP)	–	–	X	–	X	–	–	–	6	23	
	EX (SP),IY	11 111 101 11 100 011	FD E3	IY_H↔(SP+1) IY_L↔(SP)	–	–	X	–	X	–	–	–	6	23	
BLOCK TRANSFER BLOCK SEARCH	LDI	11 101 101 10 100 000	ED A0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDIR	11 101 101 10 110 000	ED B0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	LDD	11 101 101 10 101 000	ED A8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDDR	11 101 101 10 111 000	ED B8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	CPI	11 101 101 10 100 001	ED A1	A−(HL) HL←HL+1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPIR	11 101 101 10 110 001	ED B1	A−(HL),HL←HL+1,BC←BC−1 Repeat until A=(HL) or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
	CPD	11 101 101 10 101 001	ED A9	A−(HL) HL←HL−1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPDR	11 101 101 10 111 001	ED B9	A−(HL),HL←HL−1,BC←BC−1 Repeat until A=(HL)or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
8-BIT ARITHMETIC AND LOGICAL	ADD A,r	10 000 rrr	80+r	A←A+r	*	*	X	*	X	V	0	*	1	4	
	ADD A,n	11 000 110 nn nnn nnn	C6 n	A←A+n	*	*	X	*	X	V	0	*	2	7	
	ADD A,(HL)	10 000 110	86	A←A+(HL)	*	*	X	*	X	V	0	*	2	7	
	ADD A,(IX+d)	11 011 101 10 000 110 dd ddd ddd	DD 86 d	A←A+(IX+d)	*	*	X	*	X	V	0	*	5	19	
	ADD A,(IY+d)	11 111 101 10 000 110 dd ddd ddd	FD 86 d	A←A+(IY+d)	*	*	X	*	X	V	0	*	5	19	
	ADC A,r	10 001 rrr	88+r	A←A+r+CY	*	*	X	*	X	V	0	*	1	4	
	ADC A,n	11 001 110 nn nnn nnn	CE n	A←A+n+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(HL)	10 001 110	8E	A←A+(HL)+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(IX+d)	11 011 101 10 001 110 dd ddd ddd	DD 8E d	A←A+(IX+d)+CY	*	*	X	*	X	V	0	*	5	19	
	ADC A,(IY+d)	11 111 101 10 001 110 dd ddd ddd	FD 8E d	A←A+(IY+d)+CY	*	*	X	*	X	V	0	*	5	19	
	SUB r	10 010 rrr	90+r	A←A−r	*	*	X	*	X	V	1	*	1	4	
	SUB n	11 010 110 nn nnn nnn	D6 n	A←A−n	*	*	X	*	X	V	1	*	2	7	
	SUB (HL)	10 010 110	96	A←A−(HL)	*	*	X	*	X	V	1	*	2	7	
	SUB (IX+d)	11 011 101 10 010 110 dd ddd ddd	DD 96 d	A←A−(IX+d)	*	*	X	*	X	V	1	*	5	19	
	SUB (IY+d)	11 111 101 10 010 110 dd ddd ddd	FD 96 d	A←A−(IY+d)	*	*	X	*	X	V	1	*	5	19	

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : *M P/V flag is 0 if the result of BC − 1 = 0, otherwise P/V = 1.
*N Z flag is 1 if A = (HL), otherwise Z = 0.
[ ]indicates the total condition of the number of cycles and states indicated by arrow.
r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C011A Instruction Set (4/9)

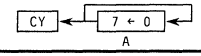
ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT ARITHMETIC AND LOGICAL	SBC A,r	10 011 rrr	98+r	A←A−r−CY	*	*	X	*	X	V	1	*	1	4
	SBC A,n	11 011 110 nn nnn nnn	DE n	A←A−n−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(HL)	10 011 110	9E	A←A−(HL)−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(IX+d)	11 011 101 10 011 110 dd ddd ddd	DD 9E d	A←A−(IX+d)−CY	*	*	X	*	X	V	1	*	5	19
	SBC A,(IY+d)	11 111 101 10 011 110 dd ddd ddd	FD 9E d	A←A−(IY+d)−CY	*	*	X	*	X	V	1	*	5	19
	AND r	10 100 rrr	A0+r	A←A∧r	*	*	X	1	X	P	0	0	1	4
	AND n	11 100 110 nn nnn nnn	E6 n	A←A∧n	*	*	X	1	X	P	0	0	2	7
	AND (HL)	10 100 110	A6	A←A∧(HL)	*	*	X	1	X	P	0	0	2	7
	AND (IX+d)	11 011 101 10 100 110 dd ddd ddd	DD A6 d	A←A∧(IX+d)	*	*	X	1	X	P	0	0	5	19
	AND (IY+d)	11 111 101 10 100 110 dd ddd ddd	FD A6 d	A←A∧(IY+d)	*	*	X	1	X	P	0	0	5	19
	OR r	10 110 rrr	B0+r	A←A∨r	*	*	X	0	X	P	0	0	1	4
	OR n	11 110 110 nn nnn nnn	F6 n	A←A∨n	*	*	X	0	X	P	0	0	2	7
	OR (HL)	10 110 110	B6	A←A∨(HL)	*	*	X	0	X	P	0	0	2	7
	OR (IX+d)	11 011 101 10 110 110 dd ddd ddd	DD B6 d	A←A∨(IX+d)	*	*	X	0	X	P	0	0	5	19
	OR (IY+d)	11 111 101 10 110 110 dd ddd ddd	FD B6 d	A←A∨(IY+d)	*	*	X	0	X	P	0	0	5	19
	XOR r	10 101 rrr	A8+r	A←A∀r	*	*	X	0	X	P	0	0	1	4
	XOR n	11 101 110 nn nnn nnn	EE n	A←A∀n	*	*	X	0	X	P	0	0	2	7
	XOR (HL)	10 101 110	AE	A←A∀(HL)	*	*	X	0	X	P	0	0	2	7
	XOR (IX+d)	11 011 101 10 101 110 dd ddd ddd	DD AE d	A←A∀(IX+d)	*	*	X	0	X	P	0	0	5	19
	XOR (IY+d)	11 111 101 10 101 110 dd ddd ddd	FD AE d	A←A∀(IY+d)	*	*	X	0	X	P	0	0	5	19
	CP r	10 111 rrr	B8+r	A−r	*	*	X	*	X	V	1	*	1	4
	CP n	11 111 110 nn nnn nnn	FE n	A−n	*	*	X	*	X	V	1	*	2	7
	CP (HL)	10 111 110	BE	A−(HL)	*	*	X	*	X	V	1	*	2	7
	CP (IX+d)	11 011 101 10 111 110 dd ddd ddd	DD BE d	A−(IX+d)	*	*	X	*	X	V	1	*	5	19
	CP (IY+d)	11 111 101 10 111 110 dd ddd ddd	FD BE d	A−(IY+d)	*	*	X	*	X	V	1	*	5	19
	INC r	00 rrr 100	04+r×8	r←r+1	*	*	X	*	X	V	0	−	1	4
	INC (HL)	00 110 100	34	(HL)←(HL)+1	*	*	X	*	X	V	0	−	3	11
	INC (IX+d)	11 011 101 00 110 100 dd ddd ddd	DD 34 d	(IX+d)←(IX+d)+1	*	*	X	*	X	V	0	−	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C011A Instruction Set (5/9)

ITEM/CLASSIFICATION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CYCLES	No. OF STATES
8-BIT ARITHMETIC AND LOGICAL	INC (IY+d)	11 111 101 00 110 100 dd ddd ddd	FD 34 d	(IY+d)←(IY+d)+1	*	*	X	*	X	V	0	–	6	23
	DEC r	00 rrr 101	05+r×8	r←r−1	*	*	X	*	X	V	1	–	1	4
	DEC (HL)	00 110 101	35	(HL)←(HL)−1	*	*	X	*	X	V	1	–	3	11
	DEC (IX+d)	11 011 101 00 110 101 dd ddd ddd	DD 35 d	(IX+d)←(IX+d)−1	*	*	X	*	X	V	1	–	6	23
	DEC (IY+d)	11 111 101 00 110 101 dd ddd ddd	FD 35 d	(IY+d)←(IY+d)−1	*	*	X	*	X	V	1	–	6	23
GENELAL-PURPOSE ARITHMETIC AND MPU CONTROL	DAA	00 100 111	27	Decimal adjust accumulator	*	*	X	*	X	P	–	*	1	4
	CPL	00 101 111	2F	A←$\overline{A}$	–	–	X	1	X	–	1	–	1	4
	NEG	11 101 101 01 000 100	ED 44	A←0−A	*	*	X	*	X	V	1	*	2	8
	CCF	00 111 111	3F	CY←$\overline{CY}$	–	–	X	X	X	–	0	*	1	4
	SCF	00 110 111	37	CY←1	–	–	X	0	X	–	0	1	1	4
	NOP	00 000 000	00	no operation	–	–	X	–	X	–	–	–	1	4
	HALT	01 110 110	76	MPU Halted	–	–	X	–	X	–	–	–	1	4
	DI	11 110 011	F3	IFF←0	–	–	X	–	X	–	–	–	1	4
	EI	11 111 011	FB	IFF←1	–	–	X	–	X	–	–	–	1	4
	IM 0	11 101 101 01 000 110	ED 46	Set interrupt mode 0	–	–	X	–	X	–	–	–	2	8
	IM 1	11 101 101 01 010 110	ED 56	Set interrupt mode 1	–	–	X	–	X	–	–	–	2	8
	IM 2	11 101 101 01 011 110	ED 5E	Set interrupt mode 2	–	–	X	–	X	–	–	–	2	8
16-BIT ARITHMETIC	ADD HL,t	00 tt1 001	09+t×10	HL←HL+t	–	–	X	X	X	–	0	*	3	11
	ADC HL,t	11 101 101 01 tt1 010	ED 4A+t×10	HL←HL+t+CY	*	*	X	X	X	V	0	*	4	15
	SBC HL,t	11 101 101 01 tt0 010	ED 42+t×10	HL←HL−t−CY	*	*	X	X	X	V	1	*	4	15
	ADD IX,p	11 011 101 00 pp1 001	DD 09+p×10	IX←IX+p	–	–	X	X	X	–	0	*	4	15
	ADD IY,s	11 111 101 00 ss1 001	FD 09+s×10	IY←IY+s	–	–	X	X	X	–	0	*	4	15
	INC t	00 tt0 011	03+t×10	t←t+1	–	–	X	–	X	–	–	–	1	6
	INC IX	11 011 101 00 100 011	DD 23	IX←IX+1	–	–	X	–	X	–	–	–	2	10
	INC IY	11 111 101 00 100 011	FD 23	IY←IY+1	–	–	X	–	X	–	–	–	2	10
	DEC t	00 tt1 011	0B+t×10	t←t−1	–	–	X	–	X	–	–	–	1	6
	DEC IX	11 011 101 00 101 011	DD 2B	IX←IX−1	–	–	X	–	X	–	–	–	2	10
	DEC IY	11 111 101 00 101 011	FD 2B	IY←IY−1	–	–	X	–	X	–	–	–	2	10
ROTATE	RLCA	00 000 111	07		–	–	X	0	X	–	0	*	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

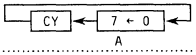
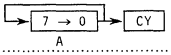
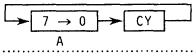
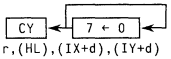
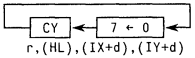
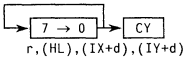
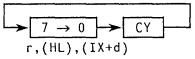
p	pp
BC	00
DE	01
IX	10
SP	11

s	ss
BC	00
DE	01
IY	10
SP	11

Note : ss is any of the register pairs BC, DE, HL, SP. PP is any of the register pairs BC, DE, IX, SP.
rr is any of the register pairs BC, DE, IY, SP.

120489

TMPZ84C011A Instruction Set (6/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RLA	00 010 111	17		–	–	X	0	X	–	0	*	1	4
	RRCA	00 001 111	0F		–	–	X	0	X	–	0	*	1	4
	RRA	00 011 111	1F		–	–	X	0	X	–	0	*	1	4
	RLC r	11 001 011 00 000 rrr	CB 00+r		*	*	X	0	X	P	0	*	2	8
	RLC (HL)	11 001 011 00 000 110	CB 06		*	*	X	0	X	P	0	*	4	15
	RLC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 000 110	DD CB d 06		*	*	X	0	X	P	0	*	6	23
	RLC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 000 110	FD CB d 06		*	*	X	0	X	P	0	*	6	23
	RL r	11 001 011 00 010 rrr	CB 10+r		*	*	X	0	X	P	0	*	2	8
	RL (HL)	11 001 011 00 010 110	CB 16		*	*	X	0	X	P	0	*	4	15
	RL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 010 110	DD CB d 16		*	*	X	0	X	P	0	*	6	23
	RL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 010 110	FD CB d 16		*	*	X	0	X	P	0	*	6	23
	RRC r	11 001 011 00 001 rrr	CB 08+r		*	*	X	0	X	P	0	*	2	8
	RRC (HL)	11 001 011 00 001 110	CB 0E		*	*	X	0	X	P	0	*	4	15
	RRC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 001 110	DD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RRC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 001 110	FD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RR r	11 001 011 00 011 rrr	CB 18+r		*	*	X	0	X	P	0	*	2	8
	RR (HL)	11 001 011 00 011 110	CB 1E		*	*	X	0	X	P	0	*	4	15
	RR (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 011 110	DD CB d 1E		*	*	X	0	X	P	0	*	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C011A Instruction Set (7/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RR (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 011 110	FD CB d 1E	7 → 0 → CY (IY+d)	*	*	X	0	X	P	0	*	6	23
	SLA r	11 001 011 00 100 rrr	CB 20+r		*	*	X	0	X	P	0	*	2	8
	SLA (HL)	11 001 011 00 100 110	CB 26		*	*	X	0	X	P	0	*	4	15
	SLA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 100 110	DD CB d 26	CY ← 7 ← 0 ← 0 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	SLA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 100 110	FD CB d 26		*	*	X	0	X	P	0	*	6	23
	SRA r	11 001 011 00 101 rrr	CB 28+r		*	*	X	0	X	P	0	*	2	8
	SRA (HL)	11 001 011 00 101 110	CB 2E		*	*	X	0	X	P	0	*	4	15
	SRA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 101 110	DD CB d 2E	7 → 0 → CY r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	SRA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 101 110	FD CB d 2E		*	*	X	0	X	P	0	*	6	23
	SRL r	11 001 011 00 111 rrr	CB 38+r		*	*	X	0	X	P	0	*	2	8
	SRL (HL)	11 001 011 00 111 110	CB 3E		*	*	X	0	X	P	0	*	4	15
	SRL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 111 110	DD CB d 3E	0 → 7 → 0 → CY r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	SRL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 111 110	FD CB d 3E		*	*	X	0	X	P	0	*	6	23
	RLD	11 101 101 01 101 111	ED 6F	A 7 4 3 0 ⇄ 7 4 3 0 (HL)	*	*	X	0	X	P	0	–	5	18
	RRD	11 101 101 01 100 111	ED 67	A 7 4 3 0 ⇄ 7 4 3 0 (HL)	*	*	X	0	X	P	0	–	5	18
BIT SET RESET AND TEST	BIT b,r	11 001 011 01 bbb rrr	CB 40+b×8+r	$Z \leftarrow \overline{r_b}$	X	*	X	1	X	X	0	–	2	8
	BIT b,(HL)	11 001 011 01 bbb 110	CB 46+b×8	$Z \leftarrow \overline{(HL)_b}$	X	*	X	1	X	X	0	–	3	12

*1

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Note : *1 : Rotate digit left and right between the accmulator and location (HL).
The content of the upper half of the accumulator is unaffected.
The notation $(HL)_b$ indicates bit $_b$ (0 to 7) within the contents of the HL register pair.
The notation r_b indicates bit $_b$ (0 to 7) within the r register.

120489

TMPZ84C011A Instruction Set (8/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
BIT SET RESET AND TEST	BIT b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 01 bbb 110	DD CB d 46+b×8	$Z \leftarrow \overline{(IX+d)_b}$	X	*	X	1	X	X	0	–	5	20
	BIT b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 01 bbb 110	FD CB d 46+b×8	$Z \leftarrow \overline{(IY+d)_b}$	X	*	X	1	X	X	0	–	5	20
	SET b,r	11 001 011 11 bbb rrr	CB C0+b×8+r	$r_b \leftarrow 1$	–	–	X	–	X	–	–	–	2	8
	SET b,(HL)	11 001 011 11 bbb 110	CB C6+b×8	$(HL)_b \leftarrow 1$	–	–	X	–	X	–	–	–	4	15
	SET b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 11 bbb 110	DD CB d C6+b×8	$(IX+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
	SET b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 11 bbb 110	FD CB d C6+b×8	$(IY+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
	RES b,r	11 001 011 10 bbb rrr	CB 80+b×8+r	$r_b \leftarrow 0$	–	–	X	–	X	–	–	–	2	8
	RES b,(HL)	11 001 011 10 bbb 110	CB 86+b×8	$(HL)_b \leftarrow 0$	–	–	X	–	X	–	–	–	4	15
	RES b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 10 bbb 110	DD CB d 86+b×8	$(IX+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
	RES b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 10 bbb 110	FD CB d 86+b×8	$(IY+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
JUMP	JP mn	11 000 011 nn nnn nnn mm mmm mmm	C3 n m	PC←mn	–	–	X	–	X	–	–	–	3	10
	JP c,mn	11 ccc 010 nn nnn nnn mm mmm mmm	C2+c×8 n m	PC←mn (Only when condition is met)	–	–	X	–	X	–	–	–	3	10
	JR $+e	00 011 000 aa aaa aaa	18 a	PC←$+e	–	–	X	–	X	–	–	–	3	12
	JR C,$+e	00 111 000 aa aaa aaa	38 a	If C=0, continue If C=1, PC←$+e	– –	– –	X X	– –	X X	– –	– –	– –	2 3	7 12
	JR NC,$+e	00 110 000 aa aaa aaa	30 a	If C=0, PC←$+e If C=1, continue	– –	– –	X X	– –	X X	– –	– –	– –	3 2	12 7
	JR Z,$+e	00 101 000 aa aaa aaa	28 a	If Z=0, continue If Z=1, PC←$+e	– –	– –	X X	– –	X X	– –	– –	– –	2 3	7 12
	JR NZ,$+e	00 100 000 aa aaa aaa	20 a	If Z=0, PC←$+e If Z=1, continue	– –	– –	X X	– –	X X	– –	– –	– –	3 2	12 7
	DJNZ $+e	00 010 000 aa aaa aaa	10 a	B←B−1, If B=0, continue B←B−1, If B<>0, continue	– –	– –	X X	– –	X X	– –	– –	– –	2 3	8 13
	JP (HL)	11 101 001	E9	PC←HL	–	–	X	–	X	–	–	–	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

e represents the extension in the relative addressing mode, a = e − 2. e is a signed two's complement number in the range of $-126 \leq e \leq 129$

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign Negative

Note :
- a = e − 2 in the Opcode provides an effective address of PC + e as PC is incremented by 2 before the addition of e.
- $ indicates the reference to the location counter value of the current segment.
- The notation $(HL)_b$, $(IX + d)_b$ indicates bit b (0 to 7) within the contents of the register pair.
- The notation r_b indicates bit b (0 to 7) within the r register.
- a = e-2 in the op-code provides effective address of PC + e as PC is incremented by 2 prior to the addition of e.

270489

TMPZ84C011A Instruction Set (9/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
JUMP	JP (IX)	11 011 101 11 101 001	DD E9	PC←(IX)	–	–	X	–	X	–	–	–	2	8
	JP (IY)	11 111 101 11 101 001	FD E9	PC←(IY)	–	–	X	–	X	–	–	–	2	8
CALL AND RETURN	CALL mn	11 001 101 nn nnn nnn mm mmm mmm	CD n m	(SP–1)←PC_H, (SP–2)←PC_L PC←mn SP←SP–2	–	–	X	–	X	–	–	–	5	17
	CALL c,mn	11 ccc 100 nn nnn nnn mm mmm mmm	C4+cX8 n m	If condition c is met, same as CALL mn. If condition c is not met,continue	– –	– –	X X	– –	X X	– –	– –	– –	5 3	17 10
	RET	11 001 001	C9	PC_L←(SP), PC_H←(SP+1) SP←SP+2	–	–	X	–	X	–	–	–	3	10
	RET c	11 ccc 000	C0+c×8	If condition c is met, same as RET. If condition c is not met,continue	– –	– –	X X	– –	X X	– –	– –	– –	3 1	11 5
	RETI	11 101 101 01 001 101	ED 4D	Return from interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RETN	11 101 101 01 000 101	ED 45	Return from non-maskable interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RST j	11 kkk 111	C7+k×8	(SP–1)←PC_H, (SP–2)←PC_L PC_H←0, PC_L←j, SP←SP–2	–	–	X	–	X	–	–	–	3	11
INPUT AND OUTPUT	IN A,(n)	11 011 011 nn nnn nnn	DB n	A←(n) n→A0~A7, A→A8~A15	–	–	X	–	X	–	–	–	3	11
	IN r,(C)	11 101 101 01 rrr 000	ED 40+rx8	r←(C) If r=110, only the flags will be affected.	*	*	X	*	X	P	0	–	3	12
	INI	11 101 101 10 100 010	ED A2	(HL)←(C), B←B–1, HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	INIR	11 101 101 10 110 010	ED B2	(HL)←(C), B←B–1, HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	IND	11 101 101 10 101 010	ED AA	(HL)←(C), B←B–1, HL←HL–1	X	*M	X	X	X	X	1	X	4	16
	INDR	11 101 101 10 111 010	ED BA	(HL)←(C), B←B–1, HL←HL–1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUT (n),A	11 010 011 nn nnn nnn	D3 n	(n)←A n→A0~A7, A→A8~A15	–	–	X	–	X	–	–	–	3	11
	OUT (C),r	11 101 101 01 rrr 001	ED 41+rX8	(C)←r	–	–	X	–	X	–	–	–	3	12
	OUTI	11 101 101 10 100 011	ED A3	(C)←(HL), B←B–1, HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	OTIR	11 101 101 10 110 011	ED B3	(C)←(HL), B←B–1, HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUTD	11 101 101 10 101 011	ED AB	(C)←(HL), B←B–1, HL←HL–1	X	*M	X	X	X	X	1	X	4	16
	OTDR	11 101 101 10 111 010	ED BB	(C)←(HL), B←B–1, HL←HL–1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16

j	kkk
00H	000
08H	001
10H	010
18H	011
20H	100
28H	101
30H	110
38H	111

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

←[B<>0] ←[B=0] *1

←[B<>0] ←[B=0]

←B[<>0] ←[B=0] *1

←[B<>0] ←[B=0]

Note :
- *M If the result of B – 1 is zero, the Z flag is set, otherwise it is reset.
- A0 through A15 indicate the address bus.
- [ ] indicates the total condition of the number of cycles and states indicated by arrow.

*1 C→A0~A7
B→A8~A15

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-Carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign negative

120489

TMPZ84C011A Instruction Map (1/7)

MPU Instruction Table (I)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	LD BC, mn	LD (BC), A	INC BC	INC B	DEC B	LD B, n	RLCA	EX AF,AF	ADD HC,BC	LD A, (BC)	DEC BC	INC C	DEC C	LD C, n	RRCA
1	DJNZ e	LD DE, mn	LD (DE), A	INC DE	INC D	DEC D	LD D,N	RLA	JE e	ADD HL,DE	LD A, (DE)	DEC DE	INC E	DEC E	LD E, n	RRA
2	JR NZ, e	LD HL, mn	LD (mn),HL	INC HL	INC D	DEC H	LD H, n	DAA	JR Z e	ADD HL,HL	LD HL,(mn)	DEC HL	INC L	DEC L	LD L, n	CPL
3	JR NC, e	LD SP, mn	LD (mn),A	INC SP	INC (HL)	DEC (HL)	LD (HL), n	SCF	JR C e	ADD HL,SP	LD A,(mn)	DEC SP	INC A	DEC A	LD A, n	CCF
4	LD B, B	LD B, C	LD B, D	LD B, E	LD B, H	LD B, L	LD B, (HL)	LD B, A	LD C, B	LD C, C	LD C, D	LD C, E	LD C, H	LD C, L	LD C, (HL)	LD C, A
5	LD D, B	LD D, C	LD D, D	LD D, E	LD D, H	LD D, L	LD D, (HL)	LD D, A	LD E, B	LD E, C	LD E, D	LD E, E	LD E, H	LD E, L	LD E, (HL)	LD E, A
6	LD H, B	LD H, C	LD H, D	LD H, E	LD H, H	LD H, L	LD H, (HL)	LD H, A	LD L, B	LD L, C	LD L, D	LD L, E	LD L, H	LD L, L	LD L, (HL)	LD L, A
7	LD (HL), B	LD (HL), C	LD (HL), D	LD (HL), E	LD (HL), H	LD (HL), L	HALT	LD (HL), A	LD A, B	LD A, C	LD A, D	LD A, E	LD A, H	LD A, L	LD A, (HL)	LD A, A
8	ADD A, B	ADD A, C	ADD A, D	ADD A, E	ADD A, H	ADD A, L	ADD A, (HL)	ADD A, A	ADC A, B	ADC A, C	ADC A, D	ADC A, E	ADC A, H	ADC A, L	ADC A, (HL)	ADC A, A
9	SUB B	SUB C	SUB D	SUB E	SUB H	SUB L	SUB (HL)	SUB A	SBC A, B	SBC A, C	SBC A, D	SBC A, E	SBC A, H	SBC A, L	SBC A, (HL)	SBC A, A
A	AND B	AND C	AND D	AND E	AND H	AND L	AND (HL)	AND A	XOR B	XOR C	XOR D	XOR E	XOR H	XOR L	XOR (HL)	XOR A
B	OR B	OR C	OR D	OR E	OR H	OR L	OR (HL)	OR A	CP B	CP C	CP D	CP E	CP H	CP L	CP (HL)	XOR A
C	RET NZ	POP BC	JP NZ, mn	JP mn	CALL NZ, mn	PUSH BC	ADD A, n	RST 00H	RET C	RET	JP Z, mn	①	CALL Z, mn	CALL mn	ADC A, n	RST 08H
D	RET NC	POP DE	JP NC, mn	OUT (n), A	CALL NC, mn	PUSH DE	SUB n	RST 10H	RET C	EXX	JP C, mn	IN A, (n)	CALL C, mn	③	SBC A, n	RST 18H
E	RET PO	POP HL	JP PO, mn	EX (SP),HL	CALL PO, mn	PUSH HL	AND n	RST 20H	RET PE	JP (HL)	JP PE, mn	EX DE,HL	CALL PI, mn	②	XOR n	RST 28H
F	RET P	POP AF	JP P, mn	DI	CALL P, mn	PUSH AF	OR n	RST 30H	RET M	LD SP, HL	JP M, mn	EI	CALL M, mn	④	CP n	RST 38H

120489

Note ①~④: Multi-Opcode Instructions (ref. Table (II)~(VII))

TMPZ84C011A Instruction Map (2/7)

① Byte 1 "CB"

Instruction Table (II) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	RLC B	RLC C	RLC D	RLC E	RLC H	RLC L	RLC (HL)	RLC A	RRC B	RRC C	RRC D	RRC E	RRC H	RRC L	RRC (HL)	RRC A
1	RL B	RL C	RL D	RL E	RL H	RL L	RL (HL)	RL A	RR B	RR C	RR D	RR E	RR H	RR L	RR (HL)	RR A
2	SLA B	SLA C	SLA D	SLA E	SLA H	SLA L	SLA (HL)	SLA A	SRA B	SRA C	SRA D	SRA E	SRA H	SRA L	SRA (HL)	SRA A
3									SRL B	SRL C	SRL D	SRL E	SRL H	SRL L	SRL (HL)	SRL A
4	BIT 0, B	BIT 0, C	BIT 0, D	BIT 0, E	BIT 0, H	BIT 0, L	BIT 0,(HL)	BIT 0, A	BIT 1, B	BIT 1, C	BIT 1, D	BIT 1, E	BIT 1, H	BIT 1, L	BIT 1, (HL)	BIT 1, A
5	BIT 2, B	BIT 2, C	BIT 2, D	BIT 2, E	BITT 2, H	BIT 2, L	BIT 2, (HL)	BIT 2, A	BIT 3, B	BIT 3, C	BIT 3, D	BIT 3, E	BIT 3, H	BIT 3, L	BIT 3, (HL)	BIT 3, A
6	BIT 4, B	BIT 4, C	BIT 4, D	BIT 4, E	BIT 4, H	BIT 4, L	BIT 4, (HL)	BIT 4, A	BIT 5, B	BIT 5, C	BIT 5, D	BIT 5, E	BIT 5, H	BIT 5, L	BIT 5, (HL)	BIT 5, A
7	BIT 6, B	BIT 6, C	BIT 6, D	BIT 6, E	BIT 6, H	BIT 6, L	BIT 6, (HL)	BIT 6, A	BIT 7, B	BIT 7, C	BIT 7, D	BIT 7, E	BIT 7, H	BIT 7, L	BIT 7, (HL)	BIT 7, A
8	RES 0, B	RES 0, C	RES 0, D	RES 0, E	RES 0, H	RES 0, L	RES 0, (HL)	RES 0, A	RES 1, B	RES 1, C	RES 1, D	RES 1, E	RES 1, H	RES 1, L	RES 1, (HL)	RES 1, A
9	RES 2, B	RES 2, C	RES 2, D	RES 2, E	RES 2, H	RES 2, L	RES 2, (HL)	RES 2, A	RES 3, B	RES 3, C	RES 3, D	RES 3, E	RES 3, H	RES 3, L	RES 3, (HL)	RES 3, A
A	RES 4, B	RES 4, C	RES 4, D	RES 4, E	RES 4, H	RES 4, L	RES 4, (HL)	RES 4, A	RES 5, B	RES 5, C	RES 5, D	RES 5, E	RES 5, H	RES 5, L	RES 5, (HL)	RES 5, A
B	RES 6, B	RES 6, C	RES 6, D	RES 6, E	RES 6, H	RES 6, L	RES 6, (HL)	RES 6, A	RES 7, B	RES 7, C	RES 7, D	RES 7, E	RES 7, H	RES 7, L	RES 7, (HL)	RES 7, A
C	SET 0, B	SET 0, C	SET 0, D	SET 0, E	SET 0, H	SET 0, L	SET 0, (HL)	SET 0, A	SET 1, B	SET 1, C	SET 1, D	SET 1, E	SET 1, H	SET 1, L	SET 1, (HL)	SET 1, A
D	SET 2, B	SET 2, C	SET 2, D	SET 2, E	SET 2, H	SET 2, L	SET 2, (HL)	SET 2, A	SET 3, B	SET 3, C	SET 3, D	SET 3, E	SET 3, H	SET 3, L	SET 3, (HL)	SET 3, A
E	SET 4, B	SET 4, C	SET 4, D	SET 4, E	SET 4, H	SET 4, L	SET 4, (HL)	SET 4, A	SET 5, B	SET 5, C	SET 5, D	SET 5, E	SET 5, H	SET 5, L	SET 5, (HL)	SET 5, A
F	SET 6, B	SET 6, C	SET 6, D	SET 6, E	SET 6, H	SET 6, L	SET 6, (HL)	SET 6, A	SET 7, B	SET 7, C	SET 7, D	SET 7, E	SET 7, H	SET 7, L	SET 7, (HL)	SET 7, A

120489

TMPZ84C011A Instruction Map (3/7)

② Byte 1 "ED"

Instruction Table (III) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4	IN B, (C)	OUT (C), B	SBC HL,BC	LD (mn),BC	NEG	RETN	IM0	LD I, A	IN C, (C)	OUT (C), C	ADC HL,BC	LD BC,(mn)		RETI		LD R, A
5	IN D, (C)	OUT (C), D	SBC HL,DE	LD (mn),DE			IM1	LD A, I	IN E, (C)	OUT (C), E	ADC HL,DE	LD DE,(mn)			IM2	LD A, R
6	IN H, (C)	OUT (C), H	SBC HL,HL	LD (mn),HL				RRD	IN L, (C)	OUT (C), L	ADC HL,HL	LD HL,(mn)				RLD
7			SBC HL,SP	LD (mn),SP					IN A, (C)	OUT (C), A	ADC HL,SP	LD SP,(mn)				
8																
9																
A	LDI	CPI	INI	OUTI					LDD	CPD	IND	OUTD				
B	LDIR	CPIR	INIR	OTIR					LDDR	CPDR	INDR	OTDR				
C																
D																
E																
F																

120489

TMPZ84C011A Instruction Map (4/7)

③ Byte 1 "DD"

Instruction Table (Ⅳ) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0										ADD IX,BC						
1										ADD IX,DE						
2		LD IX, mn	LD (mn), IX	INC IX						ADD IX, IX	LD IX,(mn)	DEC IX				
3					INC (IX + d)	DEC (IX + d)	LD (IX + d) , n			ADD IX, SP						
4							LD B, (IX + d)								LD C, (IX + d)	
5							LD D, (IX + d)								LD E, (IX + d)	
6							LD H, (IX + d)								LD L, (IX + d)	
7	LD (IX + d) , B	LD (IX + d) , C	LD (IX + d) , D	LD (IX + d) , E	LD (IX + d) , H	LD (IX + d) , L		LD (IX + d) ; A							LD A, (IX + d)	
8							ADD (IX + d)								ADC 1, (IX + d)	
9							SUB (IX + d)								SBC A, (IX + d)	
8							AND (IX + d)								XOR (IX + d)	
9							OR (IX + d)								CP (IX + d)	
C												⑤				
D																
E		POP IX		EX (SP), IX		PUSH IX				JP (IX)						
F										LD SP, IX						

120489

Note ⑤: Special 2 byte Opcode Instructions (ref. Table (Ⅵ))

TMPZ84C011A Instruction Map (5/7)

④ Byte 1 "FD"

Instruction Table (Ⅴ) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0										ADD IY,BC						
1										ADD IY,DE						
2		LD IY, nn	ID (mn), IY	INC IY						ADD IY, IY	LD IY, (mn)	DEC IY				
3					INC (IY + d)	DEC (IY + d)	LD (IY + d) , n			ADD IY, SP						
4							LD B, (IY + d)								LD C, (IY + d)	
5							LD D, (IY + d)								LD E, (IY + d)	
6							LD H, (IY + d)								LD L, (IY + d)	
7	LD (IY + d) , B	LD (IY + d) , C	LD (IY + d) , D	LD (IY + d) , E	LD (IY + d) , H	LD (IY + d) , L		LD (IY + d) , A							LD A, (IY + d)	
8							ADD (IY + d)								ADDC A, (IY + d)	
9							SUB (IY + d)								SUBC A, (IY + d)	
8							AND (IY + d)								XOR (IY + d)	
9							OR (IY + d)								CP (IY + d)	
C												⑥				
D																
E		POP IY		EX (SP), IY		PUSH IY				JP (IY)						
F										LD SP, IY						

120489

Note ⑥: Special 2 byte Opcode Instructions (ref. Table (Ⅶ))

TMPZ84C011A Instruction Map (6/7)

⑤ Byte 1 "DD"

Byte 2 "CB"

Instruction Table (VI) (Special case of 2-byte Opcode : Byte 3)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0							RLC (IX + d)								RRC (IX + d)	
1							RL (IX + d)								RR (IX + d)	
2							SLA (IX + d)								SRA (IX + d)	
3															SRL (IX + d)	
4							BIT 0, (IX + d)								BIT 1, (IX + d)	
5							BIT 2, (IX + d)								BIT 3, (IX + d)	
6							BIT 4, (IX + d)								BIT 5, (IX + d)	
7							BIT 6, (IX + d)								BIT 7, (IX + d)	
8							RES 0, (IX + d)								RES 1, (IX + d)	
9							RES 2, (IX + d)								RES 3, (IX + d)	
A							RES 4, (IX + d)								RES 5, (IX + d)	
B							RES 6, (IX + d)								RES 7, (IX + d)	
C							SET 0, (IX + d)								SET 1, (IX + d)	
D							SET 2, (IX + d)								SET 3, (IX + d)	
E							SET 4, (IX + d)								SET 5, (IX + d)	
F							SET 6, (IX + d)								SET 7, (IX + d)	

120489

TMPZ84C011A Instruction Map (7/7)

⑥ Byte 1 "FD"

Byte 2 "CB"

Instruction Table (VII) (Special case of 2-byte Opcode : Byte 3)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0							RLC (IY + d)								RRC (IY + d)	
1							RL (IY + d)								RR (IY + d)	
2							SLA (IY + d)								SRA (IY + d)	
3															SRL (IY + d)	
4							BIT 0, (IY + d)								BIT 1, (IY + d)	
5							BIT 2, (IY + d)								BIT 3, (IY + d)	
6							BIT 4, (IY + d)								BIT 5, (IY + d)	
7							BIT 6, (IY + d)								BIT 7, (IY + d)	
8							RES 0, (IY + d)								RES 1, (IY + d)	
9							RES 2, (IY + d)								RES 3, (IY + d)	
A							RES 4, (IY + d)								RES 5, (IY + d)	
B							RES 6, (IY + d)								RES 7, (IY + d)	
C							SET 0, (IY + d)								SET 1, (IY + d)	
D							SET 2, (IY + d)								SET 3, (IY + d)	
E							SET 4, (IY + d)								SET 5, (IY + d)	
F							SET 6, (IY + d)								SET 7, (IY + d)	

120489

3.3 CGC Operations

This subsection describes the system configuration, functions, and basic operations of the clock generator/controller (CGC).

3.3.1 Block Diagram

Figure 3.3.1 shows the block diagram of the CGC.

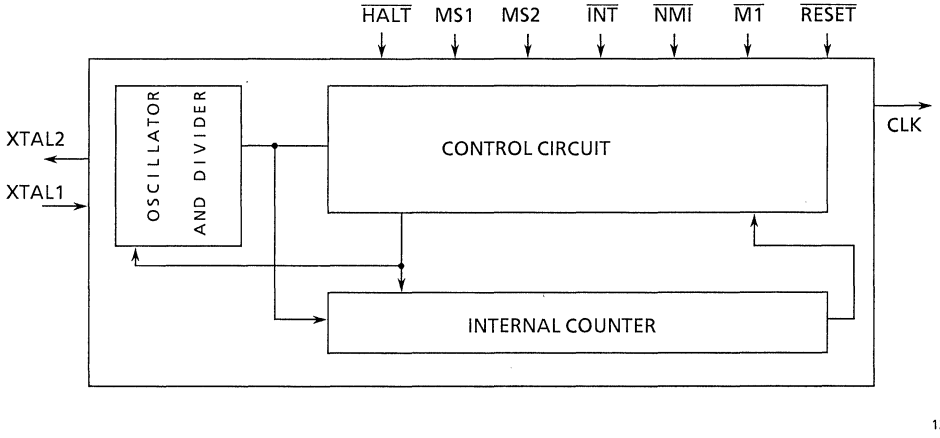


Figure 3.3.1 Block Diagram

3.3.2 CGC System Configuration

The internal configuration of the CGC is shown in Figure 3.3.1. The waveform from the external oscillated by the internal oscillator and divided by the divider is converted into the square waveform for clock. The clock is controlled by the controller and the counter to be sent to the outside the CGC. The following describes the CGC's main components and their functions.

(1) Clock Generation

(2) Operation Modes

[1] Clock Generation

The CGC contains an oscillator. By connecting oscillating elements to external pins (XTAL1 and XTAL2), the required clock can be produced easily. The CGC provides the clock whose frequency is 1/2 of the oscillation frequency. Figure 3.3.2 shows an example of crystal connection.

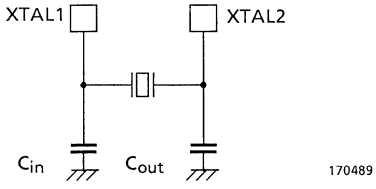


Figure 3.3.2 (a) Example of Crystal Connection

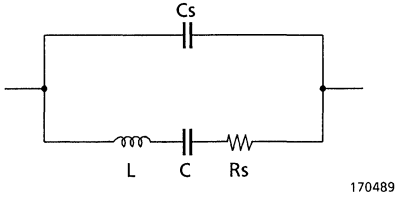


Figure 3.3.2 (b) Oscillator Equivalent Circuit

(1) For the quartz crystal oscillator, use the MR8000-C20 (oscillation frequency 8 MHz) or MR12000-C20 (oscillation frequency 12 MHz) manufactured by Tokyo Denpa Company Ltd., or the equivalent;

Product No.	Holder Type	Frequency (MHz)	Cin (pF)	Cout (pF)	Quartz Crystal Parameter (Typ.)			Drive Level (mW)	Condition
					C_1 (pF)	C_0 (pF)	R_1 (Ω)		Load Capacitance (pF)
MR8000-C20	HC-49 -U (TR-49)	8	22	33	—	4.00	30.0	—	—
MR8000-C14		8	20	20	0.0189	3.87	6.0	0.5	12.67
MR12000-C20		12	33	33	—	4.00	25.0	—	—
MR12000-C14		12	20	20	0.0190	3.81	6.9	0.5	12.55

310589

Note : The load capacitance in the condition does not include any stray capacitance.

(2) For the ceramic resonater, use the CSA8.00MT, CST8.00MT (oscillation frequency 8 MHz) or CSA12.00MT, CST12.00MT (osillation frequency 12 MHz) manufactured by Murata MFG Co., Ltd.

Product No.	Frequency (MHz)	Cin (pF)	Cout (pF)
CSA8.00MT	8	30	30
CST8.00MT	8		
CSA12.0MT	12	30	30
CST12.0MT	12		

190589

Note : The CST8.00MT and CST12.0MT need no outer capacitance.

[2] Operation Modes

The CGC has the capability to control 3 types of operational modes: Run, Idle and Stop. Anyone of them can be selected by MS1, MS2 (Mode Select 1 and 2) pin.

These modes become valid when the MPU executes a HALT instruction. Fetching a HALT instruction, the MPU sets the $\overline{\text{HALT}}$ signal to "0", telling that it has been put in the halt state. After the execution of the HALT instruction, CGC performs the operation of the specified mode. Table 3.3.1 shows the operations of these modes.

Table 3.3.1 CGC Operation Modes

MS1	MS2	Operation Mode	Description
0	0	Idle mode	Only the internal oscillator operates, stopping the supply of clock outside. The clock output (CLK) is held at "0".
0	1	——	Not used.
1	0	Stop mode	All internal operations are stopped. The clock output (CLK) is held at "0".
1	1	Run mode	The supply of clock outside is continued.

120489

The restart from the clock stop state in Idle or Stop mode is performed by reset ($\overline{\text{RESET}}$ signal) or acknowledge of maskable interrupt ($\overline{\text{INT}}$ signal) or non-maskable interrupt ($\overline{\text{NMI}}$ signal).

[3] Warm-up Time for Restart (from Stop mode)

Clearing the halt state by interrupt acknowledge, the MPU begins executing interrupt processing. Therefore, when restarting the clock by the $\overline{NMI}$ or $\overline{INT}$ restart signal in the Stop mode, its oscillation must be fully stabilized before being supplied outside. The CGC provides, by means of the internal counter, the warm-up time enough for the clock to reach the stabilize frequency. The warm-up ends at the rising edge of the internal counter output derived from dividing the oscillation frequency to start clock output. The warm-up time is equal to the time derived by dividing the frequency of the externally attached oscillator by 2^{14}.

Figure 3.3.3 shows the block diagram of the internal counter. Table 3.3.2 shows the relationship between the oscillation frequency and the warm-up time.

In the restart by the $\overline{RESET}$ signal, no warm-up is performed for the quick operation at power-up. Therefore, expand the width of the $\overline{RESET}$ signal adequately to provide the warm-up time.

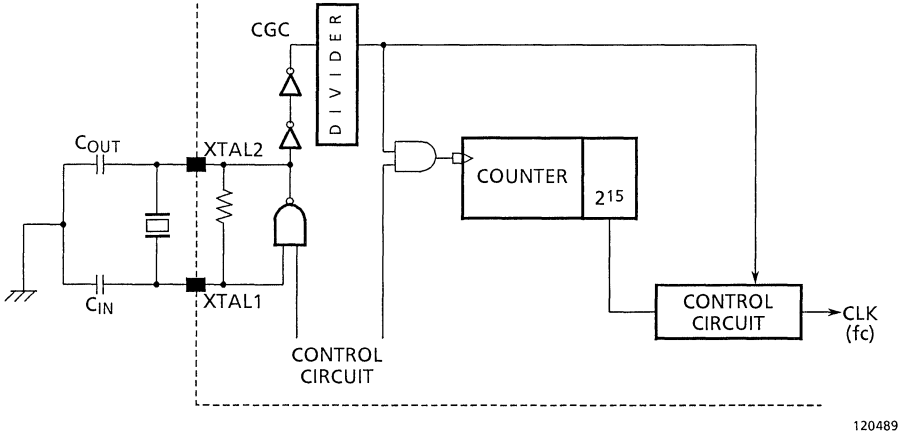


Figure 3.3.3 Block Diagram of Internal Counter

Table 3.3.2 Warm-up Time

Counter output	Warm-up time		
		fXTAL = 12MHz	fXTAL = 8MHz
2^{15}	2^{14}/fc	2.7 ms	4 ms

120489

* fc = $f_{XTAL}/2$

3.3.3 CGC Status Transition Diagram and Basic Timing

The following describes the status transition and basic timing to be provided when the CGC operates.

[1] Status Transition Diagram

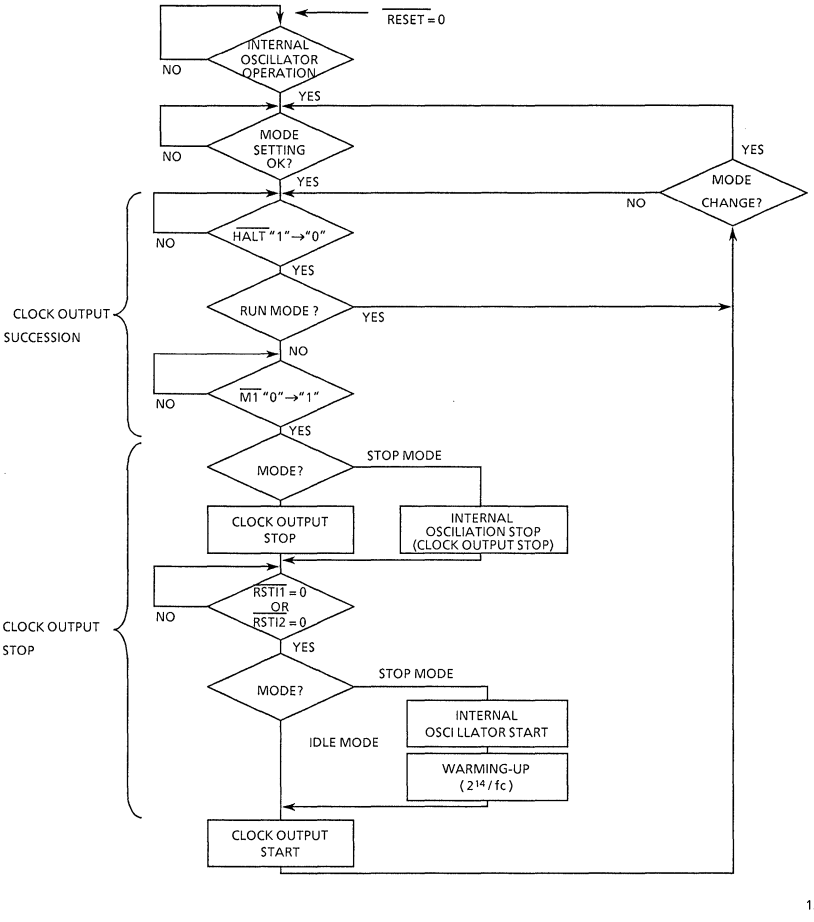


Figure 3.3.4 Status Transition Diagram

[2] Basic Timing

The following describes the CGC basic timing.

(1) Operation at execution of HALT instruction

The following describes the basic timing in each mode to be provided when the MPU executes a HALT instruction. The MPU sets the $\overline{HALT}$ signal to "0" in synchronization with the falling edge of clock state T4 of the HALT instruction Opcode fetch cycle (M1). This signal tells the CGC that the MPU is going to get in the halt state.

(a) Run mode (MS1 = 1, MS2 = 1)

Figure 3.3.5 shows the basic timing in the Run mode. In the Run mode, the CGC continues supplying the clock to the outside even when the MPU is in the halt state. Therefore, the MPU continues executing NOPs during the halt state. The systems which need memory address refresh use this mode.

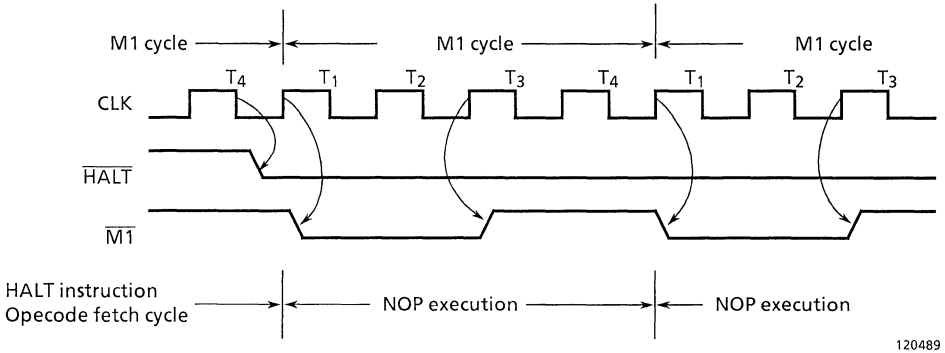


Figure 3.3.5 Basic Timing in Run Mode

(b) Idle mode (MS1 = MS2 = 0), and Stop mode (MS1 = 1, MS2 = 0)

Figure 3.3.6 shows the basic timing in the Idle and Stop modes. In these modes, the clock output is stopped with clock state T4 being "0" by the $\overline{HALT}$ signal and the $\overline{M1}$ signal which follows the HALT instruction.

However, in the Stop mode, the CGC's internal oscillator also stops.

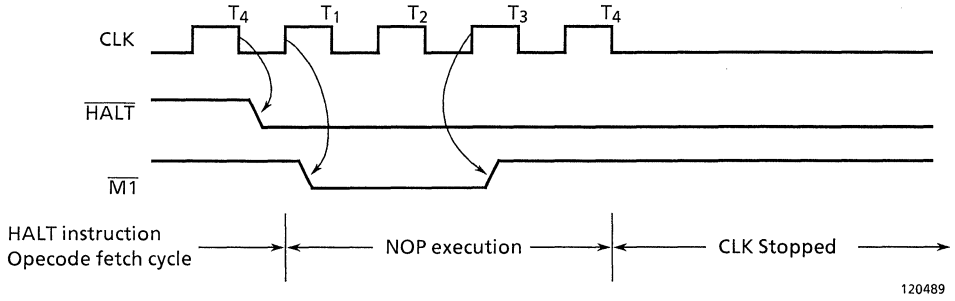


Figure 3.3.6 Basic Timing in Idle and Stop Modes

(2) Clock output restart from each mode

The clock stopped state in the Idle or Stop mode is cleared by setting any of the following signals to "0" (for the system restart operation, see Subsection 3.3.4) :

- $\overline{INT}$ (level trigger input)
- $\overline{NMI}$ (edge trigger input)
- $\overline{RESET}$ (level trigger input)

(a) Clock output restart from Idle mode

Figure 3.3.7 shows the basic timing for the sequence of the output restart from the clock stopped state in the Idle mode. In the restart in the Idle mode, the clock output is restarted in a relatively short delay time because the internal oscillator is operating even in the clock stopped state.

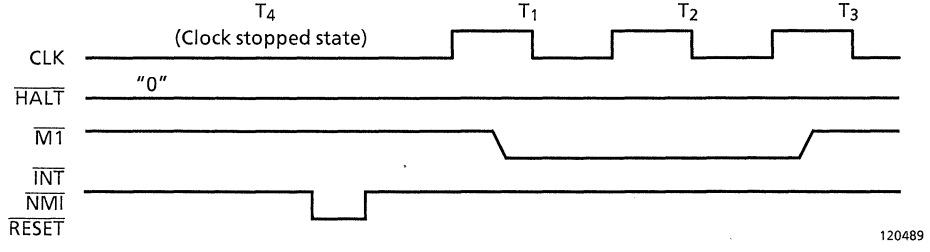


Figure 3.3.7 Basic Timing for Sequence of Restart from Clock stopped State (Idie Mode)

(b) Clock output restart from Stop mode

Figure 3.3.8 shows the basic timing for the sequence of the restart from the clock stopped state in the Stop mode. When restarting by setting the $\overline{INT}$ or $\overline{NMI}$ signal to "0", the warm-up time is automatically created by the internal counter. In the restart by the $\overline{RESET}$ signal. oscillation restarts without warm-up.

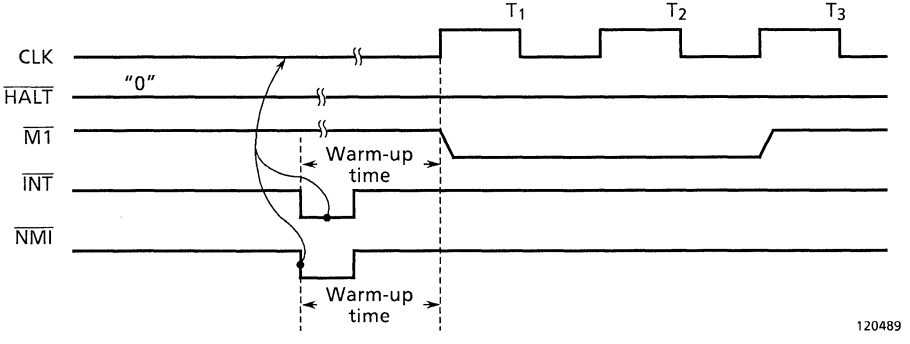


Figure 3.3.8 Basic Timing for Sequence of Restart from Clock Stopped State (Stop Mode)

3.3.4 Relationship with MPU

The following describes the relationship between the CGC and the MPU mainly in terms of the halt clear operation.

[1] RESET Signal

Figure 3.3.9 shows an example of the timing for the restart from the Stop mode on the TMPZ84C011A sharing the MPU and CGC $\overline{RESET}$ signals. To reset the MPU, the $\overline{RESET}$ signal must be set to "0" for at least 3 stable clocks. When the $\overline{RESET}$ signal goes "1", the MPU gets out of the halt state after a dummy cycle of 2 clock states to start executing instructions from address 0000H.

To restart the clock output by the $\overline{RESET}$ signal in the Stop mode, the internal counter to determine the warm-up time does not operate. Therefore, if the MPU does not restart correctly due to the unstable clock output immediately after the restart of the internal oscillator, or when the stability of the crystal at power-up is considered, the $\overline{RESET}$ signal must be held at "0" for a time long enough for the MPU to be reset securely.

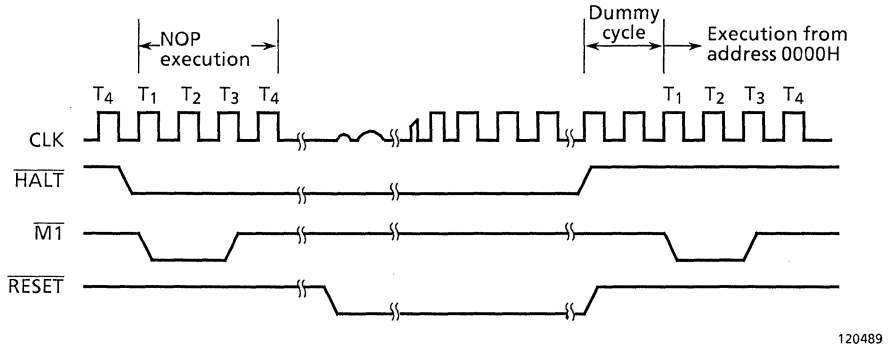


Figure 3.3.9 Example of Clock Restart Timing by $\overline{\text{RESET}}$ Signal

[2] Clearing Halt State by Interrupt Signal

The CGC restarts the clock output from the Idle or Stop mode by the input of $\overline{\text{INT}}$ or $\overline{\text{NMI}}$ signal. By this clock, the MPU starts operating. However, when the CGC restarts the clock output, the MPU is still in the halt state executing NOPs. To clear the halt state, the interrupt signal must be entered into the MPU (in the case of the $\overline{\text{INT}}$ signal, the signal for at least one instruction). The MPU interrupt is detected at the rising edge of the last clock of each instruction (NOP for the halt state).

(1) When using non-maskable interrupt (NMI)

MPU's non-maskable interrupt is triggered by edge. The MPU contains the flip-flop to detect an interrupt. That is, the state of this internal NMI flip-flop is sampled at the rising edge of the last clock of each instruction. Therefore, when a short active low ("0") pulse has been inserted before the interrupt detection timing, the interrupt is acknowledged. The $\overline{\text{NMI}}$ input of the TMPZ84C011A is connected to the $\overline{\text{NMI}}$ input of the MPU via the CGC, performing the same operations as above. (See Figure 3.3.11.)

(2) When using maskable interrupt (INT)

With a maskable interrupt, the maskable interrupt enable flip- flop (IFF) must be set to "1" by program before "0" of the $\overline{\text{INT}}$ input signal is detected. Even if the CGC accepts the $\overline{\text{INT}}$ signal to restart supply the clock, no interrupt is acknowledge unless the $\overline{\text{INT}}$ signal is kept inserted until one instruction (NOP) has been executed. Figure 3.3.10 shows the timing for clearing the halt state by the interrupt signal.

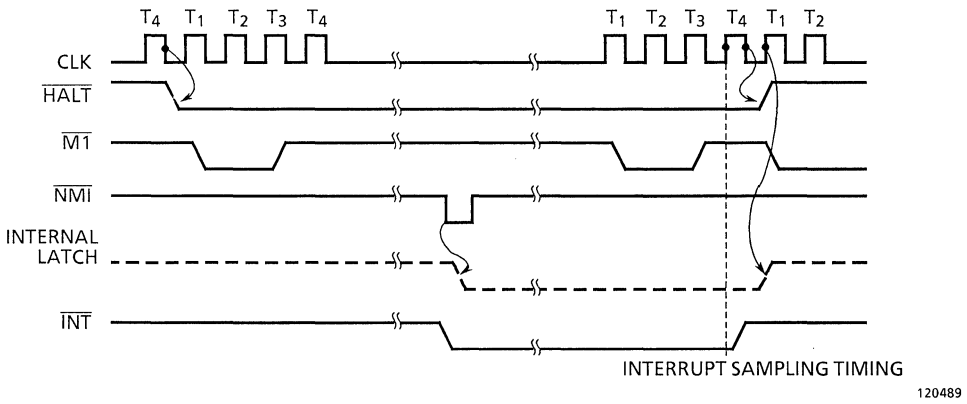


Figure 3.3.10 Timing for Clearing Halt State by Interrupt Signal

[3] Connecting CGC to MPU on TMPZ84C011A

Figure 3.3.11 shows the connection between the CGC and the MPU on the TMPZ84C011A.

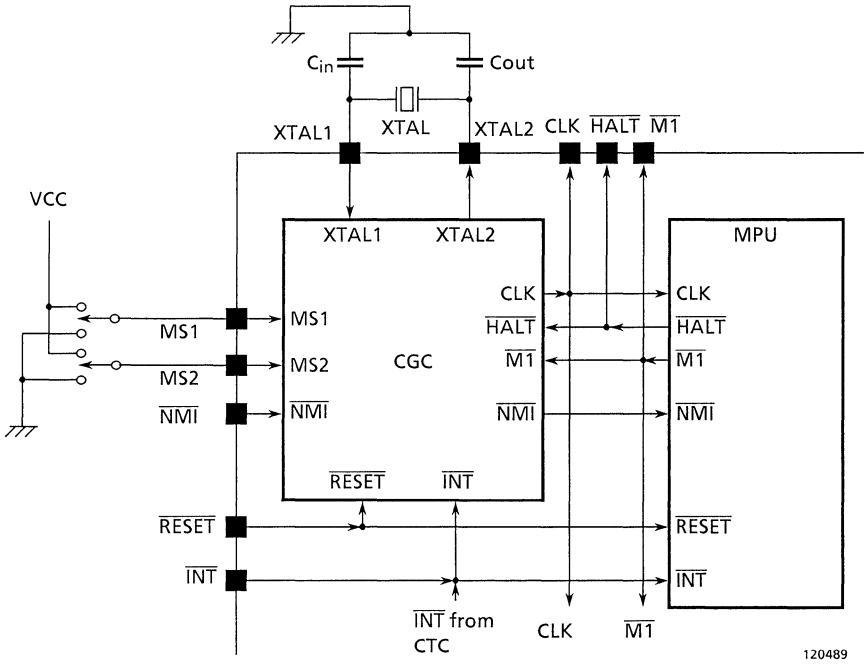


Figure 3.3.11 Connection Between CGC and MPU

3.4 CTC Operational Description

The CTC has 4 independent channels. To these channels, addresses are allocated on the TMPZ84C011A's I/O map, permitting the read/write of the channels in the MPU's I/O cycle. (See Figure 3.4.1.) This subsection mainly describes the operation to be performed after the CTC is accessed.

3.4.1 CTC Block Diagram

Figure 3.4.1 shows the block diagram of the CTC.

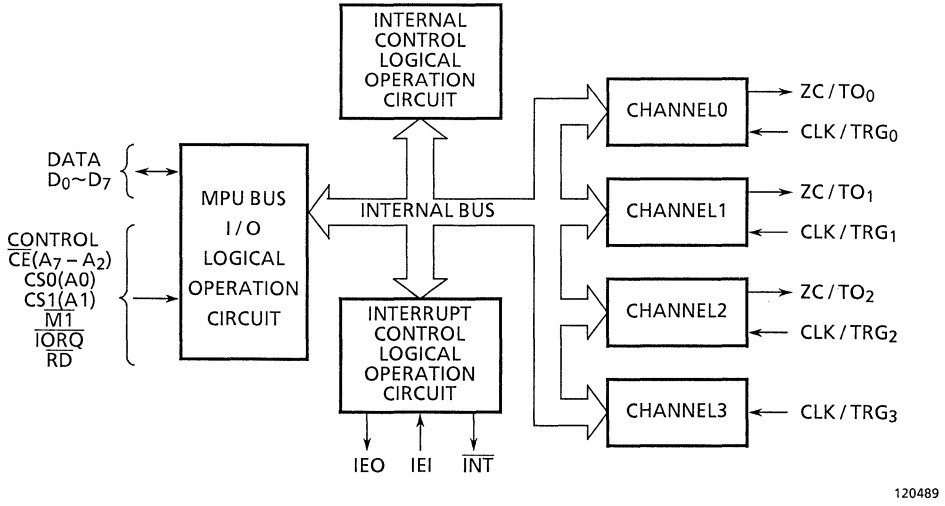


Figure 3.4.1 Block Diagram of CTC

3.4.2 CTC System Configuration

The CTC system consists of the following 4 logic circuits:

(1) MPU bus I/O logic circuit

(2) Internal control logic circuit

(3) Interrupt control logic circuit

(4) Four independent counter/timer channel logic circuits

[1] MPU Bus I/O Logic Circuit

This circuit transfer data between the MPU and the CTC.

[2] Internal Control Logic Circuit

This circuit controls the CTC's operation functions such as the CTC selector chip enable, reset, and read/write circuits.

[3] Interrupt Control Logic Circuit

This circuit performs the MPU interrupt related processing such as priority determination. The priority with other LSIs is determined by the physical location in daisy chain connection.

[4] Counter/Timer Channel Logic Circuit

This circuit consists of the following 2 registers and 2 counters. Figure 3.4.2 shows the configuration of this circuit.

- Time-constant register (8 bits)
- Channel control register (8 bits)
- Down-counter (8 bits)
- Prescaler (8 bits)

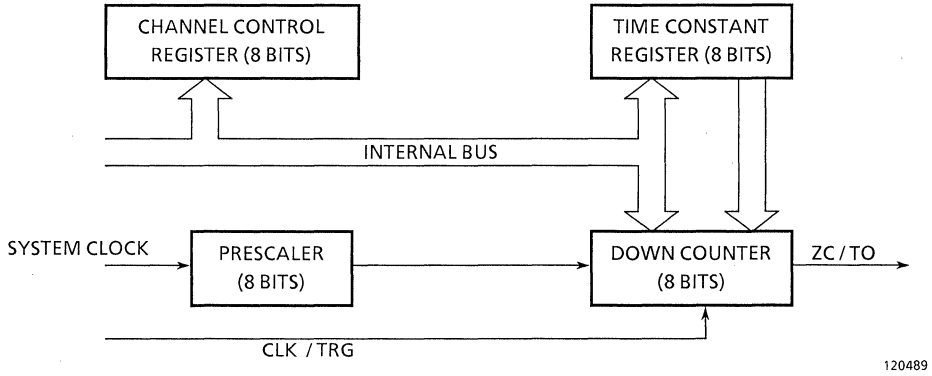


Figure 3.4.2 Configuration of Counter/Timer Channel Logic Circuit

(1) Time-constant register

This register holds the time constant to be written in the down counter. When the CTC is initialized or the down-counter has reached zero, the time constant is loaded into down- counter. The time constant is set immediately after the MPU has written the channel control word to the channel control register. For a time constant, an integer from 1 to 256 can be used.

(2) Channel control register

This register is used to choose the channel mode or condition according to the channel control word sent from the MPU.

(3) Down-counter

The contents of the time-constant register are loaded into the down counter. In the counter mode, these contents are decremented at each edge of the external clock; in the timer mode, they are decremented for each prescaler clock output. The contents of the time-constant register are loaded at initialization or when the down-counter has reached zero.
The contents of the down-counter can be read any time. Also, the system can be programmed so that an interrupt request is generated each time the down-counter has reached zero.

(4) Prescaler

The prescaler, used only in the timer mode, divides the system clock by a factor of 16 or 256. The prescaler factor can be programmed by channel control word. The output of the prescaler becomes the clock input to the down-counter.

3.4.3 CTC Basic Operations

[1] Reset

The state of the CTC is unstable after it is powered on. To initialize the CTC, the low level signal needs to be applied to the $\overline{\text{RESET}}$ pin. On any channel, the channel control word and time-constant data must be written to the channel to be started before it is started in the counter or timer mode. To program the system to enable interrupts, the interrupt vector word must be written to the interrupt controller. When these data have been written to the CTC, it is ready to start.

[2] Interrupt

The CTC can cause an interrupt when the MPU is operating in the mode 2. The CTC interrupt can be programmed for each channel. Each time the channel's down-counter has reached zero, the CTC outputs the interrupt request signal ($\overline{\text{INT}}$). When the MPU accepts the CTC's interrupt request, the CTC outputs the interrupt vector. Based on this interrupt vector, the MPU specifies the start address of the interrupt processing routine and calls it to start interrupt processing.

Because the MPU specifies the start address of the interrupt processing routine by the interrupt vector from the CTC, the user can change the vector value to call any desired address.

The interrupt processing is terminated when the MPU executes an RETI instruction. The CTC has the circuit which decodes the RETI instruction. By constantly monitoring the data bus CTC can know the termination of the interrupt processing.

The interrupt priority with the Z80 peripheral LSIs is determined by the daisy chain connection. That is, the periperal LSIs are connected in series and physically nearer the MPU, a higher priority is given. Inside the CTC, channel 0 is given the highest priority followed by channels 1, 2 and 3 in this order.

The CTC has the signal lines IEO and IEI. Connect the IEO of a higher peripheral LSI to the IEI of a lower peripheral LSI. Connect the IEI of the highest peripheral LSI to VCC. Leave the IEO of the lowest peripheral LSI unused. In this connection, the CTC interrupt is caused under the following conditions:

- When both IEI and IEO are high, no interrupt is caused. At this time, the $\overline{\text{INT}}$ signal is high. An interrupt can be requested in this state.
- When the CTC outputs the interrupt request signal ($\overline{\text{INT}}$), the IEO of the CTC becomes low. When the MPU accepts the interrupt, the $\overline{\text{INT}}$ pin goes high again.
- When the IEI goes low, the IEO also goes low.
- While the IEI is low, no interrupt can be requested.
- When the IEI goes low while an interupt is being serviced, the interrupt processing is aborted.

[3] Operation Modes

The CTC operates in either the counter mode or the timer mode. Mode selection is programmed by writting the channel control word.

(1) Counter mode

In the counter mode, the number of edge of the pulses applied to the channel's CLK/TRG pin is counted. When pulses have been input, the contents of the down-counter are decremented in synchronization with the rising edge of the next system clock. The pulse's rising edge or falling edge to be counted can be specified by the channel control word.

When the down-counter has read zero, the high level pulses is output from the ZC/TO pin. When the interrupt is enabled by the channel control word, the $\overline{INT}$ pin goes low and an interrupt is requested. When the down-counter has reached zero, the time constant data written in the time constant register is automatically loaded into the down-counter. To load a new time constant value into the down-counter, write the data to the time constant register, and it is loaded into the down-counter after the end of the current count operation.

(2) Timer mode

In the timer mode, the time intervals which are integral multiples of the system clock period. A timer interval is measured against the system clock. The system clock is supplied to the prescaler which divides it by a factor of 16 or 256. The output of the prescaler provides the clock to decrement the down-counter by 1. The time constant data is automatically loaded in the down-counter each time it has reached zero as in the counter mode. When the down-counter has reached zero, the high level pulse is output from the ZC/TO pin.

The period of this pulse is given by the following expression:

$tc*P*TC$

Where, tc = System clock period
P = Prescaler value (16 or 256)
TC = Time constant data (256 for 00H)

The user can specify, by means of the channel control word, whether to start the timer automatically or not and start the timer at the edge of the pulse at CLK/TRG pin or not, and, if the edge is used, start the timer at the rising edge or the falling edge.

3.4.4 CTC Status Transition Diagram and Basic Timing

[1] Transition Diagram

Figure 3.4.3 shows the CTC status transition diagram.

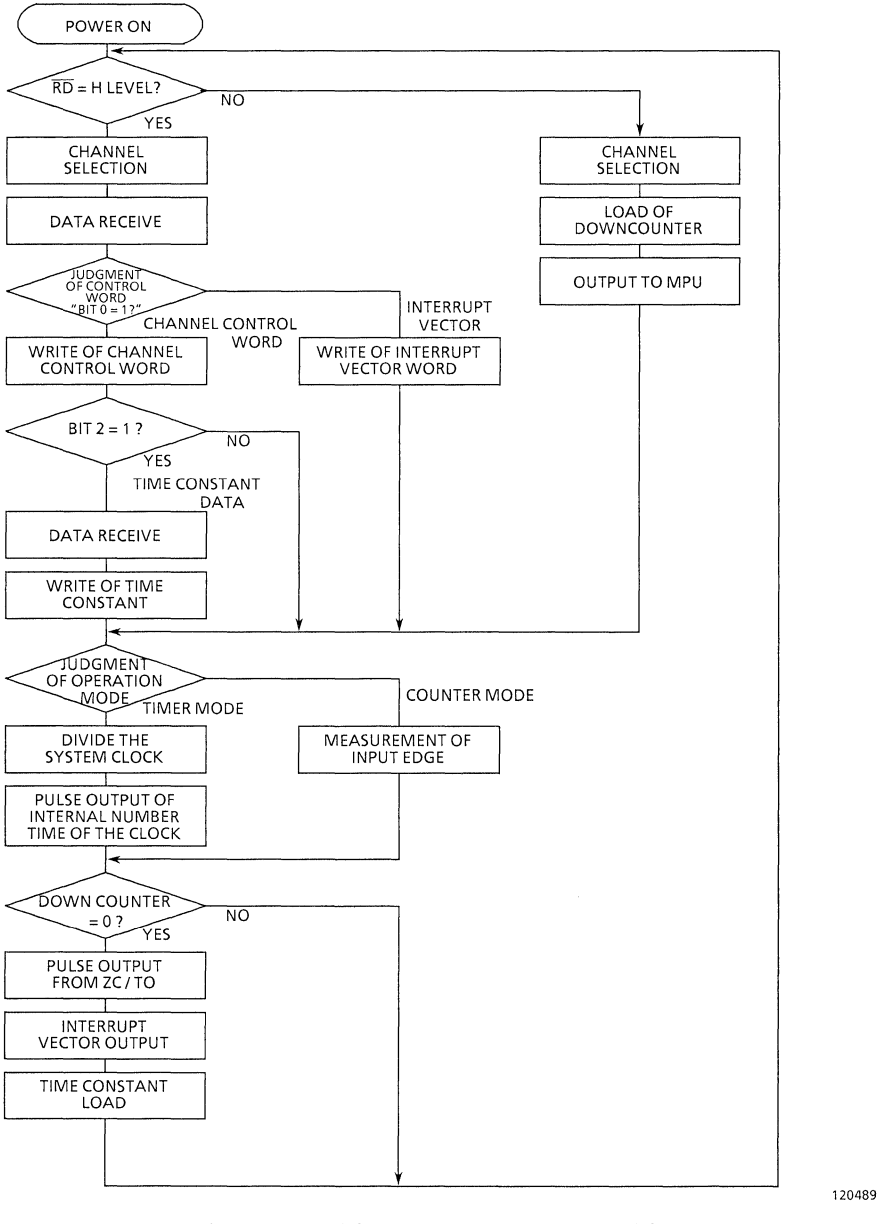


Figure 3.4.3 (a) CTC Transition Diagram (a)

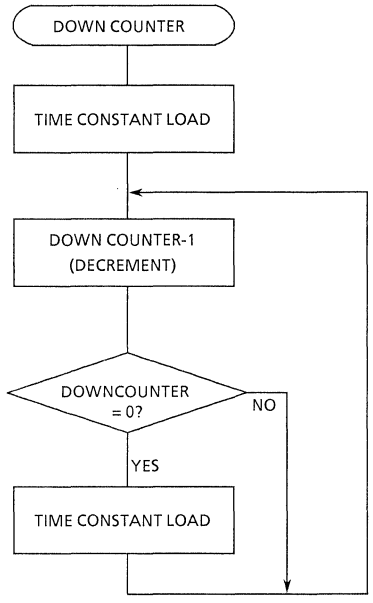


Figure 3.4.3 (b) CTC Transition Diagram (b)

[2] Basic Timing

(1) Write cycle

The write cycle is used to write a channel control word, an interrupt vector, or a time constant. The MPU makes low the $\overline{IORQ}$ pin of the CTC in the subsequent system clock cycle T2 to start the write cycle. It is required to make high the M1 pin of the CTC to indicate that the write cycle is on.

At the start of the cycle, the channel is specified by CS_1 (A_1) and CS_0 (A_0) of the CTC. Thus, the CTC's internal registers are ready to acquire data in system clock T3. Tw is the wait state to be automatically inserted by the MPU.

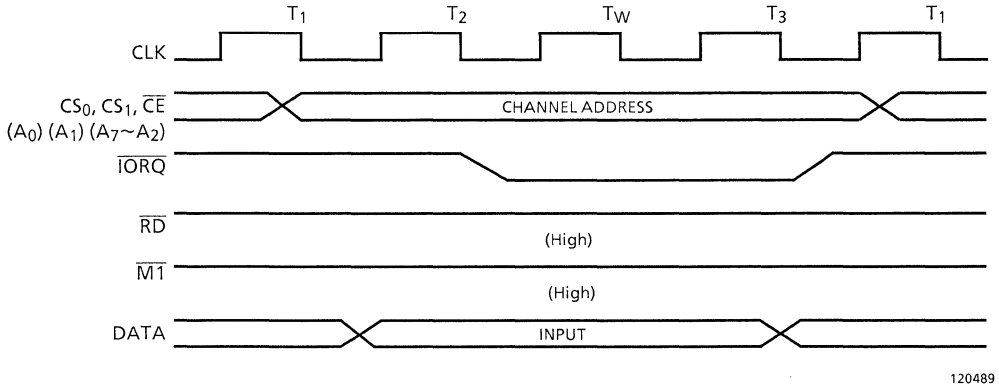


Figure 3.4.4 Write Timing

(2) Read cycle

The read cycle is used to read the contents of the down- counter. In system clock cycle T2, the MPU makes low the $\overline{RD}$ and $\overline{IORQ}$ pins of the CTC to start the read cycle. It is required to make high the $\overline{M1}$ pin to indicate that the read cycle is on. At the start of the read cycle, the channel is specified by CS_1 (A_1) and CS_0 (A_0) of the CTC. On the rising edge of system clock T_W, the contents of the down-counter at the time of the rising edge of T2 are put on the data bus. T_W is the wait state to be automatically inserted by the MPU.

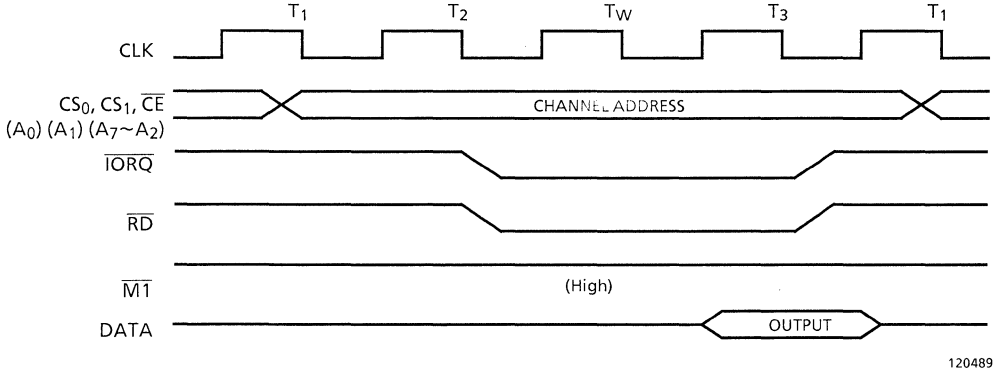


Figure 3.4.5 Read Timing

[3] Counter mode

In the counter mode, the down-counter is decremented in synchronization with the system clock, at the edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be 2 times greater than the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the down-counter is decrmented one system clock later. When the down-counter has reached zero, a high level pulse is output from the ZC/TO pin.

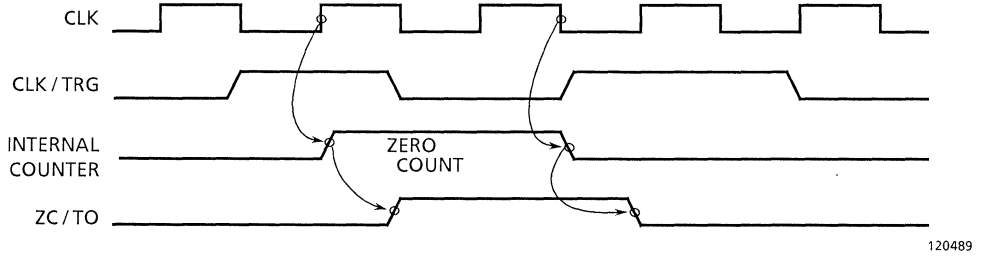


Figure 3.4.6 Counter Mode Timing

[4] Timer mode

The timer starts operating at the second rising edge of the system clock from the rising edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the timer starts one system clock cycle later.

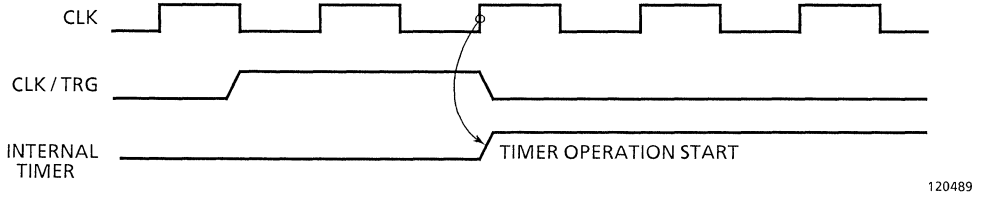


Figure 3.4.7 Timer Mode Timing

[5] Interrupt acknowledge cycle

Having received the interrupt request signal ($\overline{INT}$) from the CTC, the MPU makes low the CTC's $\overline{M1}$ pin and $\overline{IORQ}$ pin to provide the acknowledge signal. The $\overline{IORQ}$ pin goes low 2.5 system clocks later than the $\overline{M1}$ pin. To stabilize the signal lines (IEI and IEO) in daisy chain connection, the interrupt request cannot be changed on each channel while the $\overline{M1}$ pin is low. The $\overline{RD}$ pin is held high to make distinction between the instruction fetch cycle and the interrupt acknoledge cycle. While the $\overline{RD}$ pin is high, the CTC's interrupt control circuit determines the interrupt-requesting channel with the highest priority. When the CTC's IEI is high and the $\overline{M1}$ pin and $\overline{IORQ}$ pin go low, the interrupt vector is output from the interrupt requesting channel with the highest priority on the data bus. At this time, 2 system clock cycles are automatically inserted by the MPU as a wait state to maintain the stabilization of the daisy chain connection.

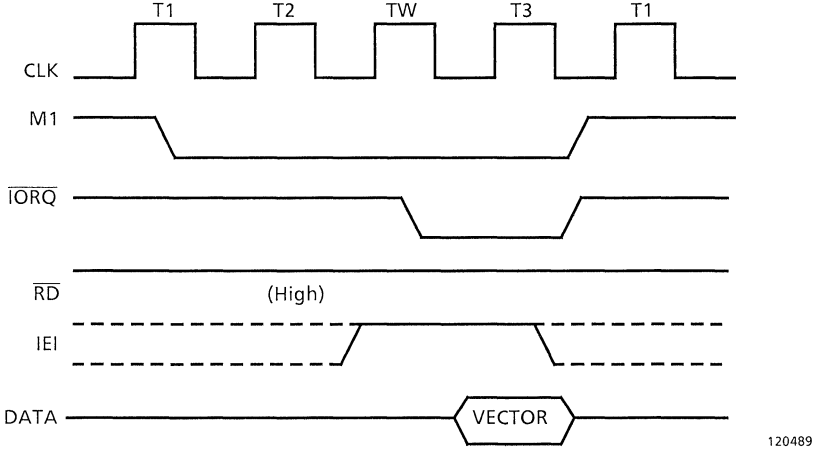


Figure 3.4.8 Interrupt Acknowledge Timing

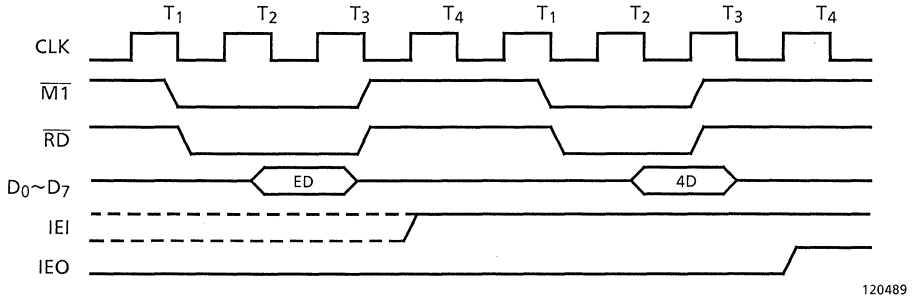
[6] Return from interrupt processing

Return from the interrupt processing is performed by MPU's executing the RETI instruction. This RETI instruction must be used at the end of the interrupt processing routine. When this instruction is executed by the MPU, the CTC's IEI and IEO return to the state active before the interrupt has been serviced.

The RETI instruction is a 2-byte instruction. Its code is EDH 4DH. The CTC decodes this instruction to check if there is the next interrupt request channel.

In the diasy chain structure, the interrupting LSI's IEI and IEO are held high and low respectively at the time the instruction code EDH has been decoded.

The code following EDH is 4DH, only the peripheral LSI which has sent the interrupt vector (that is, the LSI whose IEI is high and IEO is low) returns from the interrupt processing. This restarts the processing of the pending interrupt of the next higher peripheral LSI.



Figureure 3.4.9 Interrupt Return Timing

3.4.5 CTC Operational Procedure

To operates the CTC in the counter mode or the timer mode, the channel control word and time-constant data must be written to the CTC. To enable interrupts by the channel control word, the interrupt vector must be written to the CTC.

[1] I/O Address and Channel Control Word

To write the channel control word to the CTC, the channel must be specified by the corresponding channel I/O address. Table 3.4.1 shows the channel I/O addresses.

Table 3.4.1 Channel I/O Addresses

Channel	I/O address
0	#10
1	#11
2	#12
3	#13

120489

The channel control word to be written to the CTC consists of 8 bits. The system data bus bit D0 through D7 correspond to bit 0 through 7 respectively. Figure 3.4.10 shows the meaning of each bit. Table 3.4.2 shows the function of each bit.

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt	Counter/ timer	Prescaler	Edge	Trigger	Time constant	Reset	1

120489

Figure 3.4.10 Channel Control Word

For the channel control word, D0 must always be 1.

Table 3.4.2 Meanings and Function of Channel Control Words (1/2)

Bit	Meaning and function	
	0	1
Bit 7 (D7)	Disables channel interrupt	Enables channel interrupt. When this bit is set to "1", the interrupt vector must be written to the CTC before the down-counter starts. When the channel control word whose D7 bit is "1" is written to an already operating channel, the interrupt occurs only when the down-counter has reached zero for the first time after the writing of the new channel control word.
Bit 6 (D6)	Puts the channel in the timer mode. Puts the system clock into the prescaler and outputs the divided signal to the down-counter.	Puts the channel in the counter mode. The down-counter is decremented at the edge trigger entered at the CLK / TRG pin. In the counter mode, the prescaler is not used.
Bit 5 (D5)	Used only in the timer mode. The prescaler is set to that it divides the system clock by 16.	Used only in the timer mode. The prescaler is set to that it divides the system clock by 256.
Bit 4 (D4)	In the timer mode, the timer operation starts at the falling edge of the trigger pulse (CLK/TRG). In the counter mode, the down-counter is decremented at the falling edge of the external clock pulse (CLK/TRG)	In the timer mode, the timer operation starts at the rising edge of the trigger pulse (CLK/TRG). In the counter mode, the down-counter is decremented at the rising edge of the trigger pulse (CLK/TRG).
Bit 3 (D3)	Used only in the timer mode. The timer operation starts on the rising edge following 2 system clocks after the timer constant has been loaded into the down-counter.	Used only in the timer mode. The timer operation is started by the external trigger pulse. That is, the timer starts opreating at the trigger pulse entered after the rising edge following 2 system clocks after the time constant has been loaded into the down counter. If the interval between the system clock and the trigger pulse satisfies the setup time, the prescaler starts operating on the second rising edge. Otherwise, the prescaler starts operating on the rising edge following 3 system clocks. If the trigger pulse is entered before the time constant is loaded, the same effect as when bit 3 = "0"

120489

(2/2)

Bit	Meaning and function	
	0	1
Bit 2 (D2)	Indicates that the channel control word is not followed by the writing of time constant. When the channel is in the reset state, this bit cannot be set to "0" in the first channel control word. To change states with the time constant unchanged, the channel control word with this bit set to "0" must be entered.	Indicates that the channel control word is followed by the writing of time constant. If the time constant is written during down-counter operation, the new time constant is set to the time constant register with proceeding the current counting. When the down-counter has reached zero, the new time constant is loaded into the down-counter.
Bit 1 (D1)	Continues the current channel operaiton.	Stops the down-counter operation. When this bit is set to "1", the channel operation stops but the channel control register bits remain unchanged. When bit 2 = "1" and bit 1 = "1", the channel remains stopped until a new time constant is written. Channel restart is set up after the new time constant is programmed. The channel is restarted according to the state of bit 3. When bit 2 = "0" and bit 1 = "1", the channel operation does not start until a new channel control word is written.

120489

[2] Time-Constant Data

In either the time mode or the counter mode, the time-constant data must be loaded into the time constant register. When bit 2 (D2) of the channel control word is "1", the time constant is loaded into the time constant register immediately after the channel control word is written. A time-constant value must be an integer in a range of 1 to 256. When the 8 bits of a time constant are all "0"s, such a time constant is assumed to be 256. Figure 3.4.11 shows the bit configuration of time- constant data.

D7	D6	D5	D4	D3	D2	D1	D0
TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0

120489

Figure 3.4.11 Time-Constant Data

[3] Interrupt Vector

In the MPU mode-2 interrupt, the interrupting channel must give an interrupt vector to the MPU. An interrupt vector is written to the channel-0 interrupt vector register with bit 0 (D0) = "0". The vector is written in the same way as the channel control word is written to channel 0. However, bit 0 (D0) of the vector should always be "0". Bit 7 (D7) through bit 3 (D3) are user-defined values. Bit 2 (D2) and bit 1 (D1) are automatically given and contain the code of the interrupt- requesting channel having the highest priority. Table 3.4.3 shows the channel codes. Figure 3.4.12 shows the interrupt vector bit configuration.

Table 3.4.3 Channel Codes

Bit 2(D2)	Bit 1(D1)	Channel number	Interrupt priority
0	0	0	(High)
0	1	1	↑
1	0	2	↓
1	1	3	(low)

120489

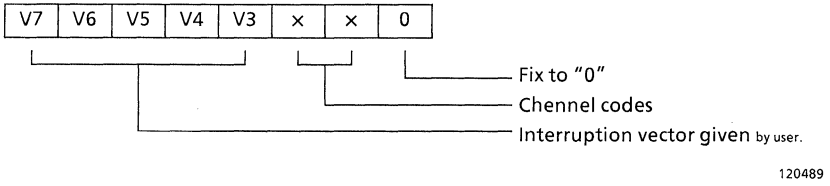


Figure 3.4.12 Interrupt Vector

3.4.6 Using CTC

[1] Counter Mode

The follwing describes how to use the CTC by referring to a program using channel 0 with interrupt disabled.

(a) The counter programming procedure is shown in Figure 3.4.13.

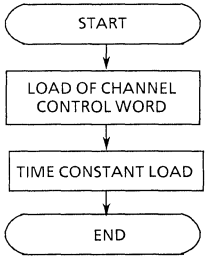


Figure 3.4.13 Counter Programming Procedure

(b) The block diagram for converting the 100 kHz system clock into the 10 kHz equivalent is shown in Figure 3.4.14.

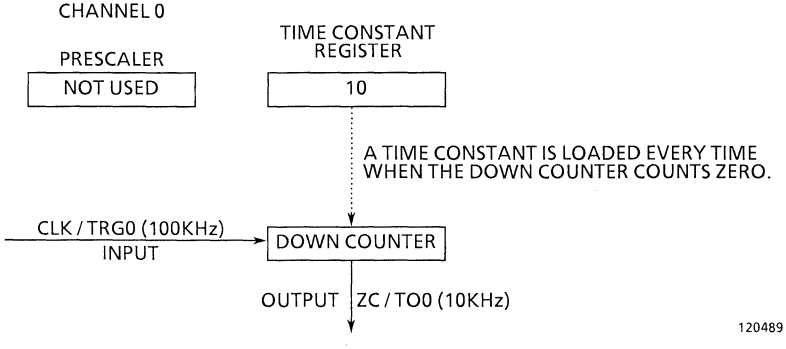


Figure 3.4.14 Down-Counter Block Diagram

(c) The channel control word configuration is shown in Figure 3.4.15.

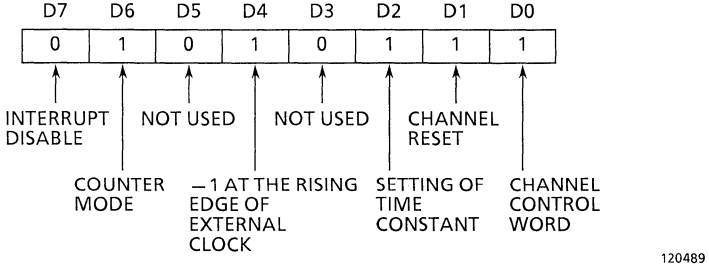


Figure 3.4.15 Channel Control Word Configuration

[2] Timer Mode

(a) The timer programming procedure without using interrupt is shown in Figure 3.4.16.

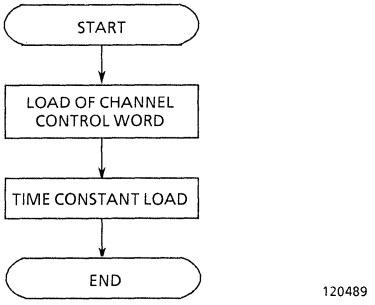


Figure 3.4.16 Timer Programming Procedure

(b) The block diagram for converting the 4 MHz system clock into the 1 kHz equivalent is shown in Figure 3.4.17.

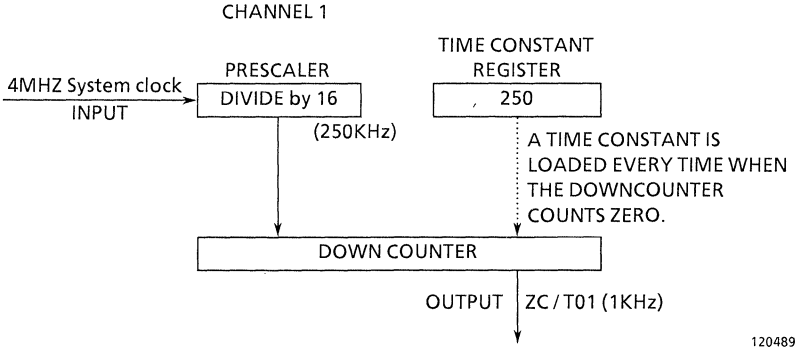


Figure 3.4.17 Timer Block Diagram

(c) The channel control word is shown in Figure 3.4.18.

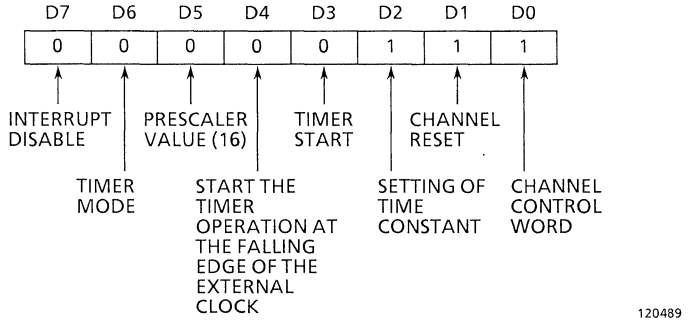


Figure 3.4.18 Channel Control Word Configuration

3.5 Description of I/O Operation

This subsection describes the system configuration and operations of the input/output (I/O) section of the TMPZ84C011A. The I/O consists of five 8-bit general-purpose ports (a total of 40 I/Os). Each bit can be programmed for an input or output port.

3.5.1 I/O Block Diagram

Figure 3.5.1 shows the I/O block diagram.

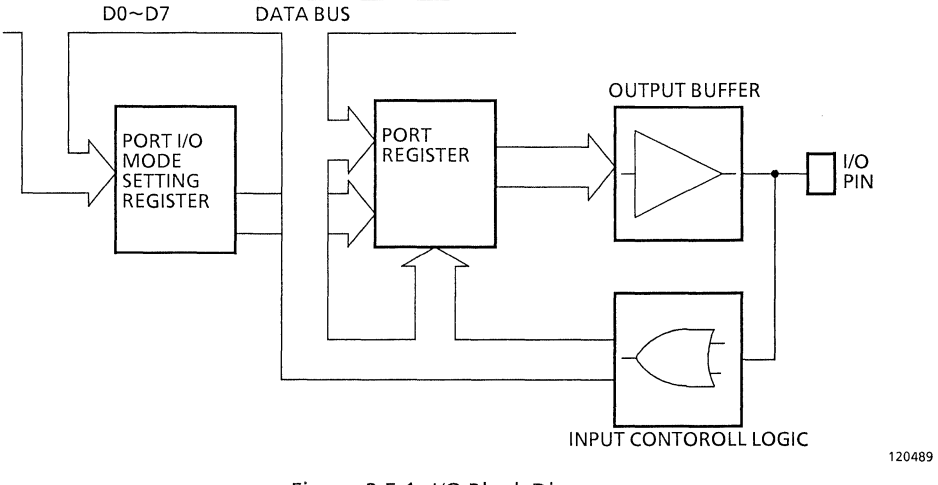


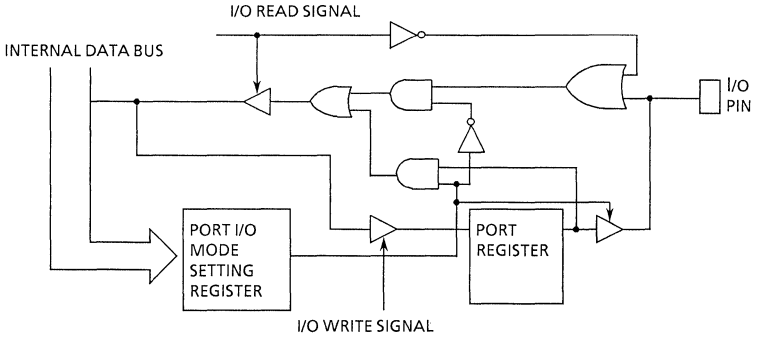
Figure 3.5.1 I/O Block Diagram

3.5.2 I/O System Configuration

The I/O system consists of the I/O setting registers, the port registers, the output buffers, and the input control logics.

The I/O setting registers specify whether the port to be used is to operate as an input port or an output port. Each register corresponds to each port bit. The port registers latch the output data to the port pin. The latched output data can also be taken by program into the TMPZ84C011A. These registers and control logics are controlled by the program-controlled $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$, and I/O address signals.

Figure 3.5.2 shows the logic diagram of the I/O port.



Notes: (1) At reset • I/O register = "0" . . . Input mode
• Port register = "0"
(2) I/O read . . . Enabled by CPU' s I/O input instruction.
(3) I/O write . . . Enabled by CPU' s I/O output instruction.dc

120489

Figure 3.5.2 I/O Port Logic Diagram

3.5.3 I/O Port Basic Operations

The I/O setting registers and port registers of the 5 port of the TMPZ84C011A are assigned with the addresses listed in Table 3.5.1 and are enabled by the MPU's I/O instructions.

Table 3.5.1 I/O Addresses of I/O Port

Register	I/O address
PA I/O setting register	#54
PB I/O setting register	#55
PC I/O setting register	#56
PD I/O setting register	#34
PE I/O setting register	#44
PA port register	#50
PB port register	#51
PC port register	#52
PD port register	#30
PE port register	#40

120489

[1] I/O Setting Registers

The I/O setting registers specify whether an I/O port is to operate as an input port or an output port. As listed in Table 3.5.2, when the register content is "0", the specified port is an input port; when it is "1", the specified port is an output port.

Table 3.5.2 I/O Setting Register Contents and Port Types

I/O setting register contents	Port type
0 (Note)	Input port
1	Output port

Note : At reset

120489

[2] Port Registers / I/O Buffer

The port registers latch the data put on the data bus from the MPU and output it to the port pin. They hold the data until it is rewritten.

When using a port register for input, the data can be rewritten by the MPU's I/O output instruction without directly affecting the register operation. When the type of a port registers are switched from input to output during operation, the contents of the port registers are output to the port pins immediately.

When reading the port output data during an output operation, the outputs of the port registers in front of the output buffer are read, if the port pin's load of the port pin gets heavy, making it possible to read the correct data when the output level goes low. When a port is used for input, it becomes the non-latch type in which the input port state of time when the MPU's I/O input instruction has been executed is read.
At reset, the port register contents are cleared to "0".

[3] I/O Setting Registers vs. Port Registers

As shown in Figure 3.5.3, the I/O setting registers correspond to the port register one by one.

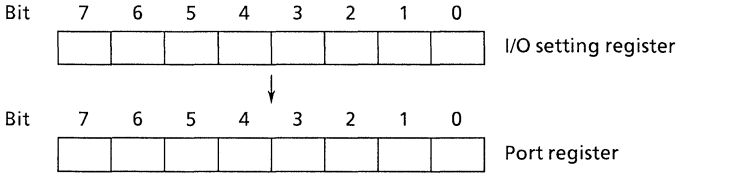


Figure 3.5.3 I/O Setting Registers vs. Port registers

Therefore, the partial change of port bits requires to rewrite contents of the I/O setting registers of the port containing the change bit.

[4] Precautions

The following points must be taken into account in using the I/O ports:

(1) When the port types are changed from input to output, the old contents of the port register are output. To avoid this, the contents of the port register must be rewritten with the data to be output before changing the port types. Then, when the port types are changed from input to output, the desired datas are output to the port pins, preventing the operational errors of the peripheral devices connected to the pin.

(2) After reset, all ports are in the input mode and the port register contents are cleared to "0". Therefore, port type change must not be performed immediately. Instead, the content of the port register corresponding the bit to be used for an output port must be replaced with the desired data by the MPU's I/O output instruction before changing the port types to the output port. This streamlines the port operation.

3.5.4 I/O System Basic Timing

Figure 3.5.4 and 3.5.5 show the basic timings of the I/O ports. The I/O ports of the TMPZ84C011A perform I/O operations according to the MPU's I/O instructions. Therefore, an I/O port address is specified by the low-order 8 bits of the address bus for I/O operations.

In the I/O operation timing, a wait state (T_W) is automatically inserted after clock state T_2 like the operation of the TLCS-Z80 MPU.

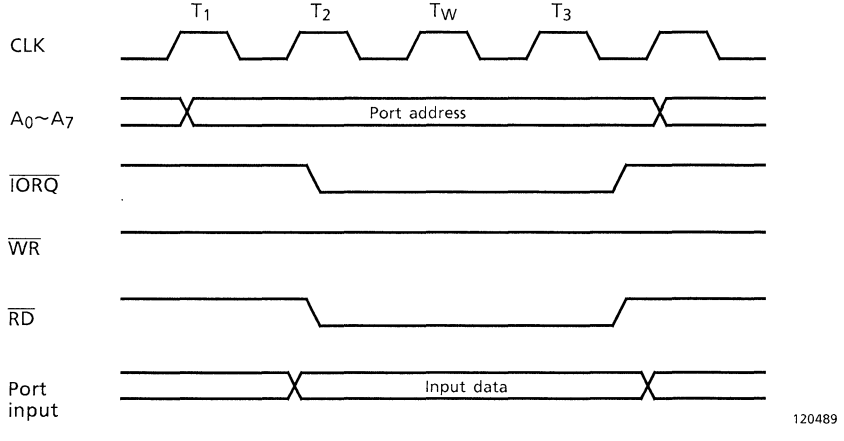


Figure 3.5.4 Input Cycle Timing Chart

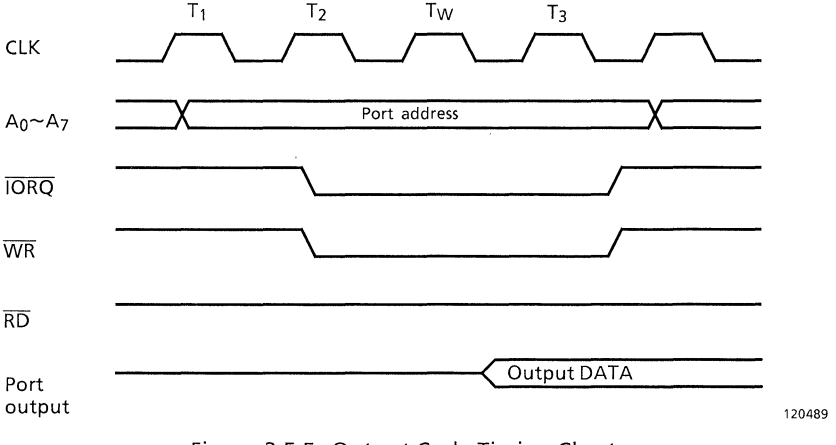


Figure 3.5.5 Output Cycle Timing Chart

4. ELECTRICAL CHARACTERISTICS

4.1 MAXIMUM RATINGS

SYMBOL	ITEM	RATING
VCC	Vcc Supply Voltage with respect to Vss	− 0.5V to 7V
VIN	Input Voltage	− 0.5V to VCC + 0.5V
PD	Power Dissipation (TA = 85°C)	250mW
TSOLDER	Soldering Temperature (Soldering Time 10 sec)	260°C
TSTG	Storage Temperature	− 65°C to 150°C
TOPR	Operating Temperature	− 40°C to 85°C

120489

4.2 DC ELECTRICAL CHARACTERISTICS (1)

$TA = -40°C \sim +85°C$, $Vcc = 5V \pm 10\%$, $Vss = 0V$

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.	MAX.	UNIT.
VIL	Input Low Voltage (except RESET, XTAL1)			− 0.5	−	0.8	V
VIH	Input High Voltage (except RESET, XTAL1)			2.2	−	VCC	V
VILR	Input Low Voltage (RESET)			− 0.5	−	0.45	V
VIHR	Input High Voltage (RESET)			VCC − 0.6	−	VCC	V
VOLC	Output Low Voltage (CLK)	IOL = 2.0mA		−	−	0.6	V
VOHC	Output High Voltage (CLK)	IOH = − 2.0mA		VCC − 0.6	−	−	V
VOL	Output Low Voltage	IOL = 2.0mA		−	−	0.4	V
VOH1	Output High Voltage 1	IOL = − 1.6mA		2.4	−	−	V
VOH2	Output High Voltage 2	IOH = − 250μmA		VCC − 0.8	−	−	V
ILI	Input Leakage Current	Vss ≦ VIN ≦ Vcc		−	−	± 10	μA
ILO	3-state Output Leakage Current in Float	Vss + 0.4 ≦ Vout ≦ VCC		−	−	± 10	μA
ICC1	Power Supply Current	Vcc = 5V, fCLK = (1) VIH = VCC − 0.2V VIL = 0.2V	4MHz	−	15	20	mA
			6MHz	−	22	30	
ICC2	Stand-by supply current	Vcc = 5V, (2) VIH = VCC − 0.2V VIL = 0.2V	4MHz / 6MHz	−	0.5	50	μA
ICC3	Power Supply Current (IDLE MODE)	Vcc = 5V, fCLK = (1) VIH = VCC − 0.2V VIL = 0.2V	4MHz	−	1.0	2.5	mA
			6MHz	−	1.5	4	

120489

Note : (1) fCLK = 1/TCC (MIN)

(2) ICC2 Stand-by Supply Current is guaranteed only when the supplied clock is stopped at a low level during T4 state of the following machine Cycle (M1) next to OP code fetch Cycle of HALT instruction.

4.2 DC ELECTRICAL CHARACTERISTICAS (2)

TA = −40°C ~ +85°C, Vcc = 2.7V ± 5.5V, Vss = 0V (Low Voltage operation)

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
VIL	Input Low Voltage (except $\overline{RESET}$)		−0.3	−	0.2	V
VIH	Input High Voltage (except $\overline{RESET}$)		VCC − 0.2	−	VCC + 0.3	V
VOL	Output Low Voltage	IOL = 1.2mA	−	−	0.4	V
VOH	Output High Voltage	IOL = −0.7mA	2.2	−	−	V
ILI	Input Leakage Current	Vss ≦ VIN ≦ Vcc	−	−	±2	μA
ILO	3-state Output Leakage Current in Float	Vss ≦ Vout ≦ VCC	−	−	±2	μA
ICC1	Power Supply Current	Vcc = 3V, fCLK = 2MHz VIH = VCC VIL = 0.0V	−	−	10	mA
ICC2	Stand-by supply current	Vcc = 3V, (2) VIH = VCC VIL = 0.0V	−	−	10	μA
ICC3	Power Supply Current (IDLE MODE)	Vcc = 3V, fCLK = 2MHz VIH = VCC VIL = 0.0V	−	1.0	2.5	mA

310589

4.3 AC ELECTRICAL CHARACTERISTICS (1) (IN ACTIVE STATE)

TA = −40°C ~ +85°C, VCC = 5V ± 10%, VSS = 0V

4.3.1 AC Characteristics of MPU (in Active State)

(1/3)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
1	TcC	Clock Cycle Time	500	−	DC		165		ns
2	TwCh	Clock Pulse Width (High)	220	−	DC	−	70	−	ns
3	TwCl	Clock Pulse Width (Low)	220	−	DC	−	70	−	ns
4	TfC	Clock Fall Time	−	−	30	−	12	−	ns
5	TrC	Clock Rise Time	−	−	30	−	12	−	ns
6	TdCr (A)	Clock ↑ to Address Valid Delay	−	−	180	−	−	90	ns
7	TdA (MREQf)	Address Valid to $\overline{MREQ}$ ↓ Delay	180	−	−	−	32	−	ns
8	TdCf (MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay	−	−	150	−	−	70	ns
9	TdCr (MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay	−	−	180	−	−	70	ns
10	TwMREQh	$\overline{MREQ}$ Pulse Width (High)	190	−	−	−	62	−	ns
11	TwMREQl	$\overline{MREQ}$ Pulse Width (Low)	460	−	−	−	135	−	ns
12	TdCf (MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay	−	−	190	−	−	70	ns

290589

(2/3)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
13	TdCf (RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay	—	—	180	—	—	80	ns
14	TdCr (RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay	—	—	180	—	—	70	ns
15	TsD (Cr)	Data Setup Time to Clock ↑	60	—	—	30	—	—	ns
16	ThD (RDr)	Data Hold Time to $\overline{RD}$ ↑	0	—	—	0	—	—	ns
17	TsWAIT (Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	80	—	—	60	—	—	ns
18	ThWAIT (Cf)	$\overline{WAIT}$ Hold Time after Clock ↓	15	—	—	10	—	—	ns
19	TdCr (M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay	—	—	170	—	—	80	ns
20	TdCr (M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay	—	—	170	—	—	80	ns
21	TdCr (RFSHf)	Clock ↑ to $\overline{RFSH}$ ↓ Delay	—	—	220	—	—	110	ns
22	TdCr (RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay	—	—	220	—	—	100	ns
23	TdCr (RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay	—	—	180	—	—	70	ns
24	TdCr (RDf)	Clock ↑ to $\overline{RD}$ ↓ Delay	—	—	150	—	—	70	ns
25	TsS (Cf)	Data Setup to Clock ↓ during M2, M3, M4 or M5 Cycles	70	—	—	40	—	—	ns
26	TdA (IORQf)	Address Stable prior $\overline{IORQ}$ ↓	400	—	—	—	110	—	ns
27	TdCr (IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay	—	—	150	—	—	65	ns
28	TdCf (IORQr)	Clock ↓ to $\overline{IORQ}$ ↑ Delay	—	—	180	—	—	70	ns
29	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	240	—	—	—	25	—	ns
30	TdCf (WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay	—	—	150	—	—	70	ns
31	TwWR	$\overline{WR}$ Pulse Width	420	—	—	—	135	—	ns
32	TdCf (WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay	—	—	180	—	—	70	ns
33	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	70	—	—	—	−58	—	ns
34	TdCr (WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay	—	—	120	—	—	60	ns
35	TdWRr (D)	Data Stable from $\overline{WR}$ ↑	200	—	—	—	27	—	ns
36	TdCf (HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓	—	—	350	—	—	260	ns
37	TwNMI	$\overline{NMI}$ Pulse Width	180	—	—	70	—	—	ns
38	TsBUSREQ (Cr)	$\overline{BUSREQ}$ Setup Time to Clock ↑	100	—	—	50	—	—	ns
39	ThBUSREQ (Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	15	—	—	10	—	—	ns
40	TdCr (BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay	—	—	180	—	—	90	ns
41	TdCf (BUSACKr)	Clock ↓ to $\overline{BUSACK}$ ↑ Delay	—	—	160	—	—	90	ns
42	TdCr (Dz)	Clock ↑ to Data Float Delay	—	—	120	—	—	80	ns
43	TdCr (CTz)	Clock ↑ to Control Outputs Float Delay ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)	—	—	140	—	—	70	ns
44	TdCr (Az)	Clock ↑ to Address Float Delay	—	—	140	—	—	80	ns

290589

(3/3)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
45	TdCr (A)	$\overline{MREQ}\uparrow$, $\overline{IORQ}\uparrow$, $\overline{RD}\uparrow$ and $\overline{WR}\uparrow$ to Address Hold Time	200	—	—	—	32	—	ns
46	TsRESET (Cr)	$\overline{RESET}$ to Clock ↑ Setup Time	140	—	—	60	—	—	ns
47	ThRESET (Cr)	$\overline{RESET}$ to Clock ↑ Hold Time	20	—	—	10	—	—	ns
48	TsINTf (Cr)	$\overline{INT}$ to Clock ↑ Setup Time	100	—	—	70	—	—	ns
49	TsINTr (Cr)	$\overline{INT}$ to Clock ↑ Hold Time	15	—	—	10	—	—	ns
50	TdM1f (IORQf)	$\overline{M1}\downarrow$ to $\overline{IORQ}\downarrow$ Delay	1050	—	—	—	362	—	ns
51	TdCf (IORQf)	Clock ↓ to $\overline{IORQ}\downarrow$ Delay	—	—	185	—	—	70	ns
52	TdCr (IORQr)	Clock ↑ to $\overline{IORQ}\uparrow$ Delay	—	—	200	—	—	70	ns
53	TdCf (D)	Clock ↓ to Data Valid Delay	—	—	300	—	—	130	ns

290589

4.3.2 AC Characteristics of CGC (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
54	TRST (INT) S	CLK restart time by $\overline{INT}$ (STOP Mode)	—	2^{14} + 2.5TcC	—	—	(2^{14} + 2.5)TcC	—	ns
55	TRST (NMI) S	CLK restart time by $\overline{NMI}$ (STOP Mode)	—	2^{14} + 2.5TcC	—	—	(2^{14} + 2.5)TcC	—	ns
56	TRST (INT) I	CLK restart time by $\overline{INT}$ (IDLE Mode)	—	2.5 *TcC	—	—	2.5 TcC	—	ns
57	TRST (NMI) I	CLK restart time by $\overline{NMI}$ (IDLE Mode)	—	2.5 *TcC	—	—	2.5 TcC	—	ns
58	TRST (RESET) I	CLK restart time by $\overline{RESET}$ (IDLE Mode)	—	TcC	—	—	TcC	—	ns

290589

4.3.3 AC Characteristics of CTC (in Active State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
59	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall (in case of generating only interrupt immediately before M1 cycle)	—	—	350	—	—	130	ns
60	TdIEI (IEOf)	Delay from IEI fall to IEO fall	—	—	200	—	—	100	ns
61	TdIEI (IEOr)	Delay from IEI rise to IEO rise (after ED decode)	—	—	250	—	—	110	ns

290589

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
62	TsCTR (INT)	CLK/TRG setup for detection of interrupt TsCTR (C) Satisfied	TcC + 200 + T68 + T48	—	—	TcC + 120 + T68 + T48	—	—	ns
		TsCTR (C) not Satisfied	2TcC + 200 + T68 + T48	—	—	2TcC + 120 + T68 + T48	—	—	ns
63	TcCTR	CLK/TRG Frequency	2TcC	—	—	2TcC	—	—	ns
64	TrCTR	CLK/TRG rising time	—	—	50	—	—	40	ns
65	TfCTR	CLK/TRG falling time	—	—	50	—	—	40	ns
66	TwCTRl	Low CLK/TRG Pulse Width	240	—	—	120	—	—	ns
67	TwCTRh	High CLK/TRG Pulse Width	240	—	—	120	—	—	ns
68	TsCTR (Cs)	Set-up time up to CLK/TRG rise for clock rise requiring immediate count (counter mode)	300	—	—	150	—	—	ns
69	TsCTR (CT)	Set-up time up to CLK/TRG rise for clock rise requiring immediate start of prescaler (timer mode)	300	—	—	150	—	—	ns
70	TdC (ZC / TOr)	Delay from clock rise to ZC/ TO rise	—	—	300	—	—	140	ns
71	TdC (ZC / TOf)	Delay from clock fall to ZC/ TO fall	—	—	220	—	—	140	ns

290589

4.3.4 AC Characteristics of I/O (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (6MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
72	TdCf (Pout)	Delay from clock fall to port data output	—	—	650	—	—	300	ns
73	TsPIN (IORDf)	Port Input to $\overline{IORQ}$ and $\overline{RD}$ fall Set-up time	0	—	—	0	—	—	ns
74	ThPIN	Port Input to $\overline{IORQ}$ and $\overline{RD}$ fall Hold Time	0	—	—	0	—	—	ns

290589

4.4 CAPACITANCE

Ta = 25°C

SYMBOL	Item	TEST COINDITION	MIN.	TUP.	MAX.	UNIT.
C_{IN}	Input Capacitance	f = 1MHz ALL terminals except that to be measured should be earthed	—	5	—	PF
Cout	Output Capacitance		—	30	—	PF

290589

4.5 AC TIMING CHARTS (1) (IN ACTIVE STATE)

4.5.1 AC Timing Charts of CPU (in Active State)

Figure 4.5.1 through 4.5.8 show the basic timing charts. The circled numbers in these charts correspond to the numbers under the number column of the AC Electrical Characteristics Tables.

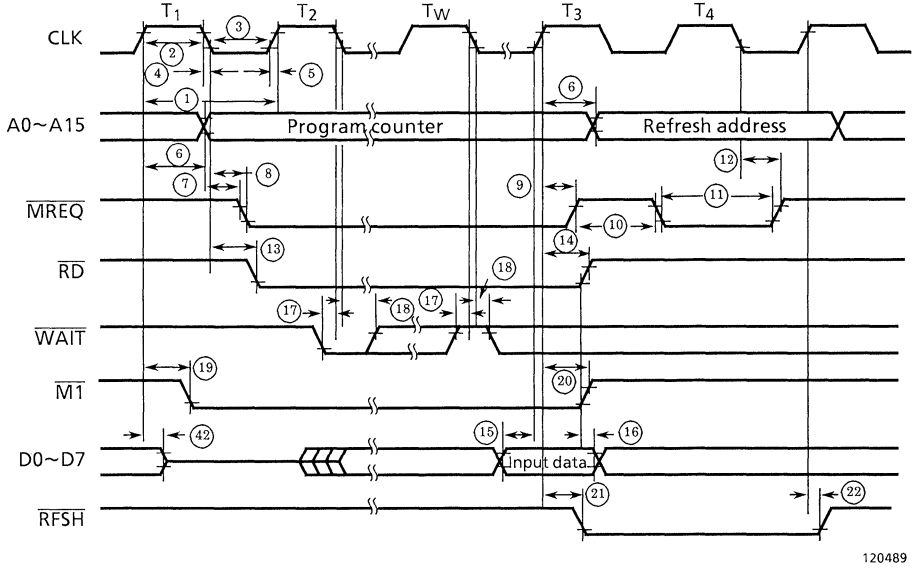


Figure 4.5.1 Opcode Fetch Cycle

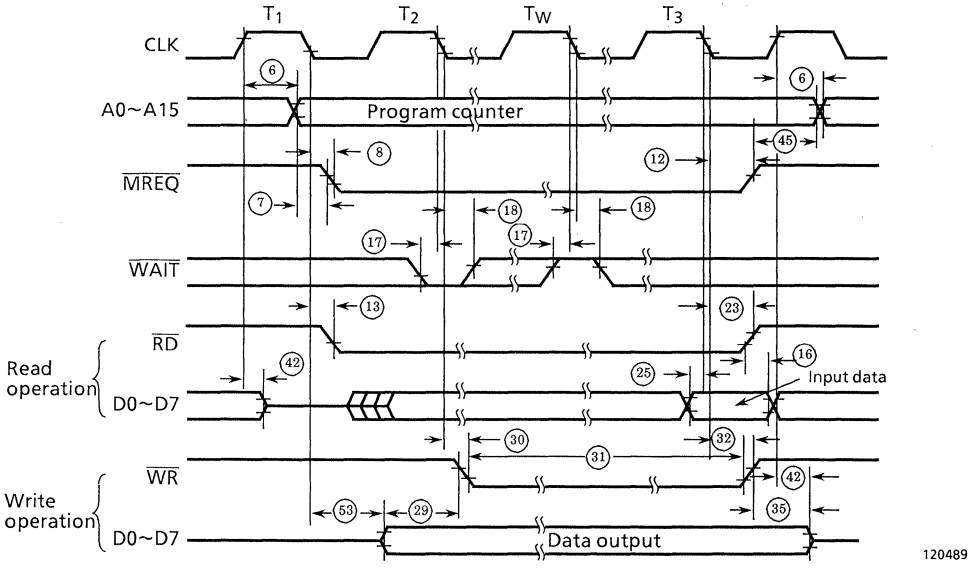
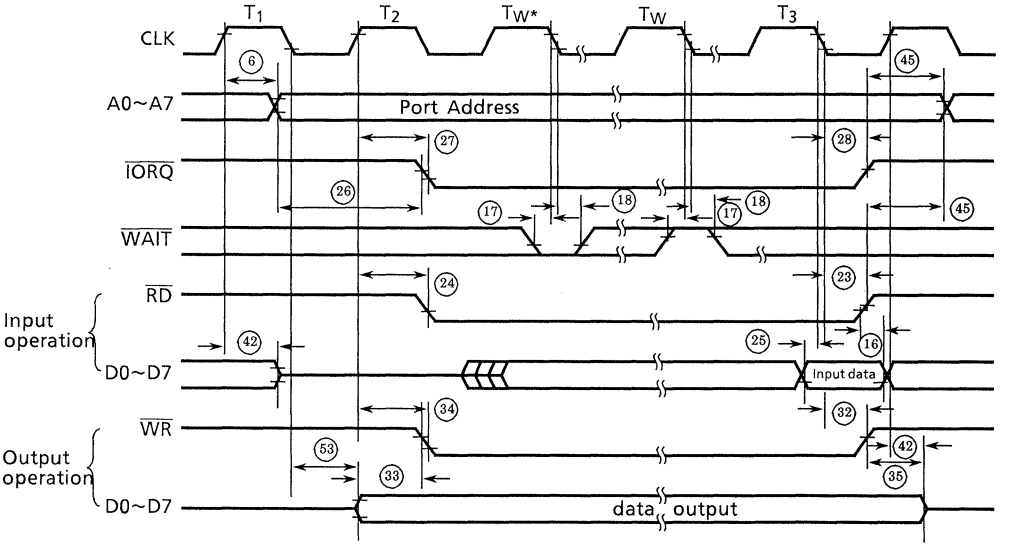
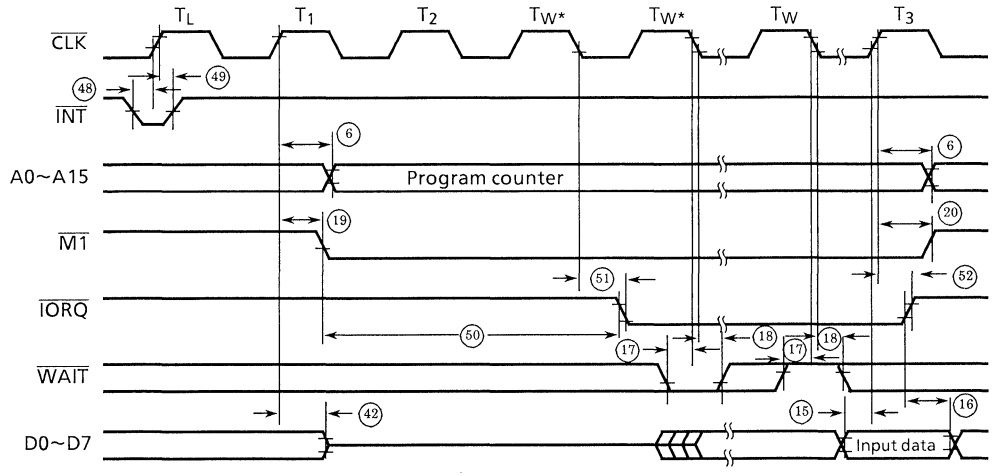


Figure 4.5.2 Memory Read/Write Cycle



Note : 1-wait state (T_W*) is inserted automatically by MPU

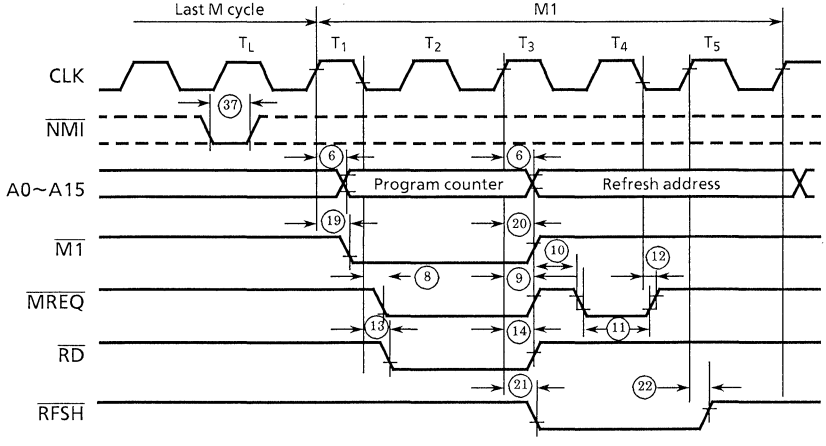
Figure 4.5.3 I/O Cycle



Note : 1. T_L : Last state of insturuction
2. 2-wait state (TW*) is inserted automatically by MPU

120489

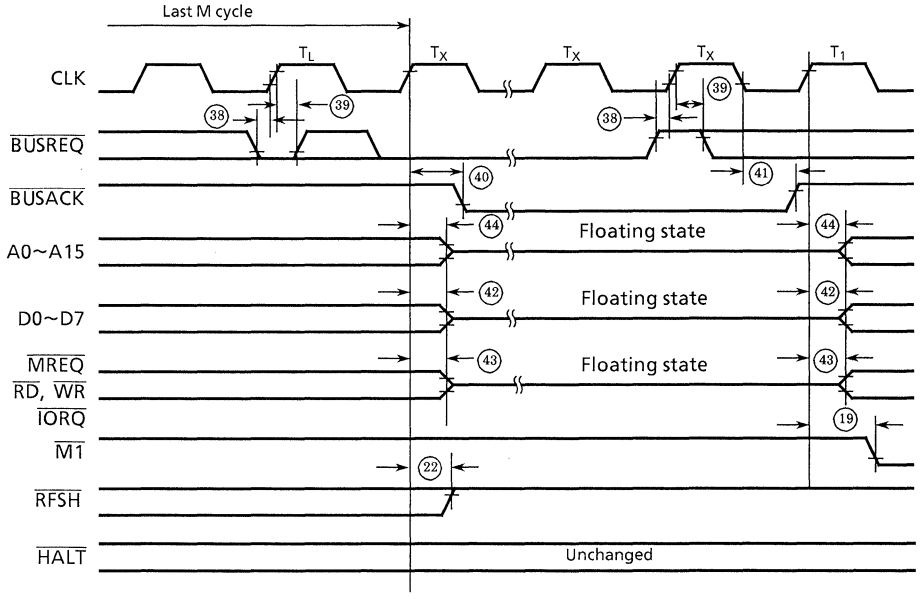
Figure 4.5.4 Interrupt Request/Acknowledge Cycle



Note : $\overline{NMI}$ is asynchronous, but its falling edge signal must occur synchronously with the rising edge of previous TL state for correct response to the supsequent machine cycle.

120489

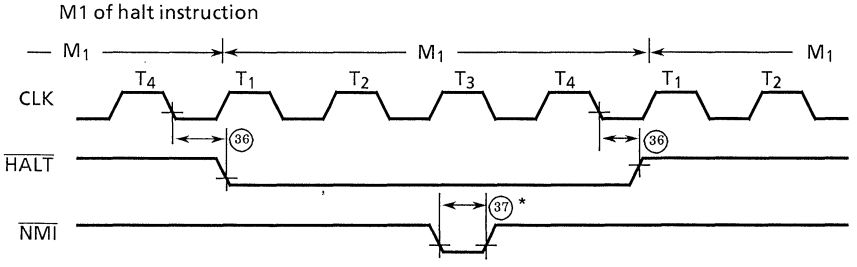
Figure 4.5.5 Non-maskable Interrupt Request Cycle



Note : 1. TL : Last state of a given machine cycle
2. Tx : Clock used by peripheral LSI that made request.

120489

Figure 4.5.6 Bus Request/Acknowlege Cycle



Note : $\overline{INT}$ ssignal is also used for releasing HALT state.

120489

Figure 4.5.7 Halt Acknowledge Cycle

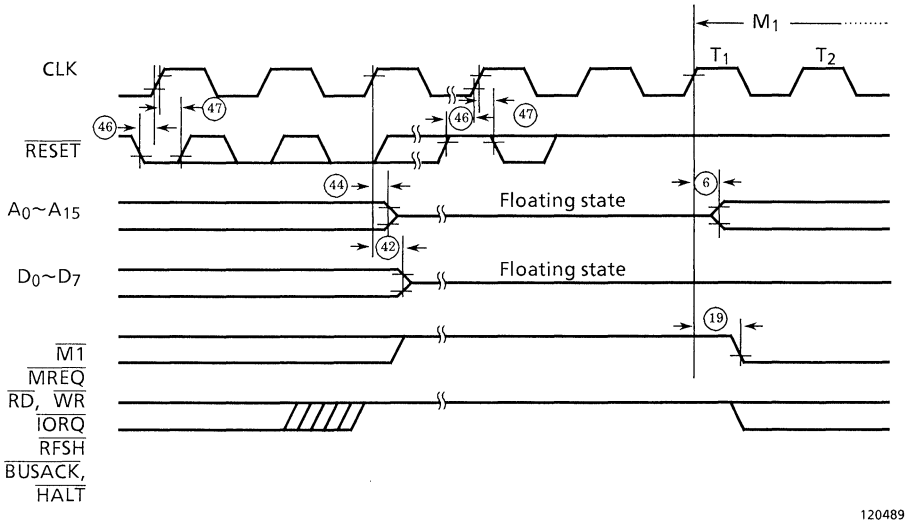


Figure 4.5.8 Reset Cycle

4.5.2 AC Timing Cfarts of CGC (in Active State)

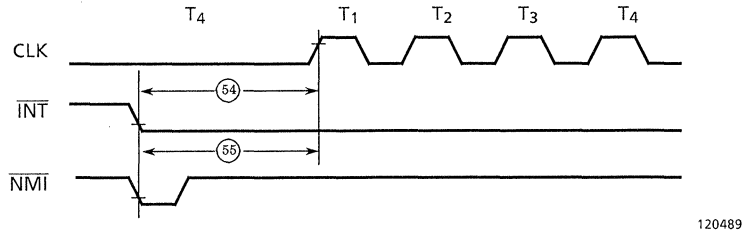


Figure 4.5.9 Clock Restart Timing (STOP mode)

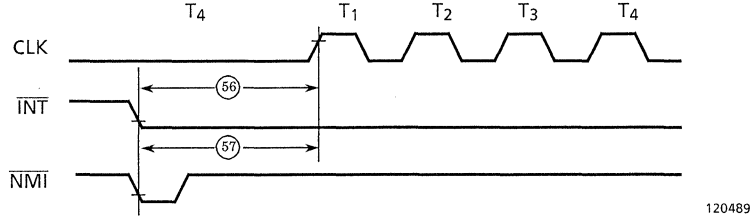


Figure 4.5.10 Clock Restart Timing (IDLE mode)

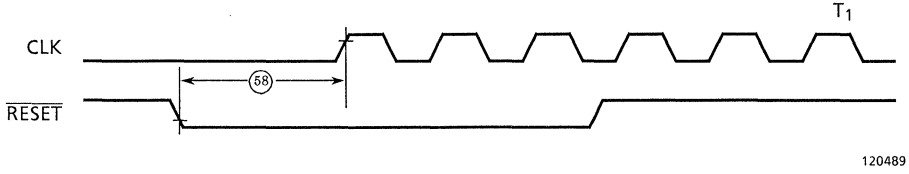


Figure 4.5.11 Timing of Clock Start by $\overline{\text{RESET}}$ (IDLE mode)

4.5.3 AC Timing Charts of CTC (in Active State)

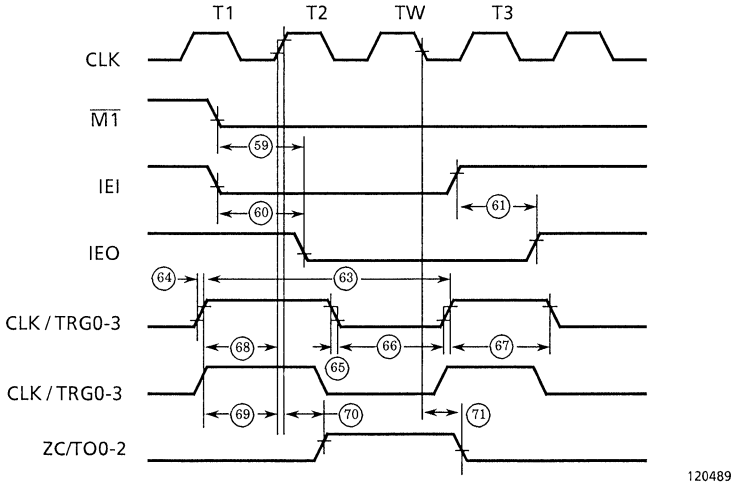


Figure 4.5.12 CTC Timing Diagram

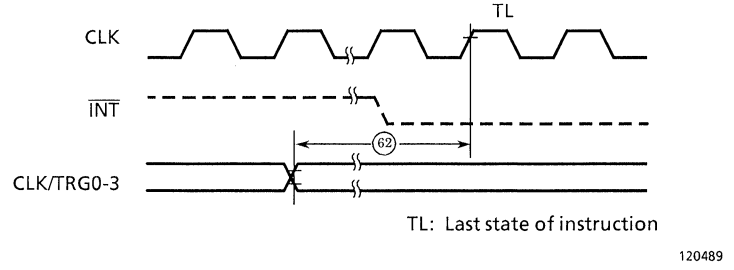


Figure 4.5.13 CTC Interrupt Occurrence Timing

4.5.4 AC Timing Charts of I/O (in Active State)

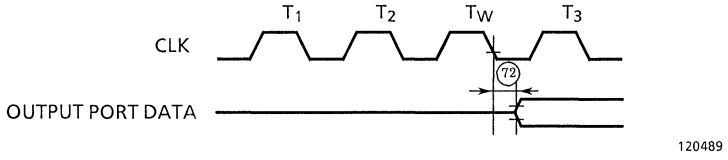


Figure 4.5.14 I/O OUTPUT Timing

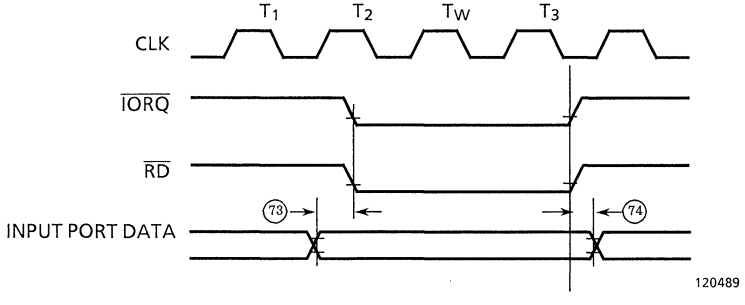


Figure 4.5.15 I/O INPUT Timing

4.6 AC ELECTRICAL CHARACTERISTICS (2) (IN INACTIVE STATE)

Ta = −40°C~ +85°C, VCC = 5V ± 10%, VSS = 0V

4.6.1 AC Characteristics of CGC (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
1	TcC	CLOCK Cycle	500	—	DC	—	165	—	ns
2	TwCh	CLOCK Width (High)	220	—	DC	—	70	—	ns
3	TwCl	CLOCK Width (Low)	220	—	DC	—	70	—	ns
4	TfC	CLOCK Fall Time	—	—	30	—	12	—	ns
5	TrC	CLOCK Rise Time	—	—	30	—	12	—	ns
6	TRST (INT) S	Clock (CLK) restart Time by $\overline{INT}$ (STOP mode)	—	2^{14} + 2.5TcC	—	—	2^{14} + 2.5TcC	—	ns
7	TRST (NMI) S	Clock (CLK) restart Time by $\overline{NMI}$ (STOP mode)	—	2^{14} + 2.5TcC	—	—	2^{14} + 2.5TcC	—	ns
8	TRST (INT) I	Clock (CLK) restart Time by $\overline{INT}$ (IDLE mode)	—	2.5 *TcC	—	—	2.5 TcC	—	ns
9	TRST (NMI) I	Clock (CLK) restart Time by $\overline{NMI}$ (IDLE mode)	—	2.5 *TcC	—	—	2.5 TcC	—	ns
10	TRST (RESET) I	Clock (CLK) restart Time by $\overline{RESET}$ (IDLEmode)	—	1TcC	—	—	1TcC	—	ns
11	TsHALT (M1r)	$\overline{HALT}$ set-up Time	10	—	—	10	—	—	ns

290589

4.6.2 AC Characteristics of CTC (in Inactive State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
12	Th	Hold Time	20	—	—	0	—	—	ns
13	TsCS (C)	CS (A1, A0) set-up time to clock ↑	280	—	—	100	—	—	ns
14	TsCE (C)	CE (A_7 to A_2) set-up time to clock ↑	220	—	—	150	—	—	ns
15	TsIO (C)	$\overline{IORQ}$ ↓ set-up time to clock ↑	280	—	—	70	—	—	ns
16	TsRD (C)	$\overline{RD}$ ↓ set-up time to clock ↑	260	—	—	70	—	—	ns
17	TdC (DO)	clock ↑ to Data Valid Delay	—	—	260	—	—	130	ns
18	TdC (DOz)	$\overline{IORQ}$, $\overline{RD}$ ↑ to Data Float Delay	—	—	250	—	—	90	ns
19	TsDI (C)	Data Input set-up time to clock ↑	90	—	—	40	—	—	ns
20	TsM1 (C)	$\overline{M1}$ set-up time to clock ↑	240	—	—	70	—	—	ns
21	TdM1 (IEO)	clock ↑ $\overline{M1}$ ↓ to IEO ↓ Delay (in case of generating only interrupt immediately befor M1) cycle	—	—	350	—	—	130	ns
22	TdIO (DOI)	$\overline{IORQ}$ ↓ to Data out delay (INTA cycle)	—	—	380	—	—	110	ns
23	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	200	—	—	100	ns
24	TdIEI (IEOr)	IEI ↑ to IEO ↑ Delay (after ED decode)	—	—	250	—	—	110	ns
25	TdC (INT)	Clock ↑ to $\overline{INT}$ ↓ Delay	—	—	TcC + 200	—	—	TcC + 120	ns
26	TdCLK (INT)	CLK/TRG ↑ to $\overline{INT}$ ↓ Delay TsCTR(c) Satisfied	TcC + 200 + T32	—	—	—	TcC + 120 + 70 + T31	—	ns
		TsCTR (c) not Satisfied	2TcC + 200 + T32	—	—	—	2TcC + 120 + 70 + T31	—	ns
27	TcCTR	CLK/TRG Frequency	2TcC	—	—	—	2TcC	—	ns
28	TrCTR	CLK/TRG rising time	—	—	50	—	—	40	ns
29	TfCTR	CLK/TRG falling time	—	—	50	—	—	40	ns
30	TwCTRl	LOW CLK/TRG Pulse width	240	—	—	120	—	70	ns
31	TwCTRh	High CLK/TRG Pulse width	240	—	—	120	—	—	ns
32	TsCTR (Cs)	Set-up time up to CLK/TRG rise for clock rise requiring immediate (counter mode)	300	—	—	150	—	—	ns

290589

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
33	TsCTR (Ct)	Set-up time up to CLK/TRG rise for clock rise requiring immediate start of prescaler (timer mode)	300	—	—	150	—	—	ns
34	TdC (ZC / TOr)	Delay from clock rise to ZC/TO rise	—	—	300	—	—	140	ns
35	TdC (ZC / TOf)	Delay from clock fall to ZC/TO fall	—	—	220	—	—	140	ns

290589

4.6.3 AC Characteristics of I/O (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C011AF-6 Vcc = 2.7V~5.5V (2MHz)			TMPZ84C011AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
36	TsCE (Cr)	CE (A7~A0) to clock ↑ Set-up time	300	—	—	150	—	—	ns
37	TsIO (C)	Set-up time from clock ↑ to $\overline{IORQ}$ ↓	150	—	—	70	—	—	ns
38	TsRD (C)	Set-up time from clock ↑ to $\overline{RD}$ ↓	150	—	—	70	—	—	ns
39	TdC (DO)	Delay from clock ↑ to Data Output	—	—	300	—	—	130	ns
40	TdC (DOz)	Delay from clock $\overline{IORQ}$ and $\overline{RD}$ ↑ to Data Float	—	—	150	—	—	90	ns
41	TsDI (C)	Data Input to clock ↑ Set-up time	0	—	—	0	—	—	ns
42	TsWR (C)	Set-up time from $\overline{WR}$ ↓ to clock ↑	150	—	—	70	—	—	ns
43	TdIOWRr (D)	Output Data Stable from $\overline{IORQ}$, $\overline{WR}$ ↓	100	—	—	20	—	—	ns
44	TdCf (Pout)	Delay from clock ↓ to Data Output	—	—	650	—	—	300	ns
45	TSPIN (IORDf)	Port Input to $\overline{IORQ}$ and $\overline{RD}$ ↓ Set-up time	0	—	—	0	—	—	ns
46	ThPIN (IORDr)	Port Input to $\overline{IORQ}$ and $\overline{RD}$ ↑ Hold Time	0	—	—	0	—	—	ns

290589

Note : Timing Measurements are made at the following voltage.

Input : VIH = 2.4V VIL = 0.4V

Output : VOHC = V_{CC} − 0.6V VOLC = 0.6V (CLK)

Output : VOH = 2.4V VOL = 0.8V (Except CLK)

C_L = 100pF

4.7 AC TIMING CHARTS (2) (IN INACTIVE STATE)

4.7.1 AC Timing Charts of CGC (in Inactive State)

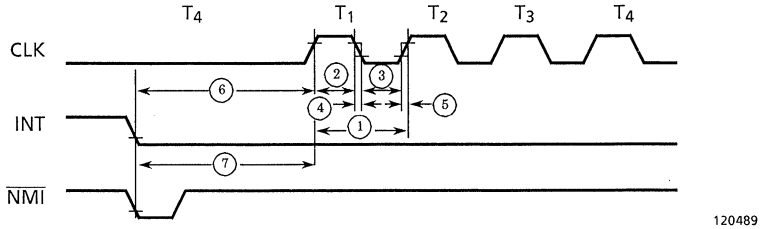


Figure 4.7.1 Clock Restart Timing (STOP mode)

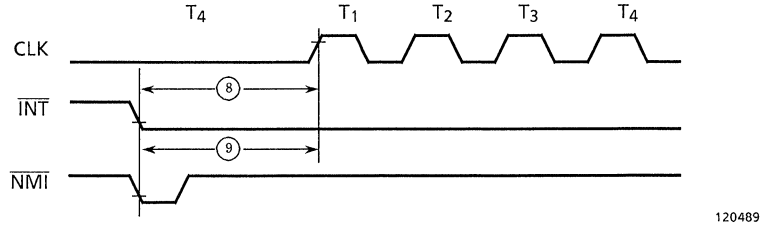


Figure 4.7.2 Clock Restart Timing (IDLE mode)

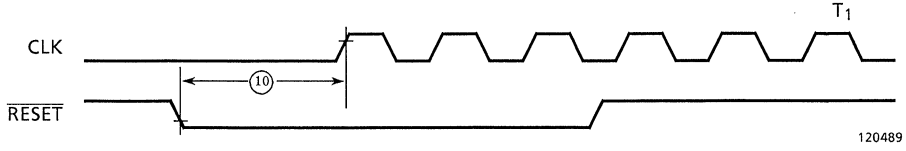


Figure 4.7.3 Timing if Clock Restart by RESET (IDLE mode)

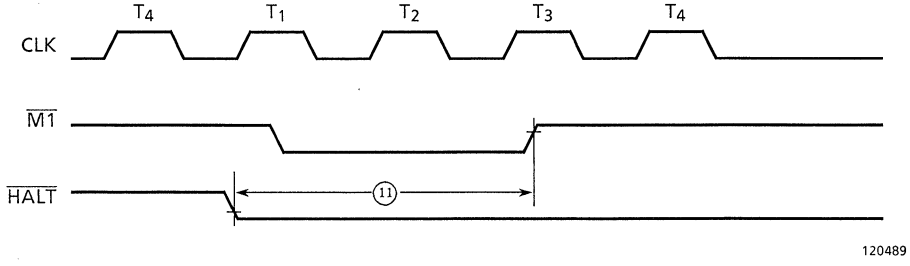


Figure 4.7.4 Clock Suspension Timing (IDLE/STOP modes)

4.7.2 AC Timing Charts of CTC (in Inactive State)

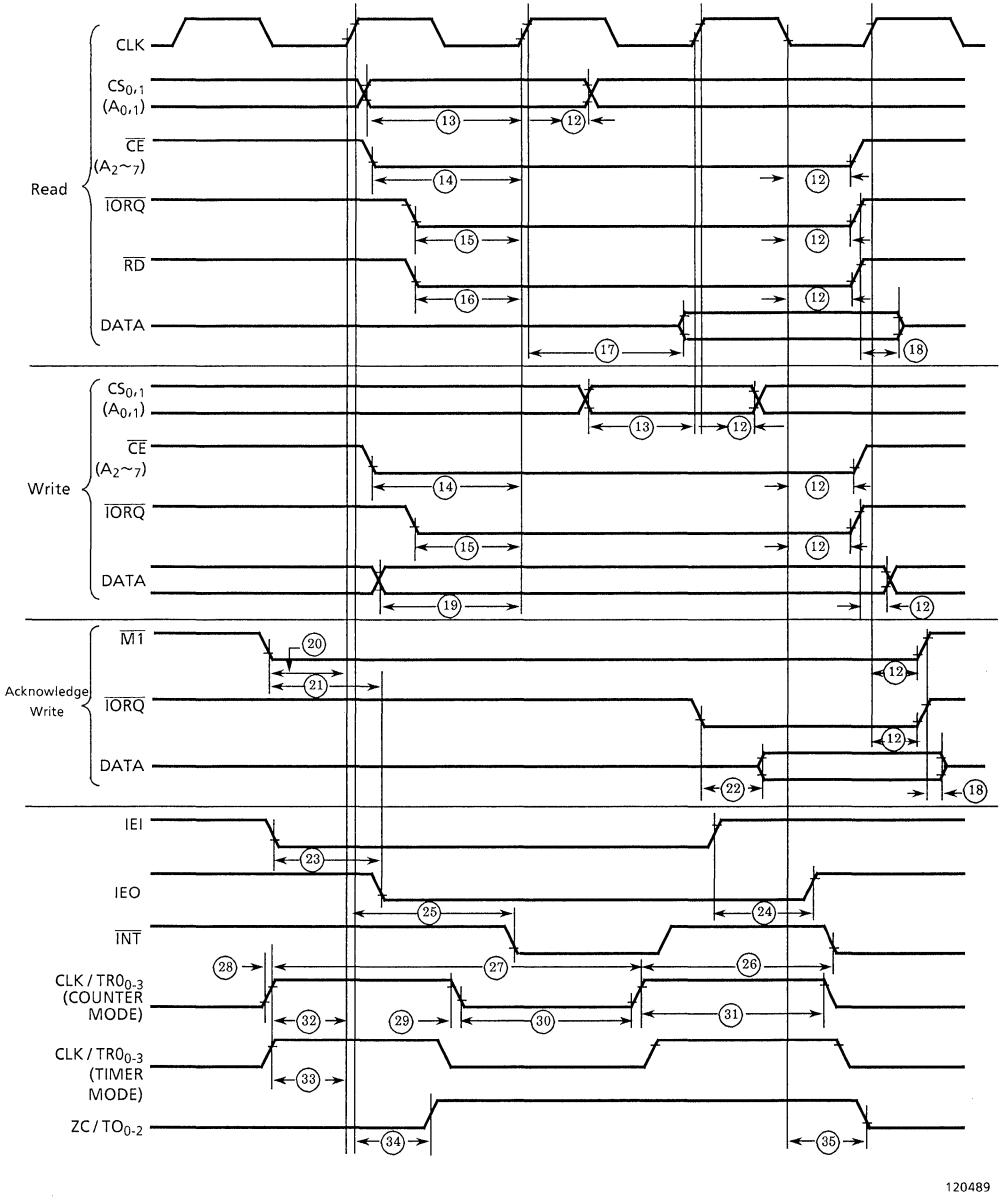


Figure 4.7.5 CTC Timing Diagram (Inactive)

4.7.3 AC Timing Charts of I/O (in Inactive State)

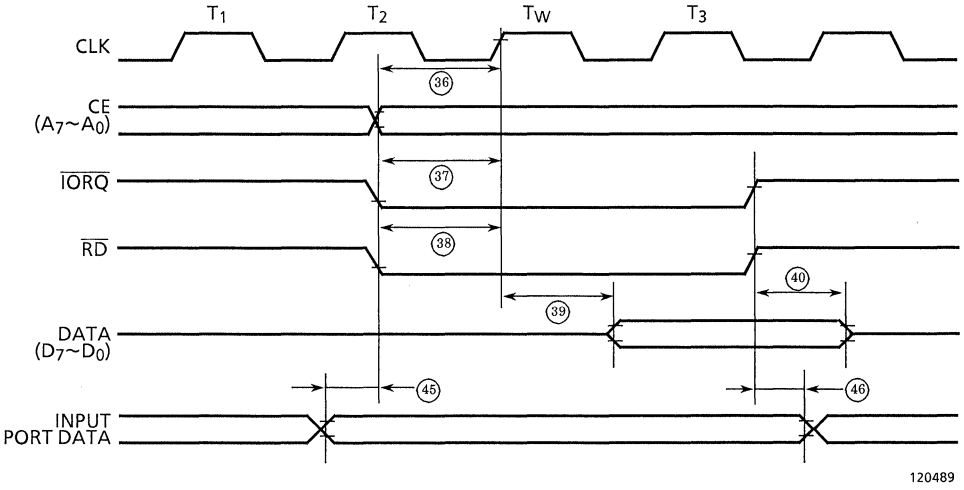


Figure 4.7.6 (a) I/O READ Timing

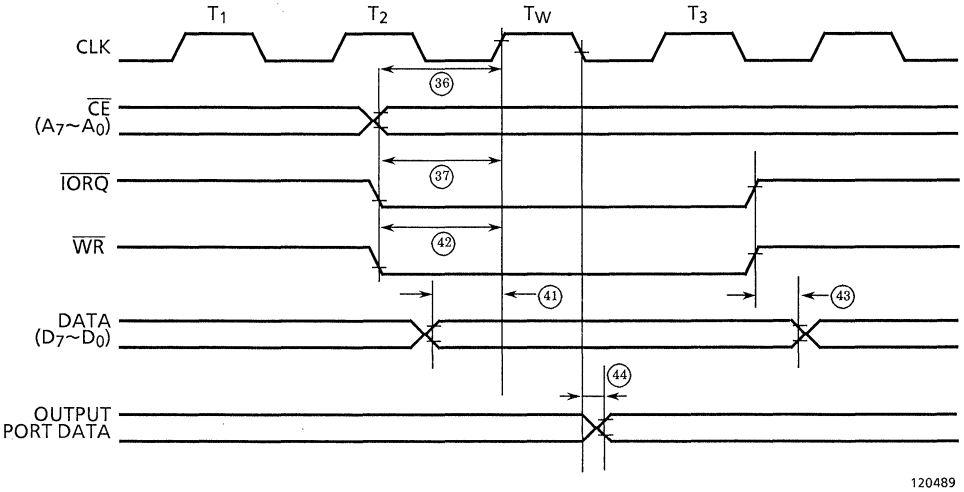
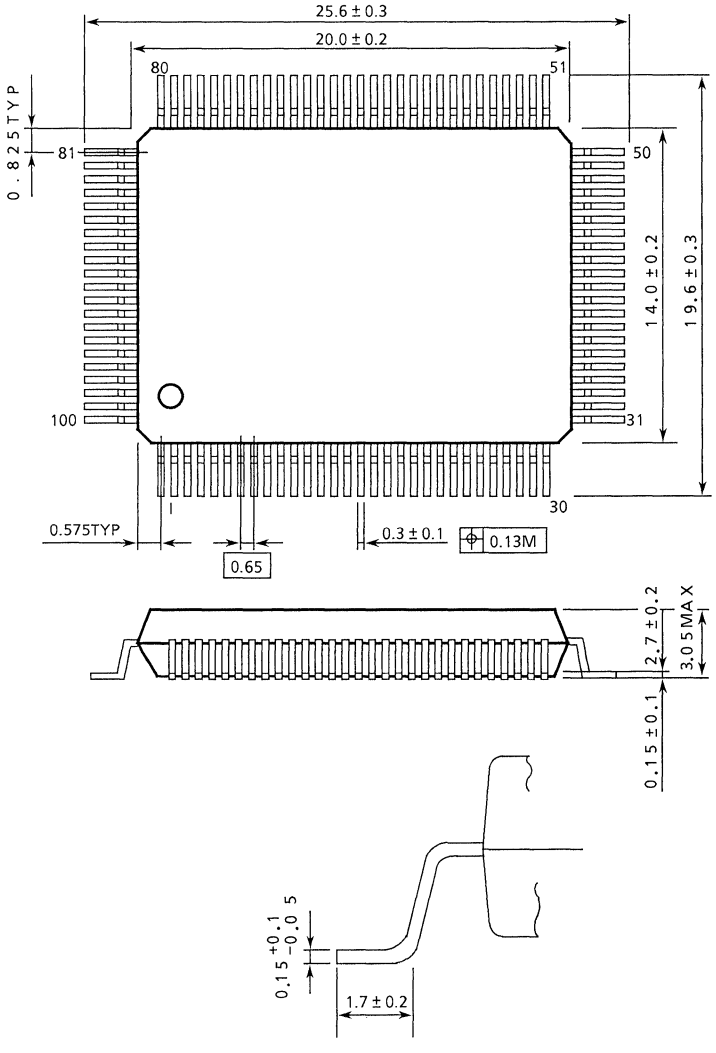


Figure 4.7.6 (b) I/O WRITE Timing

4.8 OUT LINE DRAWING

QFP100-P-1420B

Unit : mm



TMPZ84C013AT-6 / TMPZ84C013AT-8
TLCS-Z80 MICROPROCESSOR

1. OUTLINE AND FEATURES

The TMPZ84C013A is a high-performance CMOS 8-bit microprocessor incorporating the counter timer circuit (CTC), the serial I/O port (SIO), the clock generator/controller (CGC), and the watchdog timer (WDT) around the TLCS-Z80 MPU. This microprocessor inherits the basic architecture of the TLCS-Z80 series, allowing the user to utilize the software resources and development tools accumulated so far.

The TMPZ84C013A is based on the new CMOS process and housed in a standard 84-pin PLCC package, greatly contributing to system miniaturization and power saving.

The TMPZ84C013A incorporates the high-performance serial I/O port, the counter timer circuit which can be used as the baud rate generator, and the watchdog timer indispensable for control applications, offering the user a wide range of system applications such as the communication controllers including a communication adaptor and the various other controllers which need miniaturization.

Features

- Built-in TLCS-Z80 series MPU, CTC, SIO, CGC and watchdog timer features.
- High speed operation (6MHz and 8MHz operations)
- Built-in clock generator (CGC:Clock Generator Controller)
- Built-in standby capability (with the controller built in) provides 4 operation modes:

  Run mode (Normal operation)
  Idle-1 mode (Only clock oscillation goes on.)
  Idle-2 mode (Wake-up by CTC enabled.)
  Stop mode (Clock oscillation stopped; standby state)

- Wide operational voltage range (5V±10% : 6MHz VERSION, 5V±5% : 8MHz VERSION) supported.
- Wide operating temperature range (−40°C to +85°C : 6MHz VERSION, −10°C to +70°C : 8MHz VERSION)
- Low power dissipation

  In operation: (RUN mode) 22mA TYP. at 6 MHz, 31mA TYP. at 8 MHz
  In idle: (IDLE-1 mode) 1.5mA TYP. at 6 MHz, 2mA TYP. at 8 MHz
  (IDLE-2 mode) 11mA TYP. at 6 MHz, 15mA TYP. at 8 MHz
  In standby: (STOP mode) 500nA TYP.

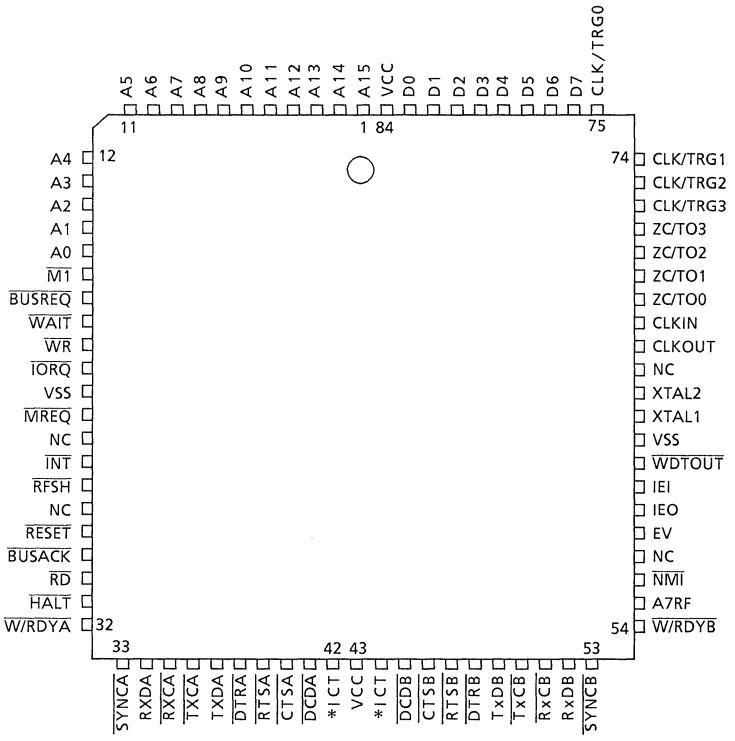
- Built-in TLCS-Z80 series SIO capability
- A pair of independent full duplex channels supports the asynchronous as well as synchronous byte-oriented (monosync and bisync) and bit-oriented HDLC and CCITT-X. 25 protocols.
- Built-in CRC generation and check capability.
- Data transfer rates of up to 1200 K bits/sec (6 MHz) and 1600 K bits/sec (8 MHz)
- Built-in TLCS-Z80 series CTC capability

  Four independent built-in channels.
  The timer or counter modes can be set .
  Also available as the SIO baud rate generator.
- Built-in watchdog timer
- Programmed daisy-chain interrupt control
- Built-in dynamic RAM refresh controller
- TTL/CMOS compatible
- Housed in compact standard 84-pin PLCC package
- The Toshiba real-time emulator (RTE80) and the commercially available Z80 ICE can be used (the TMPZ84C013A used as the evaluator).
- The Toshiba evaluator board installed.

note : Z80 is a trade mark of Zilog Inc.

2. PIN ASSIGNMENT AND FUNCTIONS

2.1 PIN ASSIGNMENTS (TOP VIEW)



Note : The ICT pin is the test pin. Do not make any external connection to this pin.

Figure 2.1.1 Pin Assignment

2.2 (A) PIN FUNCTIONS

(1/5)

Pin	Q'ty (Number)	Type	Function
D0-D7	8 (76-83)	Input / output 3-state	The 8-bit bi-directional data bus.
A0-A15	16 (1-16)	Output 3-state	The 16-bit address bus. These pins specify memory and I/O port addresses. During a refresh cycle, the refresh address is output to the low-order 7 bits and A7RF.
$\overline{M1}$	1 (7)	Output 3-state	The Machine Cycle 1 signal. In an operation code fetch cycle, this pin goes "0" with the $\overline{MREQ}$ signal. At the execution of a 2-byte operation code, this pin goes "0" for each operation code fetch. In a maskable interrupt acknowledge cycle, this pin goes "0" with the $\overline{IORQ}$ signal. When the EV signal is applied, this pin is put in the high-impedance state.
$\overline{RD}$	1 (30)	Output 3-state	The Read signal. It indicates that the MPU is ready for accepting data from memory or I/O device. The data from the addressed memory or I/O devices is gated by this signal onto the MPU data bus. When the $\overline{BUSREQ}$ signal is applied, this pin is put in the high-impedance state.
$\overline{WR}$	1 (20)	Output 3-state	The Write signal. This signal is output when the data to be stored in the addressed memory or I/O device is on the data bus. When the $\overline{BUSREQ}$ signal is applied, this pin is put in the high-impedance state.
$\overline{MREQ}$	1 (23)	Output 3-state	The Memory Request signal. When the execution address for memory access is on the address bus, this pin goes "0" . During a memory refresh cycle, this pin also goes "0" with $\overline{RFSH}$ signal.
$\overline{IORQ}$	1 (21)	Output 3-state	The Input/Output Request signal. This pin goes "0" when the address for an I/O read or write operation is on the low-order 8 bits (A0 through A7) of the address bus. The $\overline{IORQ}$ signal is also output with the $\overline{M1}$ signal at interrupt acknowledge to tell an I/O device that the interrupt response vector can be placed on the data bus. Note that the interrupt priority among the TMPZ84C013A CTC, and SIO is selected by a program.
IEO	1 (59)	Output	The Interrupt Enable Output signal. In the daisy chain interrupt control, this signal controls the interrupt from the peripheral LSIs connected next to the TMPZ84C013A. The IEO pin goes "1" only when the IEI pin is "1" and the MPU is not servicing an interrupt from the built-in peripheral LSI.

190589

(2/5)

Pin	Q'ty (Number)	Type	Function
XTAL1 XTAL2	2 (63) (64)	Input Output	The crystal oscillator connection. Connects an oscillator having the oscillation frequency 2 times as high as the system clock (CLKOUT) frequency.
CLKIN	1 (67)	Input	The Single-phase Clock Input. When the clock input is placed in the DC state (continued "1" or "0" level), this pin stops operating and holds the state of that time. Normally, this pin is connected with the CLKOUT pin. However, to operate the system with the external clock, connect the external clock to the CLKIN pin.
CLKOUT	1 (66)	Output	The Single-phase Clock Output. When a Halt instruction is executed in the Stop or Idle-1 mode, the CLKOUT output is retained at "0". In the Run and Idle-2 modes, the clock is kept output. This pin provides the clock to other peripheral ICs.
$\overline{\text{RESET}}$	1 (28)	Input	The Reset signal input. This signal resets the internal states of the TMPZ84C013A. This signal is also used to return from the standby state in the Stop or Idle mode.
$\overline{\text{INT}}$	1 (25)	Input	The Maskable Interrupt signal. An interrupt is caused by the internal CTC, SIO or the peripheral LSI. An interrupt is acknowledged when the interrupt enable flip-flop (IFF) is set to "1" by software. The $\overline{\text{INT}}$ pin is normally wire-ORed and requires an external pullup resistor for these applications. This signal is also used to return from the stand-by state in the Stop or Idle mode.
$\overline{\text{WAIT}}$	1 (19)	Input	The Wait Request signal. This signal indicates the MPU that the addressed memory or I/O device is not ready for data transfer. As long as this signal is "0", the MPU is in the Wait state.
$\overline{\text{BUSREQ}}$	1 (18)	Input	The bus Request signal. The $\overline{\text{BUSREQ}}$ signal forces the MPU address bus, data bus, and control signals $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, and $\overline{\text{WR}}$ to be placed in the high-impedance state. This signal is normally Wire-ORed and requires an external pullup resistor for these applications.
$\overline{\text{BUSACK}}$	1 (29)	Output	The Bus Acknowledge signal. In response to the $\overline{\text{BUSREQ}}$ signal, the $\overline{\text{BUSACK}}$ signal indicates to the requesting peripheral LSI that the MPU address bus, data bus, and control signals $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$ and $\overline{\text{WR}}$ have been put in the high-impedance state.

190589

(3/5)

Pin	Q'ty (Number)	Type	Function
$\overline{HALT}$	1 (31)	Output 3-state	The Halt signal. This pin goes "0" when the MPU has executed a Halt instruction and is in the Halt state. It is put in the high-impedance state when the EV signal is applied.
$\overline{RFSH}$	1 (26)	Output	The refresh signal. When the dynamic memory refresh address is on the low-order 8 bits of the address bus, this signal goes "0" . At the same time, the $\overline{MREQ}$ signal also goes active ("0"). This pin is put in the high-impedance state when the EV signal is applied.
CLK/TRG0 ~ CLK/TRG3	4 (72-75)	Input	The external clock/timer trigger. These 4 CLK/TRG pins correspond to 4 channels. In the counter mode, the down counter is decremented by 1 and in the timer mode, the timer is activated at each active edge (a rising or falling edge) of the signal which are input by these pins. It can be selected by program whether the active edge is a rising or falling edge.
ZC/TO0 ~ZC/TO3	4 (68-71)	Output	The Zero Count/Timer Out signal. In either the Timer mode, or counter mode, pulses are output from these pins when the down counter has reached zero.
IEI	1 (60)	Input	The Interrupt Enable Input signal. This signal is used with the IEO to form a priority daisy chain when there is more than one interrupt-driven peripheral LSI.
$\overline{NMI}$	1 (56)	Input	The Non-maskable Interrupt Request signal. This interrupt request has a higher priority than the maskable interrupt and is not dependent on the interrupt enable flip-flop (IFF) state. This signal is also used to return from the stand-by state in the Stop or Idle mode.
EV	1 (58)	Input	The Evaluator signal. When this signal is active, the $\overline{M1}$, $\overline{HALT}$, and $\overline{RFSH}$ pins are put in the high-impedance state. In using the TMPZ84C013A as an evaluator chip, the MPU is electrically disconnected (put in the high-impedance state) after one machine cycle is executed with the EV signal being "1" and the $\overline{BUSREQ}$ signal being "0", and follows the instructions from other MPU (such as the MPU of ICE). The signals of the disconnected MPU are A0 through A15, D0 through D7, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$, $\overline{M1}$, $\overline{HALT}$, and $\overline{RFSH}$. $\overline{BUSACK}$ needs to be disconnected by an externally connected circuit. The evaluator board is available to use the TMPZ84C013A as an evaluator chip.

190589

(4/5)

Pin	Q'ty (Number)	Type	Function
A7RF	1 (55)	Output	The 1-bit auxiliary address bus. This pin outputs the same signal as the bit 7 (A7) of the address bus. However, during a refresh cycle, this pin outputs the address which is the most significant bit of the 8-bit refresh address signal linked to the low-order 7 bits of the address bus.
$\overline{W/RDYA}$ $\overline{W/RDYB}$	2 (32, 54)	Output	The Wait/Ready signal A and the Wait/Ready signal B. These signals can be used as the Wait or Ready depending on SIO programming. When these signals are programmed as "Wait", they go active at "0" to indicate to the MPU that the addressed memory or I/O devices are not ready for data transfer, requesting the MPU to wait. When these signals are programmed as "Ready", they go active at "0" to determine when a peripheral device associated with a DMA port is ready for a read or write data. The DMA is requested to transfer data.
$\overline{SYNCA}$ $\overline{SYNCB}$	2 (33, 53)	Input / output	The Synchronization signal. In the asynchronous receive mode, these signals act as the $\overline{CTS}$ and $\overline{DCD}$ signals. In the external sync mode, these signals act as inputs and in the internal sync mode, they act as outputs.
RXDA RXDB	2 (34, 52)	Input	The Serial Receive Data signal.
$\overline{RXCA}$ $\overline{RXCB}$	2 (35, 51)	Input	The Receive Clock signal. In the asynchronous mode, the Receive Clocks may be 1, 16, 32 or 64 times the data transfer rate.
$\overline{TXCA}$ $\overline{TXCB}$	2 (36, 50)	Input	The Transmitter Clock signal. In the asynchronous mode, the Transmitter Clocks may be 1, 16, 32, or 64 times the data transfer rate.
TXDA TXDB	2 (37, 49)	Output	The serial transmit data signal.
$\overline{DTRA}$ $\overline{DTRB}$	2 (38, 48)	Output	The Data Terminal Ready signal. These signals indicate whether the data terminal is ready to receive serial data or not. When it is ready, these signals go active to enable the transmitter of the terminal. When it is not ready, these signals go inactive to disable the transfer from the terminal.
$\overline{RTSA}$ $\overline{RTSB}$	2 (39, 47)	Output	The Request to Send signal. These pins are "0" when transmitting serial data. That is, to transmit data, these signals are made active to enable their receivers.

190589

(5/5)

Pin	Q'ty (Number)	Type	Function
$\overline{CTSA}$ $\overline{CTSB}$	2 (40, 46)	Input	The Clear To Send signal. When these pins are "0", the modem having transmitted these signals is ready to receive serial data. When it is ready, these signals go active to enable the transmitter of the terminal. When it is not ready, these signals go inactive to disable the transfer from the terminal.
$\overline{DCDA}$ $\overline{DCDB}$	2 (41, 45)	Input	The Data Carrier Detect signal. When these pins are "0" the receive of serial data can be enabled. That is, to transmit data, these signals are made active to enable their receivers.
ICT	2 (42, 44)	Output	The test pins. To be used in the open state.
$\overline{WDTOUT}$	1 (61)	Output	The Watchdog Timer signal. The output pulse width depends on the externally connected pin.
VCC	2 (43, 84)		The power supply (+ 5 V) pins.
VSS	2 (22, 62)		The ground (0 V) pins.

190589

2.2 (B) TMPZ84C013A Internal I/O Address Map

Internal I/O	Channel	I/O Address
CTC (Counter Timer)	ch 0	# 10
	ch 1	# 11
	ch 2	# 12
	ch 3	# 13
SIO (Serial I/O)	ch A Send/Receive buffer	# 18
	ch A Command/Status Register	# 19
	ch B Send/Receive buffer	# 1A
	ch B Command/Status Register	# 1B
Watch Dog Timer Stand-by mode Register	WDTER, WDTPR, HALTMR	# F0
Watch Dog Timer command Register	Clear command ($4E_H$) disable command ($B1_H$)	# F1
Daisy-chaine interrupt precedence Register	Only use bit2~bit0	# F4

190589

3. OPERATIONAL DESCRIPTION

3.1 BLOCK DIAGRAM AND OPERATIONAL OUTLINE

3.1.1 Block Diagram

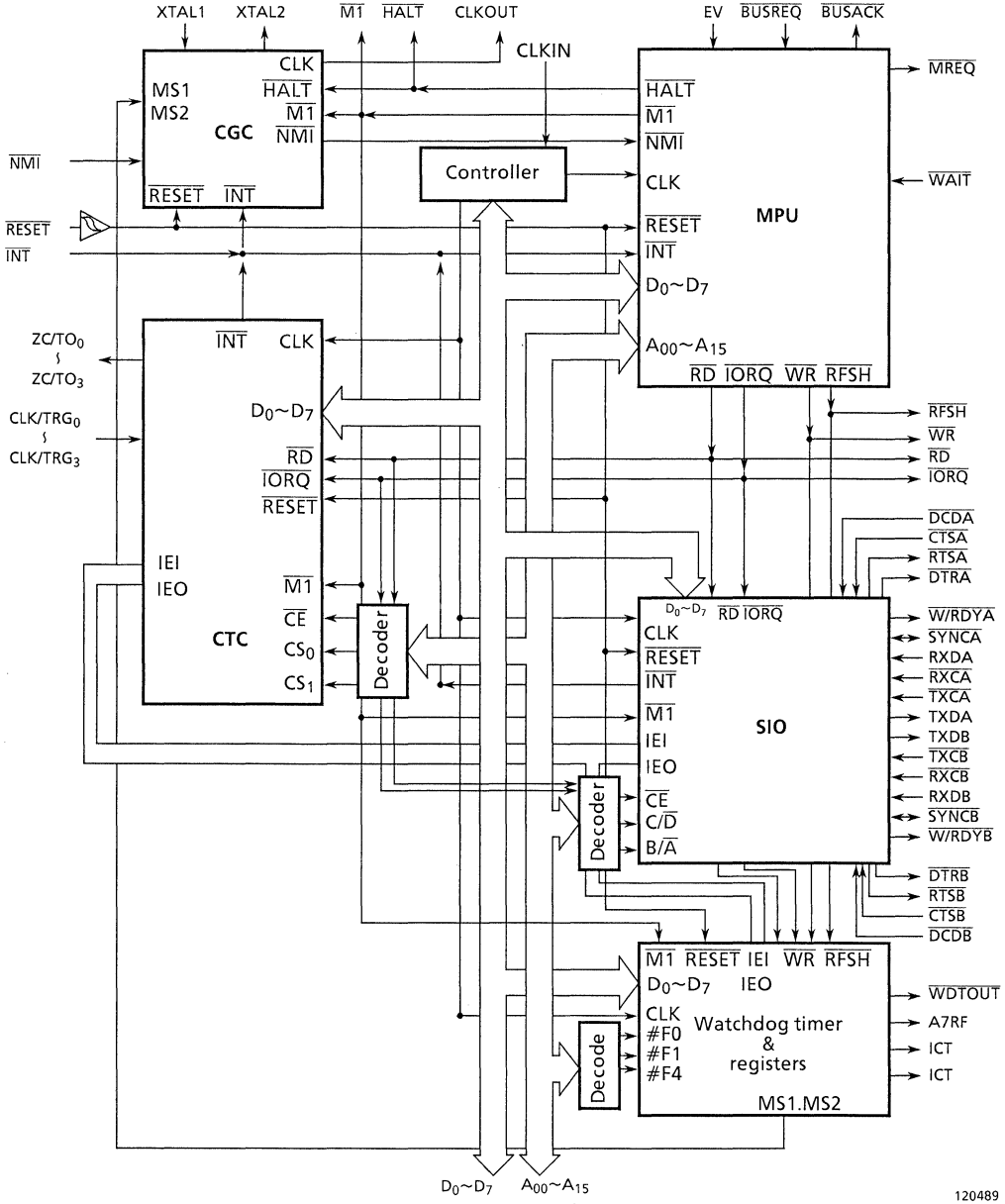


Figure 3.1.1 Block Diagram of TMPZ84C013A

3.1.2 Operational Outline

The TMPZ84C013A largely consists of a processor (MPU), a counter/timer circuit (CTC), a serial input/output controller (SIO), a watchdog timer (WDT), and a clock generator/controller (CGC).

- The MPU provides all the capabilities and pins of the Toshiba TLCS-Z80 MPU (TMPZ84C00A) to play the role of the TLCS-Z80 microprocessor perfectly.
- The CTC provides the capabilities of the Toshiba TLCS-Z80 CTC (TMPZ84C30A) and has the pins required to perform the necessary operations as a TLCS-Z80 peripheral LSI. The four independent timer channels are I/O-addressed internally.
- The SIO provides the capabilities of the Toshiba TLCS-Z80 SIO (TMPZ84C43A) and has the pins required to perform the necessary operations as a TLCS-Z80 peripheral LSI. The two independent serial channels are I/O-addressed internally.
- The WDT incorporates one-channel watchdog timer and the read/write-enabled watchdog timer control registers indispensable for control applications. The WDT also has the register to determine interrupt priorities, allowing the user to set the daisy-chain interrupt priorities by program. Additionally, the WDT has the IEI and IEO pins required to process the daisy- chain interrupts caused by the peripheral LSIs to be added both inside and outside the TMPZ84C013A.
- The CGC provides the four operation modes to control the entire TMPZ84C013A chip; the Run, Idle-1, Idle-2, and Stop modes. They are program-selectable. This chip has two clock pins: CLKOUT to supply clock from the CGC and CLKIN to get clock from the outside. Therefore, the TMPZ84C013A can be operated on the clock supplied from the outside at the CLKIN pin without using the CGC. The following briefly describes the four operation modes of the CGC with the CLKOUT and CLKIN pins connected:
- In the Run mode, the clock generated by the CGC is supplied to the TMPZ84C013A and peripheral LSIs to perform the normal programmed microcomputer operations.
- In the Idle-1 mode, clock oscillation is going on but the clock is not supplied to the TMPZ84C013A and peripheral LSI, thereby saving the system power consumption and shortening the time required for system restart.

- In the Idle-2 mode, clock oscillation is performed and the clock is output from the CLKOUT pin. The clock is supplied only to the CTC in the TMPZ84C013A, permitting a wake-up operation by the CTC. Like the Idle-1 mode, the Idle-2 mode saves the system power consumption and shortens the time required for system restart.
- In the Stop mode, clock oscillation is not performed and the system operation can be stopped completely. In this mode, the system can be restarted with the internal data retained with an extremely low power consumption level unique to the CMOS technology.

Note that these modes can be set only when the MPU has executed a HALT instruction.

Additionally, the TMPZ84C013A has also the EV pin which is used with the $\overline{\text{BUSREQ}}$ pin to put the MPU in the high-impedance state for electrical disconnection, thus functioning as an evaluator chip. That is, the MPU in the TMPZ84C013A is electrically disconnected by these two pins to implement the emulation by the signal from the in-circuit emulator (ICE). For emulation, one machine cycle is performed on the MPU in the TMPZ84C013A with EV being "1" and the $\overline{\text{BUSREQ}}$ being "0" then the emulation is performed as instructed by the MPU. The MPU signals to be disconnected are A0 through A15, D0 through D7, $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, $\overline{\text{M1}}$, $\overline{\text{HALT}}$, and $\overline{\text{RFSH}}$, $\overline{\text{BUSACK}}$ needs to be disconnected by an externally connected circuit.

3.2 MPU OPERATIONS

This subsection describes the system configuration, functions and the basic operations of the MPU of the TMPZ84C013A.

3.2.1 Block Diagram

Figure 3.2.1 shows the block diagram of the MPU.

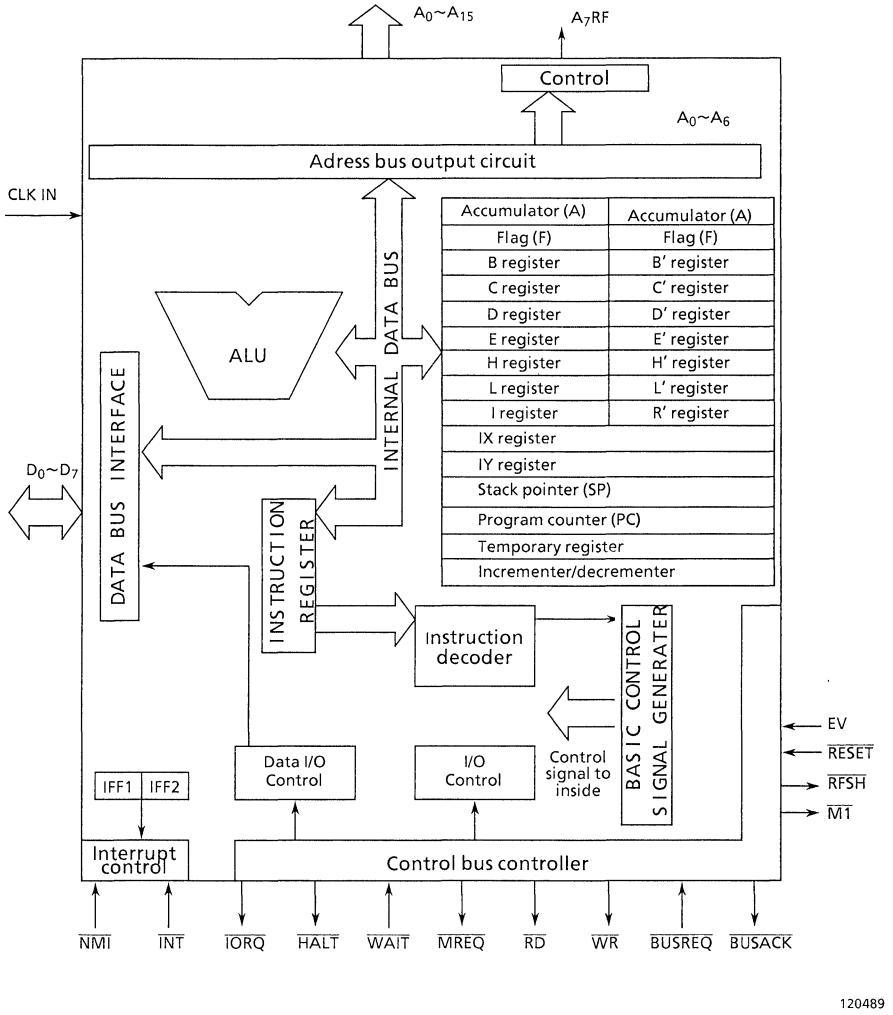


Figure 3.2.1 MPU Block Diagram

3.2.2 MPU System Configuration

The MPU has the configuration shown in Figure 3.2.1. The address signal is put on the address bus via the address buffer. The data bus is controlled for input or output by the data bus interface. Both the address and data buses are put in the high-impedance state by the $\overline{\text{BUSREQ}}$ signal input to make them available for other peripheral LSIs. The Opcode read from memory via the data bus is written to the instruction register. This Opcode is decoded by the instruction decoder. According to the result of the decoding, control signals are sent to the relevant devices. Receiving these control signals, the ALU performs various arithmetic operations. The register array temporarily hold the information required to perform operation.

The following describes the MPU's main components and functions which the user must understand to operate the TMPZ84C013A:

[1] Internal Register Groups

The configuration of the internal register groups is as follows:

(1) Main registers

A, F, B, C, D, E, H, L

(2) Alternate registers

A', F', B', C', D', E', H', L'

(3) Special purpose registers

I, R, IX, IY, SP, PC

Figure 3.2.3 shows the configuration of the internal register groups. The register groups, each being of a static RAM, consists of eighteen 8-bit registers and four 16-bit registers. The following describes the function of each register:

(1) Main registers (A, F, B, C, D, E, H, L)

(a) Accumulator (A)

The accumulator is an 8-bit register used for arithmetic and data transfer operations.

(b) Flag register (F) (see Figure 3.2.2)

The flag register is an 8-bit register to hold the result of each arithmetic operation. Actually, the 6 of the 8 bits are set ("1")/reset ("0") according to the condition specified by an instruction.

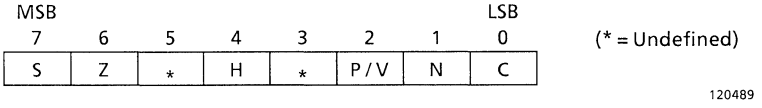


Figure 3.2.2 Flag Register Configuration

The following 4 bits are directly available to the programmer for setting the jump, call and return instruction conditions:

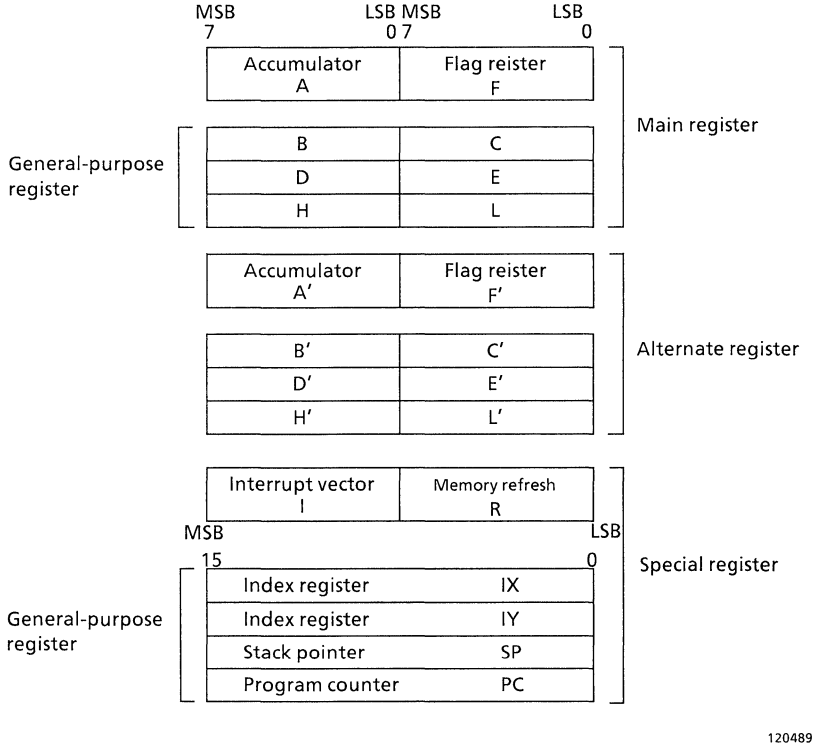


Figure 3.2.3 Flag Register Configuration

- Sign flag (S)

  When the result of an operation is negative, the S flag is set to "1" . Actually, the content of bit 7 of accumulator is stored in this flag.

- Zero flag (Z)

  When all bits turn out to be "0" s after operation, the Z flag is set to "1" . Otherwise, it is set to "0". With a block search instruction (CPI, CPIR, CPD or CPDR), the Z flag is set to "1" if the source data and the accumulator data match. With a block I/O instruction (INI, IND, OUTI or OUTD), the Z flag is set to "1" if the content of the B register used as the byte counter is "0" at the end of comparison.

- Parity/overflow flag (P/V)

  This flag has two functions. One is the parity flag (P) that indicates the result of a logical operation (AND A, B etc.). The P flag is set to "1" if the parity is even as a result of the operation on signed values by two's complement. It is reset to "0" if the parity is odd. With a block search instruction (CPI, CPIR, CPD or CPDR) and a block transfer instruction (LDI or LDD), the P flag indicates the state of the byte counter (register pair B and C). It is set to "1" if the byte counter is not "0" and reset to "0" when the byte counter becomes "0" (at the end of comparison or data transfer). The content of the interrupt enable flip-flop (IFF) is saved to the P flag when the contents of the R register or I register are transferred to the accumulator.

  The other use of the P/V flag is the overflow flag (V) that indicates whether an overflow has occurred or not as a result of an arithmetic operation. The V flag is set to "1" when the value in the accumulator gets out of a range of the maximum value +127 and the minimum value −128 and therefore cannot be correctly represented as a two's complement notation.

  Whether the P/V flag operates as the P flag or V flag is determined by the type of the instruction executed.

- Carry flag (C)

  The C flag is set to "1" if a carry occurs from bit 7 of the accumulator or a borrow occurs as a result of an operation.

  The following two flags are not available to the programmer for the test and set ("1")/reset ("0") purposes. They are internally used by the MPU for BCD arithmetic operations.

- Half carry flag (H)

  The H flag is used for holding the carry or borrow from the low-order 4 bits of a BCD operation result. When a DAA instruction (decimal adjust) is executed, the MPU automatically uses the H flag to adjust the result of a decimal addition or subtraction.

- Add/subtract flag (N)

  In BCD operation, algorithm is different between addition and subtraction. The N flag indicates whether the executed operation is addition or subtraction.

  For change of the flag state depending on the instruction, see 3.2.4 "TMPZ84C013A Instruction Set".

(c) General-purpose registers (B, C, D, E, H, L)

General-purpose registers consist of 8 bits each. They are used as 16-bit register pairs (BC, DE, HL) as well as independent 8-bit registers to supplement the accumulator. The B register and the register pair BC are used as counters when a block I/O, block transfer, or search instruction is executed. The register pair HL has various memory addressing features as compared with the register pairs BC and DE.

(2) Alternate registers (A', F', B', C', D', E', H', L')

The configuration of the alternate registers is exactly the same as that of the main registers. There is no instruction that handles the alternate registers directly. The data in the alternate registers are processed by moving them into the main registers by means of exchange instructions as shown below:

EX AF, AF' (A↔A', F↔F')

EXX (B↔B', C↔C', D↔D', E↔E', H↔H', L↔L')

When a hign-speed interrupt response has been requested within the system, these instruction can be used to quickly move the contents of the accumulator, flag registers, and general-purpose registers into the corresponding registers. This eliminates the need for transferring the register contents to/from the external stack during execution of the interrupt handling routine, thereby shortening the interrupt servicing time greatly.

(3) Special purpose registers (I, R, IX, IY, SP, PC)

(a) Interrupt page address register (I)

The TMPZ84C013A provides two kinds of interrupts :maskable interrupt (INT) and non-maskable interrupt (NMI). The maskable interrupt provides three modes (0, 1, and 2) in which the interrupt is handled. These modes can be selected by instructions IM0, IM1, and IM2 respectively. In Mode 2, any memory location can be called indirectly depending on the interrupt. For this purpose, the I register stores the high-order 8 bits of the indirect address. The low-order 8 bits are supplied from the interrupting peripheral LSI. This scheme permits calling the interrupt handling routine from any memory location in an extremely short access time. For the details of interrupts, see [4] "Interrupt Capability".

(b) Memory refresh register (R)

The R register is used as the memory refresh counter when the dynamic RAM is used for memory. This permits using of the dynamic memory in the same manner as the static memory. This 8-bit register is automatically incremented for each instruction fetch. While the MPU decodes and executes the fetched instruction, the contents of the R register are synchronized with the refresh signal to place the low-order 7 bits and A7RF on the address bus. This operation is all performed by the MPU and, therefore, dose not need a special processing by program. The MPU operation is not delayed by this operation. During refresh, the contents of the I register are placed on the high-order 8 bits of the address bus.

(c) Index registers (IX, IY)

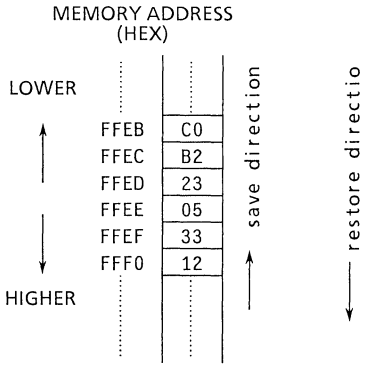
The two independent index registers IX and IY hold the 16-bit base address when used in the index addressing mode. In this addressing mode, the memory address obtained by adding the contents of an index register to the displacement value (for example, LD IX+40H) is specified. This mode is convenient for using data tables. Also these registers can be used separately for memory addressing and data retaining registers.

(d) Stack pointer (SP)

The stack pointer is a 16-bit register to provide the start address information in the stack area in the external RAM. The content of the stack pointer is decremented at the execution of a call instruction or PUSH instruction or interrupt handling and is incremented at the execution of a return instruction or POP instruction. At the execution of a call instruction or interrupt handling, the current content of the program counter is saved into the stack. At the execution of a return instruction, the content is restored from the stack to the program counter. These operations are all performed by the MPU automatically. However, the other registers are not saved or restored automatically. For the storing of the contents of these registers, an exchange instruction (EX or EXX) for alternate register, a PUSH or a POP instructions must be used. When a PUSH instruction is executed, the contents of the specified register are saved into the stack. When a POP instruction is executed, the contents of the stack are moved to the specified register.

These data are restored on a last-in, first-out basis. Use of the stack permits processing of multiple-level interrupts, deep subroutine nestings, and various data manipulation very easily. The stack pointer is not initialized in the hardware approach. Therefore, it is required to allocate the stack area in RAM to specify initialization (at the highest address of the stack area) in the initial program.

(ex)



MEMORY ADDRESS (HEX)	INSTRUCTION		The contents of the SP before the instruction is accepted.
1230	CALL	1500H	FFF1
1500	PUSH	AF	FFEF ; A=05, F=23
1501	PUSH	BC	FFED ; B=B2, C=C0
1600	POP	BC	FFEB
1601	POP	AF	FFED
1602	RET		FFEF

The foregoing example shows the stack pointer and stack operations in which the instructions starting with the CALL at address 1230H and ending with the RET at address 1602H have been executed. However, it is assumed that there is no instruction or interrupt other than shown above that uses the stack during the execution. When the value of the stack pointer before executing the CALL instruction at address 1230H indicates address FFF1H, address 1233H is stored at addresses FFF0H and FFEFH because the CALL instruction consists of 3 bytes, then the stack pointer is decremented. Similarly, the data are saved or restored sequentially according to the instructions. These stack and stack pointer operations are all performed automatically.

(e) Program counter (PC)

The program counter holds, in 16 bits, the memory address of the instruction to be executed next. The MPU fetches the instruction from the memory location indicated by the program counter. When the content of the program counter is put on the address bus, the program counter is incremented automatically. However, it is not incremented with a jump instruction, a call instruction, or interrupt processing. Instead, the specified new address is set on it. With a return instruction, the content restored from the stack is set on the program counter.

These operations are all performed automatically and therefore, no care is required for programming.

[2] Halt Capability

When a HALT instruction has been executed, the MPU is put in the halt state. The halt capability can be used to halt the MPU against the external interrupts, thereby reducing the power dissipation. In the halt state the states of MPU's internal registers are retained. The halt state is cleared by reset or when an interrupt is accepted. For the details of halt operation, see [3] "Basic Timing".

(1) Halt operation

When a HALT instruction has been executed, the MPU sets the $\overline{\text{HALT}}$ signal to "0" to indicate that the MPU is going to be put in the halt state. Actually, the MPU in the halt state automatically continues executing NOP instructions if there is the system clock input. However, the program counter is not incremented. This keeps the refresh signal generated when the dynamic memory is used. During halt, the MPU's internal states are retained. The TMPZ84C013A contains the clock generator/controller, easily implementing the clock input control for these halt operations.

(2) Releasing the halt state

The halt state is cleared by accepting an interrupt (the $\overline{INT}$ or $\overline{NMI}$ signal input) or by reset (the $\overline{RESET}$ signal input). When an interrupt is accepted, the halt state is cleared and the interrupt handling routine is executed. However, a maskable interrupt (INT) cannot be accepted unless the interrupt enable flip-flop (IFF) is set.

Note that when the halt state is cleared by the $\overline{RESET}$ signal, the MPU is reset and the program counter is set to "0".

[3] RESET Signal

Holding the $\overline{RESET}$ pin at the low level ("0") under the following conditions, the MPU's internal states are reset:

(1) The supply voltage level is within the operational voltage range.

(2) System clock stabilization.

(3) Holding the $\overline{RESET}$ signal at the low level ("0") for at least 3 full clock cycles. When the $\overline{RESET}$ signal goes high ("1"), the MPU starts executing instructions from address 0000H after at least 2T state dummy cycles.

When reset, the MPU performs the following processing:

(a) Program counter

0000H is set.

(b) Interrupt

The interrupt enable flip-flop (IFF) is reset to "0" to disable the maskable interrupt. For the maskable interrupt processing, mode 0 is specified.

(c) Control output

All control outputs are made inactive ("1"). Therefore, the halt state is also cleared.

(d) Interrupt page address register (I register)

The content of the R register becomes 00H.

(e) Refresh register (R register)

The content of the R register becomes 00H.

The contents of the registers other than above and the external memory do not change.

Therefore, they must be initialized as required.

[4] Interrupt Capability

The interrupt capability is used to suspend the execution of the current program and execute the processing of the requested peripheral LSI. Normally, this interrupt processing routine contains the data exchange and transfer of status and control information between the MPU and the peripheral LSI. When this routine has been completed, the MPU returns to the state before the interrupt has been accepted.

The TMPZ84C013A provides the non-maskable interrupt (NMI) and maskable interrupt (INT) capabilities which are detected by the $\overline{NMI}$ and $\overline{INT}$ interrupt request signals, respectively. A non-maskable interrupt, when caused by a peripheral LSI, is accepted unconditionally. This interrupt is used to support critical functions such as the protection of the system from unpredictable happening including power failure. A maskable interrupt can be enabled or disabled by program. For example, if the timer is used and, therefore, an interrupt is not desired, the system can be programmed to disable the interrupt. Table 3.2.1 lists the processing by interrupt source.

(1) Interrupt enable/disable

A non-maskable interrupt cannot be disabled by program, while a maskable interrupt can be enabled or disabled by program. The MPU has the interrupt enable flip-flop (IFF). A maskable interrupt can be enabled or disabled by setting this flip-flop to "1" (set) or "0" (reset) through an EI instruction (enable) or a DI instruction (disable) in program. Actually, the IFF consists of two flip-flops IFF1 and IFF2. IFF1 is used to select between the enable and disable of a maskable interrupt. IFF2 holds the state of IFF1 before a maskable interrupt has been accepted. Both IFF1 and IFF2 are reset to "0" when any of the following conditions occurs, disabling an interrupt:

- MPU reset
- Execution of DI instruction
- Acceptance of maskable interrupt

Both IFF1 and IFF2 are set to "1" when the the following condition occurs, enabling an interrupt:

- Execution of EI instruction

Actually, the waiting maskable interrupt request is accepted after the execution of the instruction that follows the EI instruction.
This delay by one instruction is caused by accepting an interrupt after completion of the execution of a return instruction if the instruction following the EI instruction is a return instruction.

In the above operation, the contents of IFF1 and IFF2 are the same.

Table 3.2.1 Processing by Interrupt Source

Interrupt Source	Priority	Programmed condition		Vector address	Interrupt return instruction
Non-maskable interrupt (the falling edge of $\overline{NMI}$)	1	None		Address 66H	RETN
Maskable interrupt ($\overline{INT}$ becomes "0" at instruction's last clock)	2	IFF = 1	Mode 0	Instruction from peripheral LSI. Normally, CALL or RST instruction.	(Note) RETI
			Mode 1	Address 38H.	
			Mode 2	The address indicated by the data table (memory) at the address specified by I register (high-order 8 bits) and data from peripheral LSI (low-order 8 bits, LSB = "0").	

120489

Note : Mode 0 applies when the instruction from peripheral LSI is CALL or RST instruction.

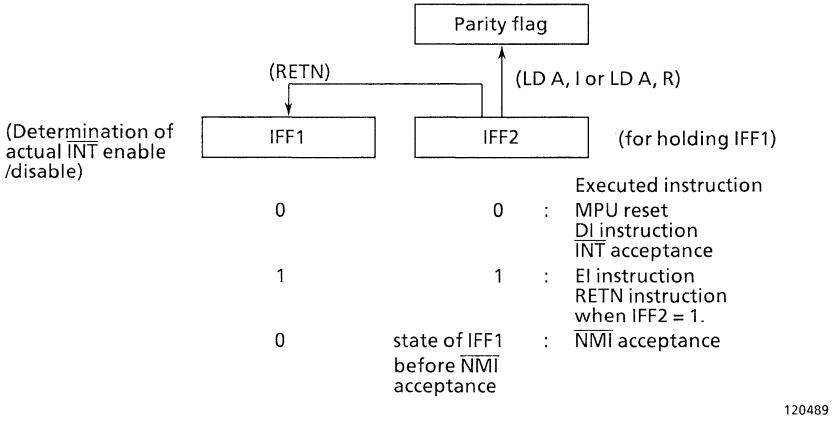


Figure 3.2.4 Interrupt Enable Flip-Flop (IFF)

When a non-maskable interrupt has been accepted, IFF1 is reset to "0" (interrupt disable) until an EI or RETN instruction is executed, so as to prevent from accepting the next interrupt. For this purpose, the state (interrupt enable/disable) of IFF1 immediately before non-maskable interrupt acceptance must be stored. This state is copied into IFF2 upon acceptance of a non-maskable interrupt. The content of IFF2 is copied into the parity flag at the execution of the following instructions, so that the copied data can be tested or stored:

- The load instruction (LD A, I) to load the contents of the I register into the accumulator.
- The load instruction (LD A, R) to load the contents of the R register into the accumulator.

When the return instruction (RETN) from the non-maskable interrupt is executed, the contents of the current IFF2 are copied back to IFF1. If an operation which changes the contents of IFF2 (due to the execution of EI or DI instruction, for example) has not been performed during interrupt handling, IFF1 automatically returns to the state immediately before the interrupt acceptance. Table 3.2.1 lists the states of IFF1 and IFF2 after execution of interrupt-related instructions.

Table 3.2.2 State of IFF1 and IFF2

Operation sequence	IFF1	IFF2	Remarks
MPU reset	0	0	
EI	1	1	
NMI acceptance	0	1	
LD A, I	*	*	Parity flag←IFF2
RETN	1	1	IFF1←IFF2
LD A, R	*	*	Parity flag←IFF2
INT acceptance	0	0	
RETI	*	*	
EI	1	1	
NMI acceptance	0	1	
DI	0	0	
RETN	*	*	

Note : * = no change. 120489

(2) Interrupt processing

With a non-maskable interrupt, the internal NMI flip-flop is set to "1" on the falling edge of the interrupt signal, $\overline{NMI}$. The state of this flip-flop is sampled on the rising edge of the last clock of each instruction to accept an interrupt. A maskable interrupt is accepted if the interrupt signal $\overline{INT}$ is low ("0") on the rising edge of the last clock of each instruction and the interrupt enable state (IFF = 1 and $\overline{BUREQ}$ signal = inactive ("1")) is on. The following is the processing to be performed after a non-maskable interrupt and a maskable interrupt are accepted:

(a) Non-maskable interrupt (NMI)

When a non-maskable interrupt has been accepted, the MPU performs the following processing:

1 The internal NMI flip-flop is reset to "0".

2 IFF1 is reset to "0", disabling the maskable interrupt.

The contents of the IFF1 immediately before the interrupt acceptance are copied into the IFF2.

3 The content of the current program counter is saved into the stack.

4 The instructions starting from non-maskable interrupt vector address 66H are executed.

The non-maskable interrupt processing program terminates after executing the RETN instruction. This return instruction performs the followings:

1 The contents of the current IFF2 are copied into IFF1.

2 The contents of the program counter are restored from the stack.

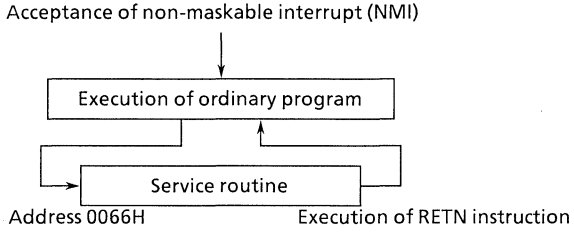


Figure 3.2.5 Non-Maskable Interrupt Processing

(b) Maskable interrupt (INT)

When a maskable interrupt has been accepted, the MPU performs the following processings:

1 Both IFF1 and IFF2 are reset to "0", disabling the maskable interrupts.

2 The content of the current program counter is saved into the stack.

3 A maskable interrupt is serviced in one of the three modes 0, 1 and 2. A mode is selected by executing the instruction IM0, IM1or IM2 before the interrupt is serviced. The instructions are executed starting from the vector address corresponding to the selected mode.

- Mode 0

In mode 0, the interrupting peripheral LSI puts a restart instruction (RST) or a call instruction (CALL) on the data bus and the MPU executes the interrupt service routine according to that instruction. At reset, this mode is automatically set.

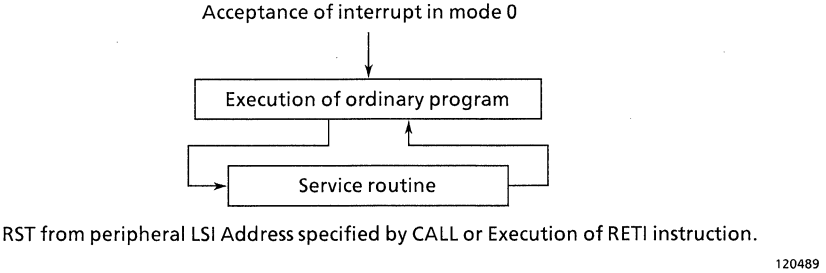


Figure 3.2.6 Interrupt Processing in Mode 0

- Mode 1

When an interrupt is accepted in mode 1, restart is performed from address 0038H. Therefore, the service routine for this interrupt is programmed from the address 0038H .

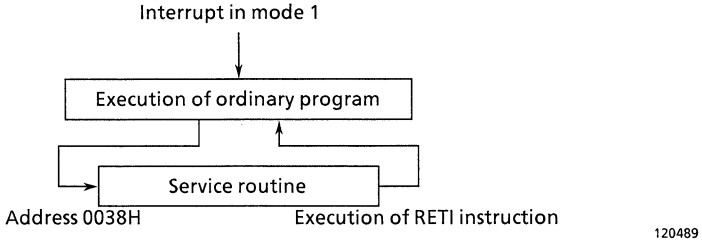


Figure 3.2.7 Interrupt Processing in Mode 1

- Mode 2

The interrupt processing in mode 2 requires a 16-bit pointer consisting of the high-order 8 bits of the I register and the low-order 8 bits (with the LSB="0") of the data fetched from the interrupting CTC or TLCS-Z80 family peripheral LSI. Therefore, the necessary value must be loaded in the I register beforehand. This pointer is used to specify the memory address in the table. The contents of the specified address and the next address provide the start address of the service routine. Therefore, use of this mode requires the table of the service routine's start address (16 bits) to be set at appropriate location under software control . This location can be anywhere in memory. The LSB of the table pointer is set to "0" because a 2-byte data is needed to specify the service routine start address in 16 bits and start that address from an even-number address. In the table, the start address begins with the low-order byte followed by the high-order byte as shown in Fig. 3.2.8.

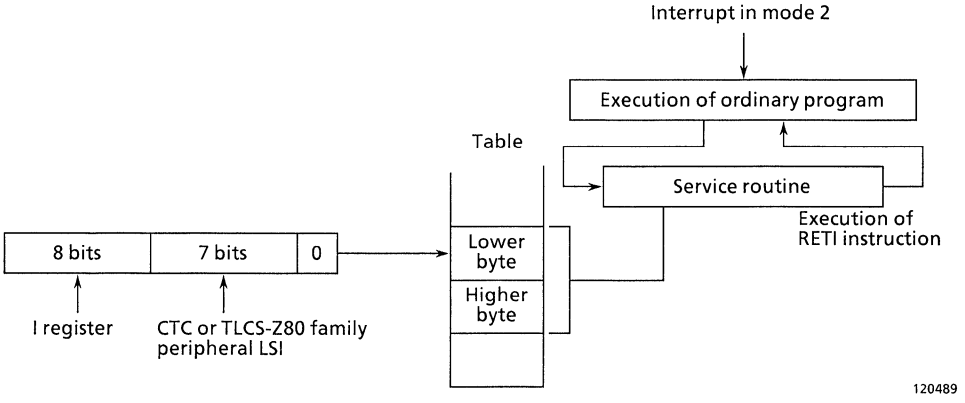


Figure 3.2.8 Interrupt Processing in Mode 2

Mode 2 is used in the daisy chain interrupt processing using the CTC and TLCS-Z80 family LSI. The CTC and TLCS-Z80 family peripheral LSIs all contain the interrupt priority controller in daisy chain structure. In this interrupt structure, the interrupt request signals are connected one after another and given priorities for processing when two or more maskable interrupt requests occur at a time. Only the interrupt vector from the peripheral LSI having the highest priority is put on the data bus. By receiving the interrupt vector in mode 2, the processing for that peripheral LSI can be performed. When an interrupt requested by a peripheral LSI having a priority higher than that of the current peripheral LSI during the execution of the interrupt processing routine, the higher priority interrupt can be enabled by the EI instruction to form an interrupt nesting.

The maskable interrupt processing program terminates by executing an RETI instruction. This return instruction performs the following processings:

- Restores the content of the program counter from the stack.
- Notifies the requesting peripheral LSI of the termination of interrupt processing.

3.2.3 MPU Status Transition Diagram and Basic Timing

The following describes the MPU status transition and the basic timing of each MPU operation.

[1] Instruction Cycle

Each TMPZ84C013A instruction is executed by combining the basic operations of memory read/write, input/output, bus request/acknowledge, and interrupt. These basic operations are performed synchronizing with the system clock (the CLK signal).

One clock period is called a state (T). The smallest unit of each basic operation is called a machine cycle (M). Each instruction consists of 1 to 6 machine cycles and each machine cycle consists of 3 to 6 clock states basically. However, the number of clock states in a machine cycle can be increased by the $\overline{WAIT}$ signal described later on. Figure 3.2.9 shows an example of the basic timing of a 3-machine-cycle instruction.

The first machine cycle (M1) of each instruction is the cycle in which the Opcode of the instruction to be executed next is read (this is called the Opcode fetch cycle). The Opcode fetch cycle basically consists of 4 to 6 clock states. In the machine cycle that follows the Opcode fetch cycle, data is transferred between the MPU and the memory or peripheral LSIs. This operation basically consists of 3 to 5 clock states.

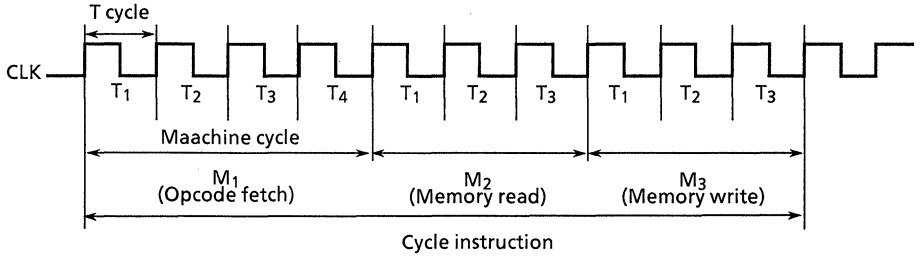


Figure 3.2.9 Example of MPU Basic Timing (3-Machine-Cycle Instruction)

[2] Status Transition Diagram

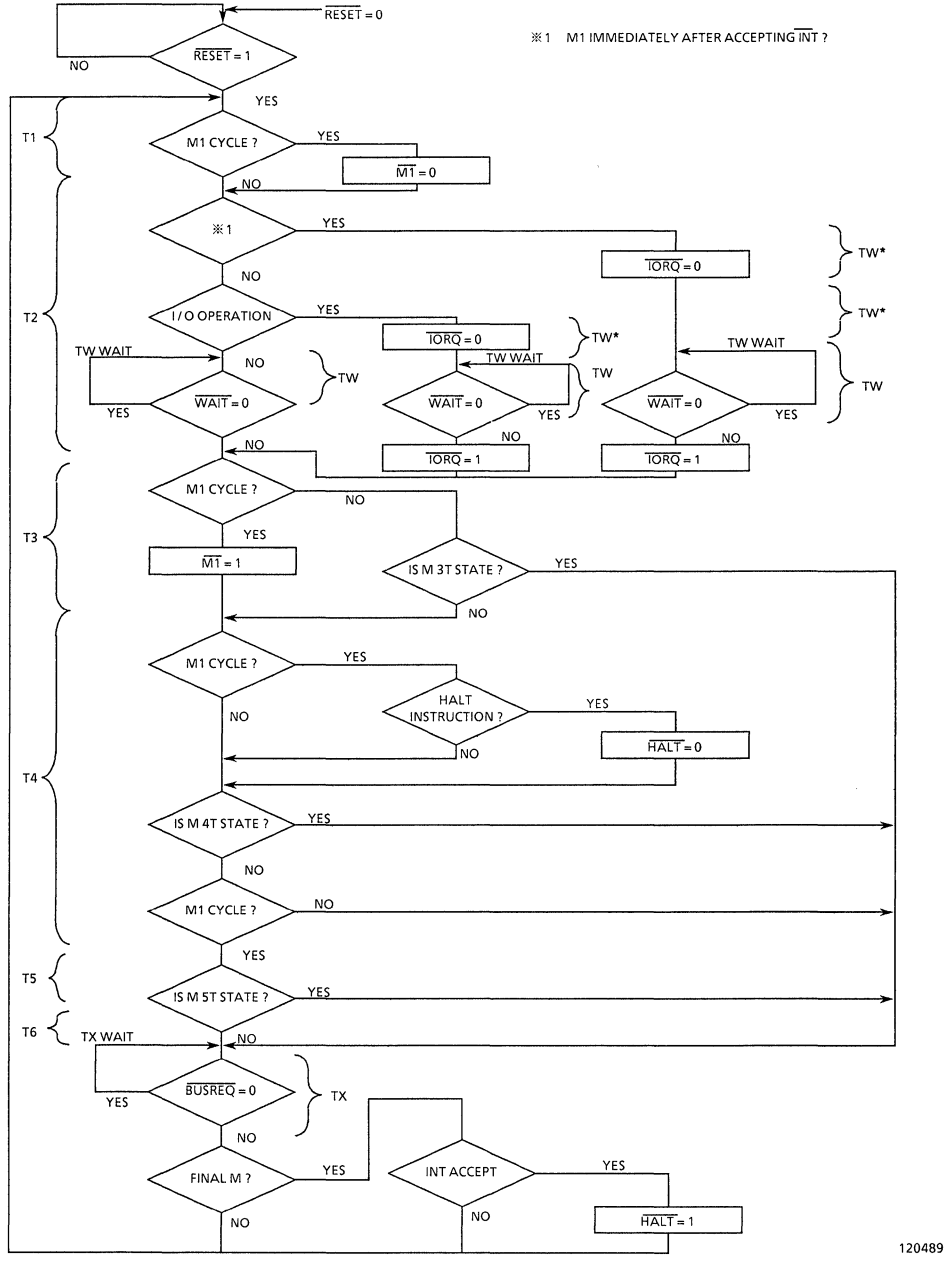


Figure 3.2.10 Status Transition Diagram

[3] Basic Timing

(1) Opcode fetch cycle (M1)

In the Opcode fetch cycle, MPU fetches an Opcode in the machine-language codes in memory. This is also called the M1 cycle because it is the first machine cycle to execute each instruction.

Figure 3.2.12 shows the basic timing of a basic Opcode fetch cycle.

In clock state T1, the content of the program counter is put on the address bus. The $\overline{M1}$ signal goes "0", indicating to the MPU that this is the Opcode fetch cycle. At the same time, $\overline{MREQ}$ and $\overline{RD}$ signals go "0". When the $\overline{MREQ}$ signal goes "0", the address signal has already been stabilized. Therefore, this signal can be used for the memory chip enable signal. The $\overline{RD}$ signal indicates that the MPU is ready to accept the data from memory. By these signals, the MPU accesses memory to fetch the Opcode in the instruction register. The MPU samples the $\overline{WAIT}$ signal on the falling edge of clock state T2. If the $\overline{WAIT}$ signal is "0" on the falling edge of clock state T2 and the following wait state (TW), the next state becomes clock state TW. Figure 3.2.13 shows the delay state of the Opcode fetch cycle caused by the $\overline{WAIT}$ signal.

The data (Opcode) on the data bus is fetched on the rising edge of clock state T3 then, the $\overline{MREQ}$, $\overline{RD}$, and $\overline{M1}$ signals go "1". In clock state T3, a memory refresh address is put on the 8 bits consisting of the low-order 7 bits of the address bus and the A7RF corresponding to bit 8 and the $\overline{RFSH}$ signal goes "0" and the $\overline{MREQ}$ signal goes "0" again. This signal indicates that the memory refresh cycle is on. At this time, the contents of the I register are put on the high-order 8 bits of the address bus and the 7 bits of the R register contents and the A7RF signal corresponding to bit 8 are put on the low-order 8 bits of the address bus. By using the $\overline{RFSH}$ and $\overline{MREQ}$ signals, memory refresh is performed in clock state T3 and T4. However, the $\overline{RD}$ signal remains "1" because the contents of the memory refresh address are not put on the data bus.

The address bus of 8 bits consisting of the low-order 7 bits of address (A6 through A0) and the A7RF are used as the 8-bits refresh address. That is, when A7RF is used for the refresh address, signals "00H" through "FFH" are output. In cycles other than the refresh cycle, the signal equivalent to A7RF are output. However, at reset, the signals to be output are uncertain. Figure 3.2.11 shows the refresh timing.

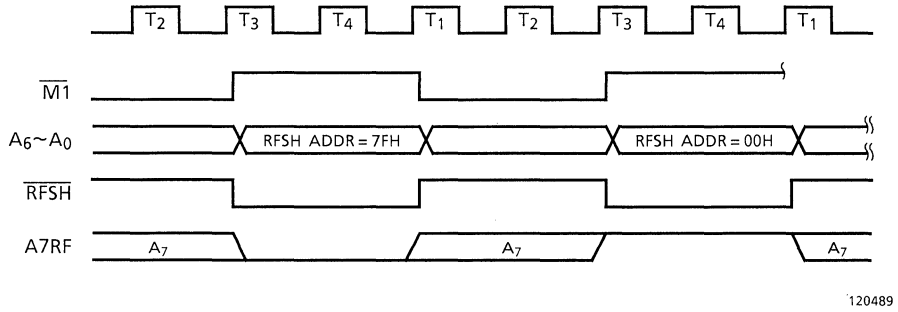


Figure 3.2.11 Refresh Timing

In clock state T4, the $\overline{MREQ}$ signal returns to "1". The refresh address is kept output until the rising edge of the clock state T1 in the next machine cycle, keeping the $\overline{RFSH}$ signal set to "0". The cycle delay state caused by setting the $\overline{WAIT}$ signal to "0" is the same in the memory read/write, input/output, and maskable interrupt acknowledge cycles. The diagram of the cycle delay state caused by the $\overline{WAIT}$ signal set to "0" is omitted in the following description.

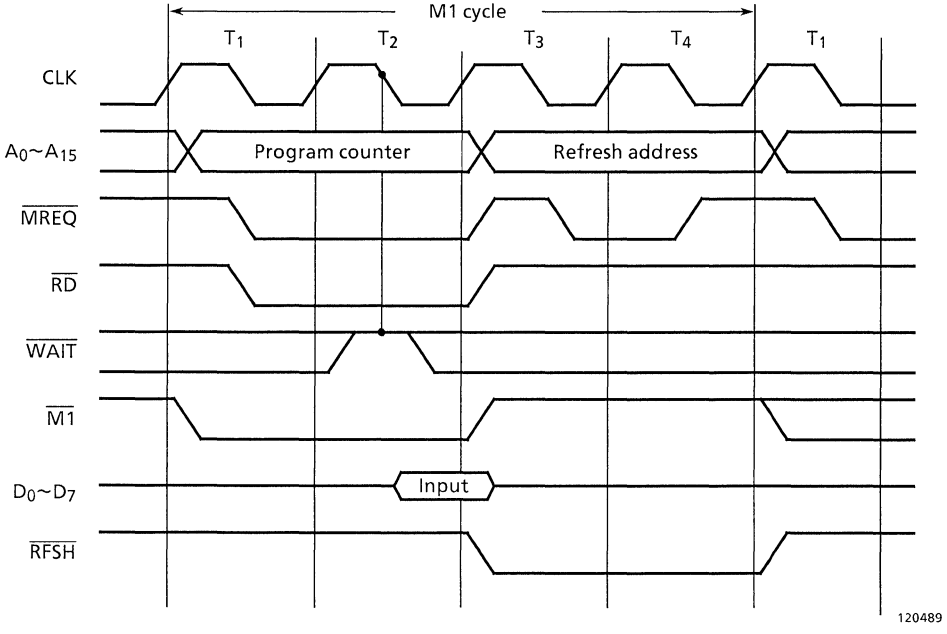


Figure 3.2.12 Opcode Fetch Timing

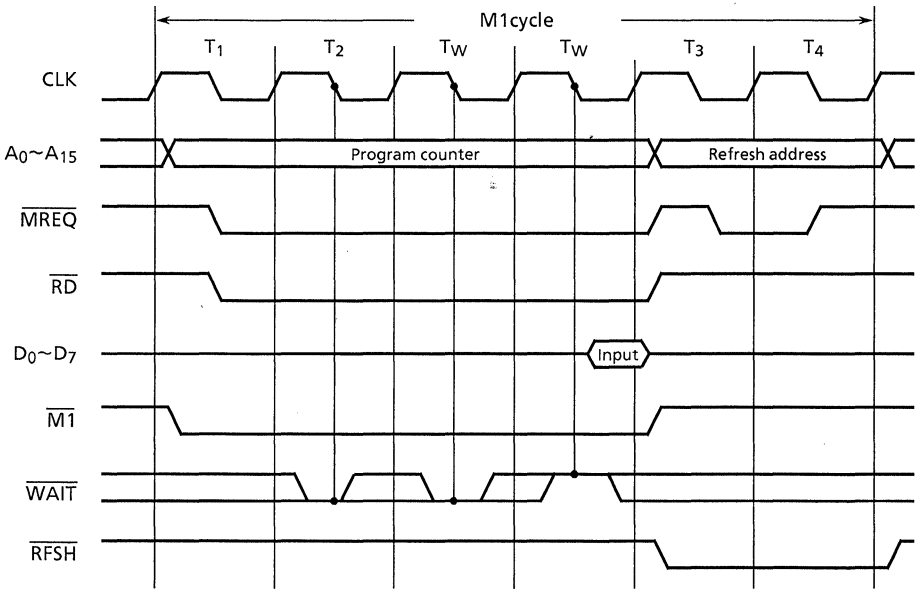


Figure 3.2.13 Opcode Fetch Timing Including Wait State

(2) Memory read/write operations

Figure 3.2.14 shows the basic timing of memory read/write operations (except for the Opcode fetch cycle) in the same diagram for convenience.

In each operation, the memory address signal to read/write data on the address bus is output in clock state T1. The operation in which the $\overline{WAIT}$ signal is sampled in clock state T2 and the following TW state is the same as the Opcode fetch cycle.

In memory read, memory data is put on the data bus by the address, $\overline{MREQ}$, and $\overline{RD}$ signals. The MPU reads this data.

In memory write, the memory address signal is put on the address bus then the $\overline{MREQ}$ signal is set to "0" to put the write data onto the data bus. When the data bus has been stabilized, the $\overline{WR}$ signal is output in clock state T2. The $\overline{WR}$ signal can be used as the memory write signal.

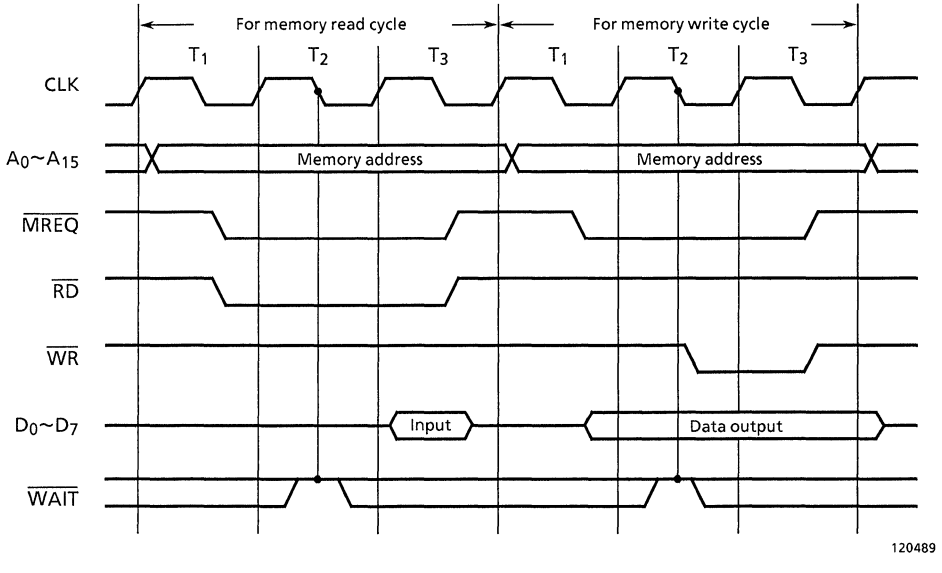


Figure 3.2.14 Memory Read/Write Cycle Timing

(3) Input/output operations

Figure 3.2.15 shows the basic timing of input/output operations. The feature of the I/O operation timing is that, regardless of the state of the $\overline{WAIT}$ signal in clock state T2, the I/O cycle automatically goes in the wait state (TW*) after clock T2. The $\overline{WAIT}$ signal is sampled on the falling edge of TW*. If the $\overline{WAIT}$ signal is "0" on the falling edges of TW* and the following clock state, the I/O operation enters into clock state TW*. Clock state TW* is inserted because the $\overline{IORQ}$ signal goes "0" in clock state T2, so that it is too late to sample the $\overline{WAIT}$ signal after decoding the I/O port address. In each of input and output operations, the I/O port address is put on the low-order 8 bits of the address bus in clock state T1. On the high-order 8 bits, the contents of the accumulator or B register are output. In clock state T2, the $\overline{IORQ}$ signal goes "0" instead of the $\overline{MREQ}$ signal. The $\overline{IORQ}$ signal can be used as the chip enable signal for a peripheral LSI.

In an input operation, the contents of the input port are read onto the data bus by the address, $\overline{IORQ}$, or $\overline{RD}$ signals. The MPU reads this data.

In an output operation, the output port address and the output data are respectively put on the address bus and data bus in clock state Tl, then the $\overline{IORQ}$ and $\overline{WR}$ signals go "0" in clock state T2. The $\overline{WR}$ signal can be used as the output port write signal.

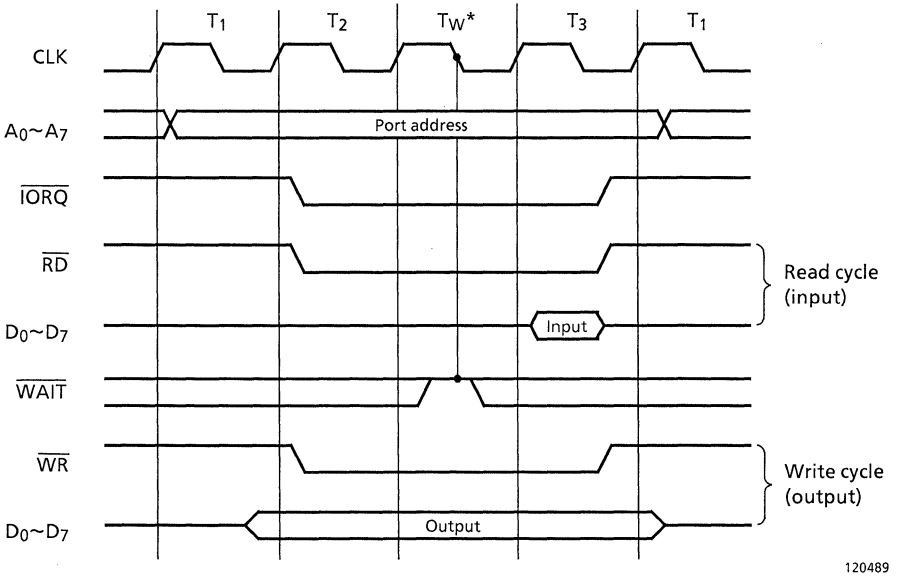


Figure 3.2.15 I/O Operating Timing

(4) Bus request and bus acknowledge operations

Figure 3.2.16 shows the basic timing of bus request and bus acknowledge operations.

The address bus (A0 through A15), data bus (D0 through D7), $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ signals controlled by the MPU can be put in the high-impedance state (floating) to electrically disconnect them from the MPU. This operation, after sampling the $\overline{BUSREQ}$ signal on the rising edge of the last clock of each machine cycle, starts on the rising edge of the next clock if this signal is found "0".

Subsequently, these buses are controlled by external peripheral LSIs. For example, data can be directly transferred between memory and these peripheral LSIs. This state is cleared if the $\overline{BUSREQ}$ signal is found "1" after sampling it on the rising edge of each subsequent clock state (TX), and enters into the next machine cycle. During the floating state, the $\overline{BUSACK}$ signal goes "0" to indicate it to the peripheral LSIs.

In this state, however, no memory refresh is performed and, therefore, the $\overline{RFSH}$ signal is set to "1". Hence, to maintain this state for a long time with a system using dynamic memory, memory refresh must be performed by the external controller.

Note that, in the floating state, neither maskable interrupt (INT) nor non-maskable interrupt (NMI) can be accepted.

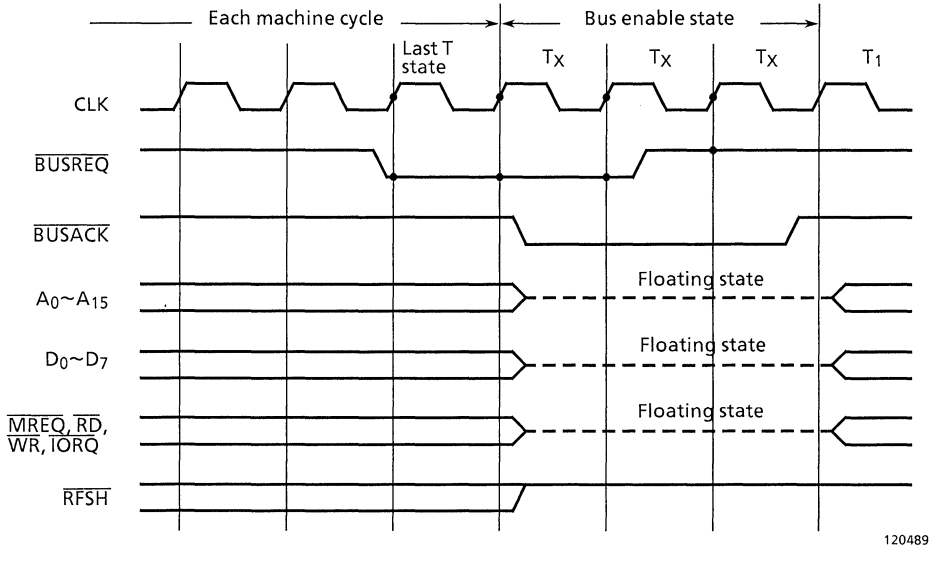


Figure 3.2.16 Bus Request and Bus Acknowledge Timing

(5) Maskable interrupt acknowledge operation

Figure 3.2.17 shows the basic timing of the maskable interrupt acknowledge.

The MPU samples the maskable interrupt request signal ($\overline{INT}$) on the rising edge of the last clock of each instruction execution. If the $\overline{INT}$ signal is found "0", a maskable interrupt is accepted except in the following cases:

- The interrupt enable flip-flop is reset to "0".
- The $\overline{BUSREQ}$ signal is "0".

When a maskable interrupt has been accepted, a special Opcode fetch cycle is generated. In this cycle, 2 clock states of wait state (TW*) is automatically inserted after the clock state T2. The $\overline{WAIT}$ signal is sampled on the falling edges of the second clock state TW* and the following clock state TW and, if the $\overline{WAIT}$ signal is found "0", the instruction cycle enters in the next clock state TW. In this Opcode fetch cycle, the $\overline{IORQ}$ signal goes "0" in the first TW* state instead of the $\overline{MREQ}$ signal while, in a normal Opcode fetch cycle, the $\overline{MREQ}$ signal goes "0" in clock state T1. This indicates to the maskable interrupt requesting LSI that the 8-bit interrupt vector can be put on the data bus. The MPU reads this data to perform interrupt processing. Therefore, the contents of the program counter put on the address bus are not used. Unlike an ordinary I/O operation, the $\overline{RD}$ signal does not go "0".

In clock state T3, the memory refresh address signal is put on the address bus for memory refresh like normal Opcode fetch cycle and the $\overline{RFSH}$ signal goes "0". In the subsequent machine cycles (M2 and M3) , the contents of the current program counter are saved into the stack. In machine cycles M4 and M5, the contents of the I register (the high-order 8 bits) and the contents of the address indicated by the address of the vector (the low-order 8 bits) from the CTC and the peripheral LSI are fetched in the program counter.

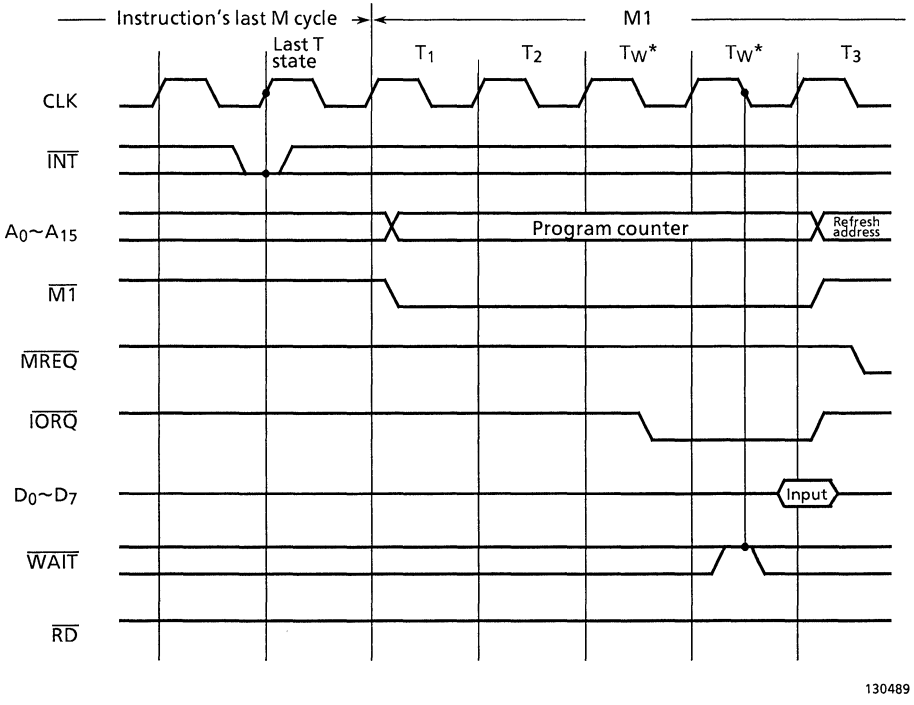


Figure 3.2.17 Maskable Interrupt Acknowledge Timing

(6) Non-maskable interrupt acknowledge operation

Figure 3.2.18 shows the basic timing of non-maskable interrupt acknowledge.

When the non-maskable interrupt request signal ($\overline{NMI}$) goes low, the internal non-maskable flip-flop is set to "1". The $\overline{NMI}$ signal is detected in any timing of each instruction. However, the internal NMI flip-flop is sampled on the rising edge of the last clock of each instruction. Therefore, the $\overline{NMI}$ signal should go low by the last clock state of an instruction.

The Opcode fetch cycle for non-maskable interrupt request acknowledge is generally the same as the ordinary Opcode fetch cycle. However, the Opcode on the data bus at the time is ignored. The contents of the current program counter are saved into the stack in the subsequent machine cycles (M2 and M3) . In the following machine cycle, the operation jumps to address 0066H, the non-maskable interrupt vector address. The machine cycles after these depend on the contents of the fetched Opcode.

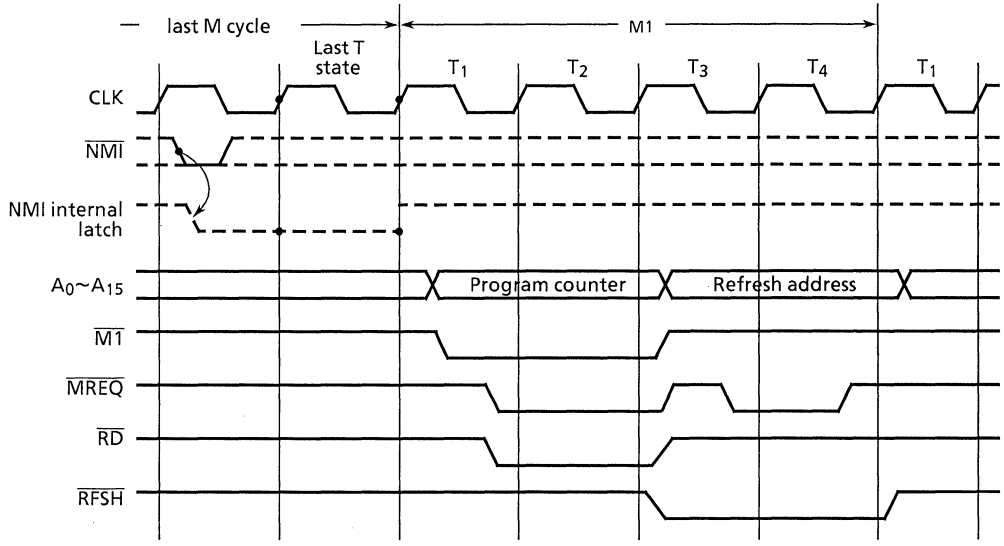


Figure 3.2.18 Non-Maaskable Inpterrupt Acknowledge Timing

(7) Halt operation

When a HALT instruction is fetched in the Opcode fetch cycle, the MPU sets the $\overline{\text{HALT}}$ signal to "0" synchronized with the falling edge of clock state T4 to indicate it to the peripheral LSI and stops operating. If the system clock is kept supplied in the halt state, the MPU continues executing NOP instructions. This is done to output refresh signals when the dynamic memory is used. The NOP instruction execution cycle is the same as the ordinary Opcode fetch cycle except the data on the data bus are ignored.

The halt state is cleared when an interrupt is accepted or the $\overline{\text{RESET}}$ signal is set to "0" to reset the MPU. Figure 3.2.19 shows the halt state clear operation by interrupt acknowledge. An interrupt is sampled on the rising edge of the last clock (clock state T4) of the NOP instruction. A maskable interrupt can be accepted when the $\overline{\text{INT}}$ signal is "0". A non-maskable interrupt is accepted when the internal NMI flip-flop which is set on the falling edge of the $\overline{\text{NMI}}$ signal is set at "1". However, it is required that the interrupt enable flip-flop is set to "1" for a maskable interrupt to be accepted. The interrupt processing for the accepted interrupt starts from the next cycle.

However, when the supply of the system clock from the CGC has been stopped by the power down operation, it is required to restart the supply of the system clock and input the $\overline{INT}$ signal until the execution of one instruction is completed or the $\overline{RESET}$ signal until 3 clocks are output. Figure 3.2.20 shows the timing of clearing the halt state caused by power down.

For the reset operation, see (8) "Reset operation". Note that the $\overline{INT}$ and $\overline{NMI}$ signals are shown on the same diagram in Figures 3.2.19 and 3.2.20 for convenience.

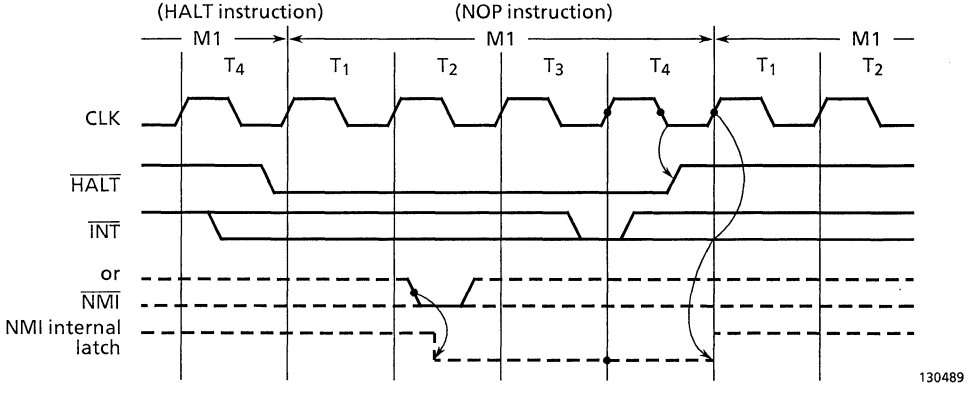


Figure 3.2.19 Timing of Clearing Halt State Caused by Interrupt Acknowledge

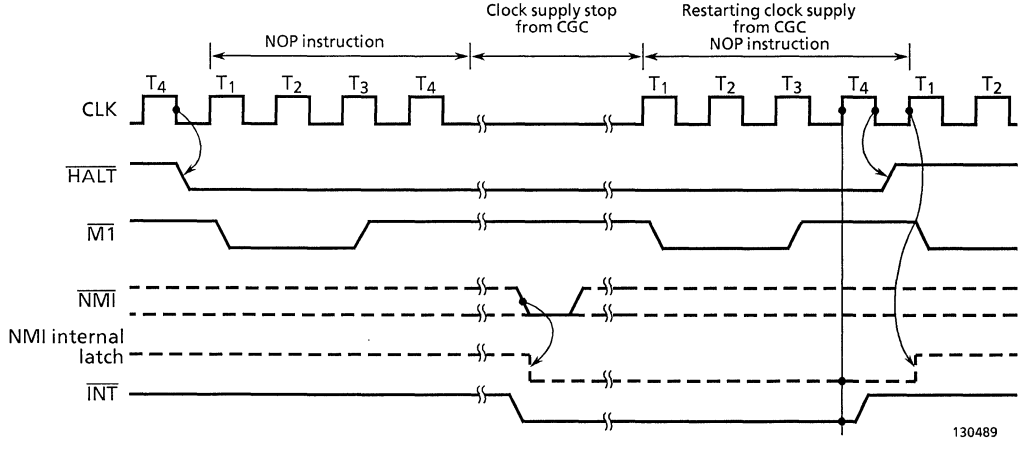


Figure 3.2.20 Timing of Clearing Halt State Caused by Power Down

(8) Reset operation

Figure 3.2.21 shows the basic timing of reset operation.

To reset the MPU, the $\overline{\text{RESET}}$ signal must be kept at "0" for at least 3 clocks. When the $\overline{\text{RESET}}$ signal goes "1", instruction execution starts from address 0000H after a dummy cycle of at least 2 clock states.

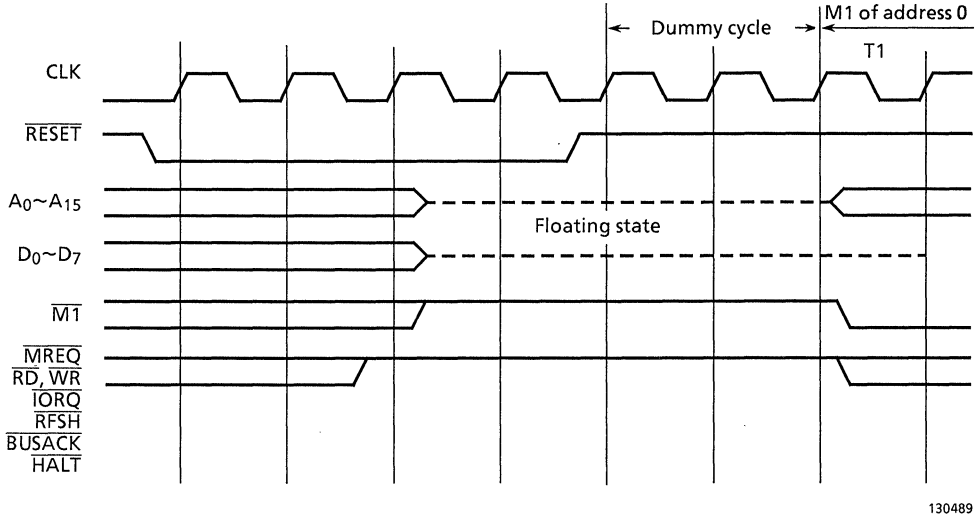


Figure 3.2.21 Reset Timing

To clear the power down state by using the $\overline{\text{RESET}}$ signal, the $\overline{\text{RESET}}$ signal must be input until 3 clocks or more are supplied by restarting the supply of the system clock from the CGC.

(9) Evaluation operation

Each of the MPU signals (A0 through A15, D0 through D7, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$, $\overline{HALT}$, $\overline{M1}$, and $\overline{RFSH}$) can be put in the high-impedance state by EV and $\overline{BUSREQ}$ signals to electrically disconnect them from the MPU.

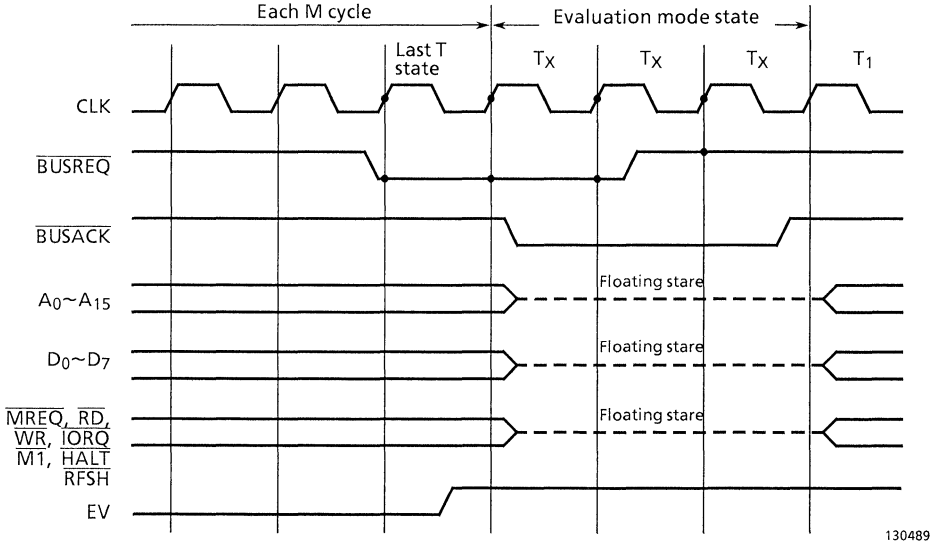


Figure 3.2.22 Evaluation Timing

Figure 3.2.23 shows the block diagram of the TMPZ84C013A operating as an evaluator in the evaluation mode.

The operations controlled by signals from the external MPU in the evaluation mode are the same as those of each device constituting the TMPZ84C013A. (However, for the watchdog timer operations, see "WDT Operational Description" because the watchdog timer is of random logic configuration.)

For the electrical characteristics and timing of each device, see "Inactive State".

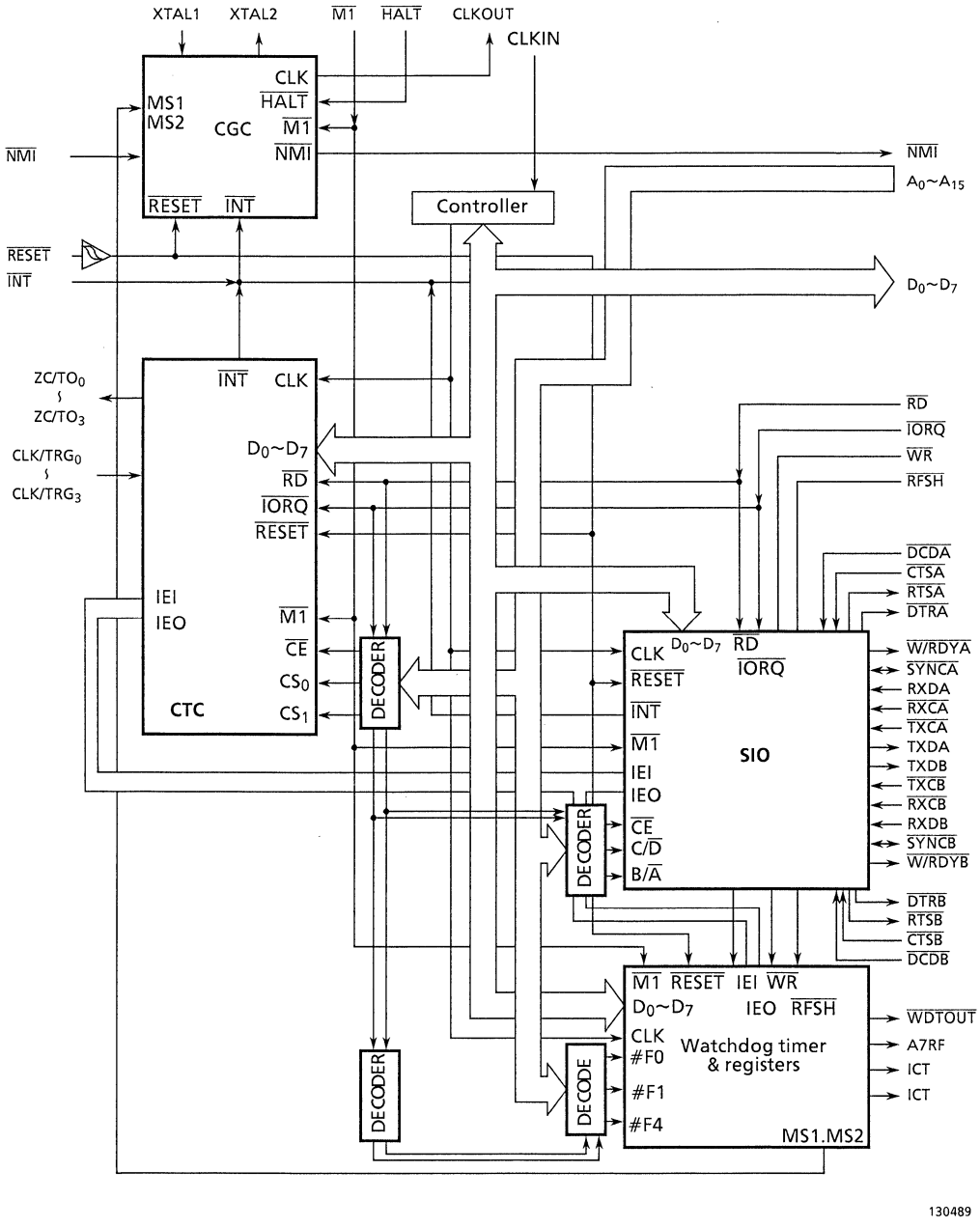


Figure 3.2.23 Block Diagram of the TMPZ84C013A Functioning As Evaluator

3.2.4 TMPZ84C013A Instruction Set

This subsection lists the TMPZ84C013A instruction codes and their functions. The table below lists the symbols and abbreviations used to describe the instruction set. The symbols which require special attention are described in the locations in which they appear.

- Symbols (1/2)

Classification	Symbol	Meaning
Register	r, g	Register B, C, D, E, H, L, A,
	t	Register pair BC, DE, HL
		Stack pointer SP
	q	Register pair BC, DE, HL, AF
	p	Register pair BC, DE
		Index register IX
		Stack pointer SP
	s	Register pair BC, DE
		Index register IY
		Stack pointer SP
	t_H	Higher register of register pair (B, D, H)
		Higher 8 bits of stack pointer (SP)
	q_H	Higher register of register pair (B, D, H, A)
	IX_H	Higher 8 bits of index register IX
	IY_H	Higher 8 bits of index register IY
	PC_H	Higher 8 bits of program counter (PC)
	t_L	Lower register of register pair (C, E, L)
		Lower 8 bits of stack pointer (SP)
	q_L	Lower register of register pair (C, E, L, F)
	IX_L	Lower 8 bits of index register IX
	IY_L	Lower 8 bits of index register IY
	PC_L	Lower 8 bits of program counter (PC)
	rb	Bit b (0-7) of register (B, C, D, E, H, L, A)

120489

- Symbols (2/2)

Classification	Symbol	Meaning
Memory	mn	Memory address represented in 16 bits. m indicates higher 8 bits and n, lower 8 bits.
	$(HL)_b$	Bit b (0-7) of the contents of the memory address indicated by register pair HL.
	$(IX+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IX.
	$(IY+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IY.
Flag change symbol	0	Reset to "0" by operation.
	1	Set to "1" by operation.
	-	No change
	*	Affected by operation
	X	Undefined
	P	Handled as parity flag. P = 0: odd parity P = 1: even parity
	V	Handled as overflow flag. V = 0: No overflow V = 1: Overflow
Operator	←	Transfer
	↔	Exchange
	+	Add
	−	Subtract
	∧	Logical and between bits.
	∨	Logical or between bits.
	⊕	Exclusive or between bits
Others	IFF	Interrupt enable flip-flop
	CY	Carry flag
	Z	Zero flag

120489

TMPZ84C013A Instruction Set (1/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT DATA LOAD	LD r,g	01 rrr ggg	40+r×8+g	r←g	–	–	X	–	X	–	–	–	1	4
	LD r,n	00 rrr 110 nn nnn nnn	06+r×8 n	r←n	–	–	X	–	X	–	–	–	2	7
	LD r,(HL)	01 rrr 110	46+r×8	r←(HL)	–	–	X	–	X	–	–	–	2	7
	LD r,(IX+d)	11 011 101 01 rrr 110 dd ddd ddd	DD 46+r×8 d	r←(IX+d)	–	–	X	–	X	–	–	–	5	19
	LD r,(IY+d)	11 111 101 01 rrr 110 dd ddd ddd	FD 46+r×8 d	r←(IY+d)	–	–	X	–	X	–	–	–	5	19
	LD (HL),r	01 110 rrr	70+r	(HL)←r	–	–	X	–	X	–	–	–	2	7
	LD (IX+d),r	11 011 101 01 110 rrr dd ddd ddd	DD 70+r d	(IX+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),r	11 111 101 01 110 rrr dd ddd ddd	FD 70+r d	(IY+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (HL),n	00 110 110 nn nnn nnn	36 n	(HL)←n	–	–	X	–	X	–	–	–	3	10
	LD (IX+d), n	11 011 101 00 110 110 dd ddd ddd nn nnn nnn	DD 36 d n	(IX+d)←n	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),n	11 111 101 00 110 110 dd ddd ddd nn nnn nnn	FD 36 d n	(IY+d)←n	–	–	X	–	X	–	–	–	5	19
	LD A,(BC)	00 001 010	0A	A←(BC)	–	–	X	–	X	–	–	–	2	7
	LD A,(DE)	00 011 010	1A	A←(DE)	–	–	X	–	X	–	–	–	2	7
	LD A,(mn)	00 111 010 nn nnn nnn mm mmm mmm	3A n m	A←(mn)	–	–	X	–	X	–	–	–	4	13
	LD (BC),A	00 000 010	02	(BC)←A	–	–	X	–	X	–	–	–	2	7
	LD (DE),A	00 010 010	12	(DE)←A	–	–	X	–	X	–	–	–	2	7
	LD (mn),A	00 110 010 nn nnn nnn mm mmm mmm	32 n m	(mn)←A	–	–	X	–	X	–	–	–	4	13
	LD A,I	11 101 101 01 010 111	ED 57	A←I	*	*	X	0	X	IFF	0	–	2	9
	LD A,R	11 101 101 01 011 111	ED 5F	A←R	*	*	X	0	X	IFF	0	–	2	9
	LD I,A	11 101 101 01 000 111	ED 47	I←A	–	–	X	–	X	–	–	–	2	9
	LD R,A	11 101 101 01 001 111	ED 4F	R←A	–	–	X	–	X	–	–	–	2	9
16-BIT DATA LOAD	LD t,mn	00 tt0 001 nn nnn nnn mm mmm mmm	01+t×10 n m	t←mn	–	–	X	–	X	–	–	–	3	10
	LD IX,mn	11 011 101 00 100 001 nn nnn nnn mm mmm mmm	DD 21 n m	IX←mn	–	–	X	–	X	–	–	–	4	14

r	rrr
g	ggg
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

Note : r,g means any of the registers A, B, C, D, E, H, L.
IFF in "Flag" column indicates that the content of the interrupt enable flip-flop is copied into the P/V flag.

120489

TMPZ84C013A Instruction Set (2/9)

ITEM/CLASSI-FICATION	Assembler mnemonic	Object code Binary	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY-CLES	No. OF STA-TES
		76 543 210												
16-BIT DATA LOAD	LD IY,mn	11 111 101 00 100 001 nn nnn nnn mm mmm mmm	FD 21 n m	IY←mn	–	–	X	–	X	–	–	–	4	14
	LD HL,(mn)	00 101 010 nn nnn nnn mm mmm mmm	2A n m	H←(mn+1) L←(mn)	–	–	X	–	X	–	–	–	5	16
	LD t,(mn)	11 101 101 01 tt1 011 nn nnn nnn mm mmm mmm	ED 4B+t×10 n m	t_H←(mn+1) t_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IX,(mn)	11 011 101 00 101 010 nn nnn nnn mm mmm mmm	DD 2A n m	IX_H←(mn+1) IX_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IY,(mn)	11 111 101 00 101 010 nn nnn nnn mm mmm mmm	FD 2A n m	IY_H←(mn+1) IY_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD (mn),HL	00 100 010 nn nnn nnn mm mmm mmm	22 n m	(mn+1)←H (mn)←L	–	–	X	–	X	–	–	–	5	16
	LD (mn),t	11 101 101 01 tt0 011 nn nnn nnn mm mmm mmm	ED 43+t×10 n m	(mn+1)←t_H (mn)←t_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IX	11 011 101 00 100 010 nn nnn nnn mm mmm mmm	DD 22 n m	(mn+1)←IX_H (mn)←IX_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IY	11 111 101 00 100 010 nn nnn nnn mm mmm mmm	FD 22 n m	(mn+1)←IY_H (mn)←IY_L	–	–	X	–	X	–	–	–	6	20
	LD SP,HL	11 111 001	F9	SP←HL	–	–	X	–	X	–	–	–	1	6
	LD SP,IX	11 011 101 11 111 001	DD F9	SP←IX	–	–	X	–	X	–	–	–	2	10
	LD SP,IY	11 111 101 11 111 001	FD F9	SP←IY	–	–	X	–	X	–	–	–	2	10
	PUSH q	11 qq0 101	C5+q×10	(SP−2)←q_L,(SP−1)←q_H, SP←SP−2	–	–	X	–	X	–	–	–	3	11
	PUSH IX	11 011 101 11 100 101	DD E5	(SP−2)←IX_L,(SP−1)←IX_H SP←SP−2	–	–	X	–	X	–	–	–	4	15
	PUSH IY	11 111 101 11 100 101	FD E5	(SP−2)←IY_L,(SP−1)←IY_H SP←SP−2	–	–	X	–	X	–	–	–	4	15
	POP q	11 qq0 001	C1+q×10	q_H←(SP+1),q_L←(SP), SP←SP+2	–	–	X	–	X	–	–	–	3	10
	POP IX	11 011 101 11 100 001	DD E1	IX_H←(SP+1),IX_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
	POP IY	11 111 101 11 100 001	FD E1	IY_H←(SP+1),IY_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
*1	EX DE,HL	11 101 011	EB	DE↔HL	–	–	X	–	X	–	–	–	1	4
	EX AF,AF'	00 001 000	08	AF↔AF'	–	–	X	–	X	–	–	–	1	4
	EXX	11 011 001	D9	BC↔BC',DE↔DE',HL↔HL'	–	–	X	–	X	–	–	–	1	4

t	tt
BC	00
DE	01
HL	10
SP	11

q	qq
BC	00
DE	01
HL	10
AF	11

Note : t is any of the register pairs BC, DE, HL, SP.
q is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively. (Ex) BC_L = C, AF_H = A.

*1 : EXCHANGE

120489

TMPZ84C013A Instruction Set (3/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES	
EXCHANGE	EX (SP),HL	11 100 011	E3	H↔(SP+1),L↔(SP)	–	–	X	–	X	–	–	–	5	19	
	EX (SP),IX	11 011 101 11 100 011	DD E3	IX_H↔(SP+1) IX_L↔(SP)	–	–	X	–	X	–	–	–	6	23	
	EX (SP),IY	11 111 101 11 100 011	FD E3	IY_H↔(SP+1) IY_L↔(SP)	–	–	X	–	X	–	–	–	6	23	
BLOCK TRANSFER BLOCK SEARCH	LDI	11 101 101 10 100 000	ED A0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDIR	11 101 101 10 110 000	ED B0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	LDD	11 101 101 10 101 000	ED A8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDDR	11 101 101 10 111 000	ED B8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	CPI	11 101 101 10 100 001	ED A1	A−(HL) HL←HL+1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPIR	11 101 101 10 110 001	ED B1	A−(HL),HL←HL+1,BC←BC−1 Repeat until A=(HL) or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
	CPD	11 101 101 10 101 001	ED A9	A−(HL) HL←HL−1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPDR	11 101 101 10 111 001	ED B9	A−(HL),HL←HL−1,BC←BC−1 Repeat until A=(HL)or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
8-BIT ARITHMETIC AND LOGICAL	ADD A,r	10 000 rrr	80+r	A←A+r	*	*	X	*	X	V	0	*	1	4	
	ADD A,n	11 000 110 nn nnn nnn	C6 n	A←A+n	*	*	X	*	X	V	0	*	2	7	
	ADD A,(HL)	10 000 110	86	A←A+(HL)	*	*	X	*	X	V	0	*	2	7	
	ADD A,(IX+d)	11 011 101 10 000 110 dd ddd ddd	DD 86 d	A←A+(IX+d)	*	*	X	*	X	V	0	*	5	19	
	ADD A,(IY+d)	11 111 101 10 000 110 dd ddd ddd	FD 86 d	A←A+(IY+d)	*	*	X	*	X	V	0	*	5	19	
	ADC A,r	10 001 rrr	88+r	A←A+r+CY	*	*	X	*	X	V	0	*	1	4	
	ADC A,n	11 001 110 nn nnn nnn	CE n	A←A+n+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(HL)	10 001 110	8E	A←A+(HL)+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(IX+d)	11 011 101 10 001 110 dd ddd ddd	DD 8E d	A←A+(IX+d)+CY	*	*	X	*	X	V	0	*	5	19	
	ADC A,(IY+d)	11 111 101 10 001 110 dd ddd ddd	FD 8E d	A←A+(IY+d)+CY	*	*	X	*	X	V	0	*	5	19	
	SUB r	10 010 rrr	90+r	A←A−r	*	*	X	*	X	V	1	*	1	4	
	SUB n	11 010 110 nn nnn nnn	D6 n	A←A−n	*	*	X	*	X	V	1	*	2	7	
	SUB (HL)	10 010 110	96	A←A−(HL)	*	*	X	*	X	V	1	*	2	7	
	SUB (IX+d)	11 011 101 10 010 110 dd ddd ddd	DD 96 d	A←A−(IX+d)	*	*	X	*	X	V	1	*	5	19	
	SUB (IY+d)	11 111 101 10 010 110 dd ddd ddd	FD 96 d	A←A−(IY+d)	*	*	X	*	X	V	1	*	5	19	

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : *M P/V flag is 0 if the result of BC − 1 = 0, otherwise P/V = 1.
*N Z flag is 1 if A = (HL), otherwise Z = 0.
[ ]indicates the total condition of the number of cycles and states indicated by arrow.
r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C013A Instruction Set (4/9)

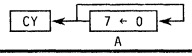
ITEM/CLASSI-FICATION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY-CLES	No. OF STA-TES
8-BIT ARITHMETIC AND LOGICAL	SBC A,r	10 011 rrr	98+r	A←A−r−CY	*	*	X	*	X	V	1	*	1	4
	SBC A,n	11 011 110 nn nnn nnn	DE n	A←A−n−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(HL)	10 011 110	9E	A←A−(HL)−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(IX+d)	11 011 101 10 011 110 dd ddd ddd	DD 9E d	A←A−(IX+d)−CY	*	*	X	*	X	V	1	*	5	19
	SBC A,(IY+d)	11 111 101 10 011 110 dd ddd ddd	FD 9E d	A←A−(IY+d)−CY	*	*	X	*	X	V	1	*	5	19
	AND r	10 100 rrr	A0+r	A←A∧r	*	*	X	1	X	P	0	0	1	4
	AND n	11 100 110 nn nnn nnn	E6 n	A←A∧n	*	*	X	1	X	P	0	0	2	7
	AND (HL)	10 100 110	A6	A←A∧(HL)	*	*	X	1	X	P	0	0	2	7
	AND (IX+d)	11 011 101 10 100 110 dd ddd ddd	DD A6 d	A←A∧(IX+d)	*	*	X	1	X	P	0	0	5	19
	AND (IY+d)	11 111 101 10 100 110 dd ddd ddd	FD A6 d	A←A∧(IY+d)	*	*	X	1	X	P	0	0	5	19
	OR r	10 110 rrr	B0+r	A←A∨r	*	*	X	0	X	P	0	0	1	4
	OR n	11 110 110 nn nnn nnn	F6 n	A←A∨n	*	*	X	0	X	P	0	0	2	7
	OR (HL)	10 110 110	B6	A←A∨(HL)	*	*	X	0	X	P	0	0	2	7
	OR (IX+d)	11 011 101 10 110 110 dd ddd ddd	DD B6 d	A←A∨(IX+d)	*	*	X	0	X	P	0	0	5	19
	OR (IY+d)	11 111 101 10 110 110 dd ddd ddd	FD B6 d	A←A∨(IY+d)	*	*	X	0	X	P	0	0	5	19
	XOR r	10 101 rrr	A8+r	A←A∀r	*	*	X	0	X	P	0	0	1	4
	XOR n	11 101 110 nn nnn nnn	EE n	A←A∀n	*	*	X	0	X	P	0	0	2	7
	XOR (HL)	10 101 110	AE	A←A∀(HL)	*	*	X	0	X	P	0	0	2	7
	XOR (IX+d)	11 011 101 10 101 110 dd ddd ddd	DD AE d	A←A∀(IX+d)	*	*	X	0	X	P	0	0	5	19
	XOR (IY+d)	11 111 101 10 101 110 dd ddd ddd	FD AE d	A←A∀(IY+d)	*	*	X	0	X	P	0	0	5	19
	CP r	10 111 rrr	B8+r	A−r	*	*	X	*	X	V	1	*	1	4
	CP n	11 111 110 nn nnn nnn	FE n	A−n	*	*	X	*	X	V	1	*	2	7
	CP (HL)	10 111 110	BE	A−(HL)	*	*	X	*	X	V	1	*	2	7
	CP (IX+d)	11 011 101 10 111 110 dd ddd ddd	DD BE d	A−(IX+d)	*	*	X	*	X	V	1	*	5	19
	CP (IY+d)	11 111 101 10 111 110 dd ddd ddd	FD BE d	A−(IY+d)	*	*	X	*	X	V	1	*	5	19
	INC r	00 rrr 100	04+r×8	r←r+1	*	*	X	*	X	V	0	−	1	4
	INC (HL)	00 110 100	34	(HL)←(HL)+1	*	*	X	*	X	V	0	−	3	11
	INC (IX+d)	11 011 101 00 110 100 dd ddd ddd	DD 34 d	(IX+d)←(IX+d)+1	*	*	X	*	X	V	0	−	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C013A Instruction Set (5/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT ARITHMETIC AND LOGICAL	INC (IY+d)	11 111 101 00 110 100 dd ddd ddd	FD 34 d	(IY+d)←(IY+d)+1	*	*	X	*	X	V	0	–	6	23
	DEC r	00 rrr 101	05+r×8	r←r−1	*	*	X	*	X	V	1	–	1	4
	DEC (HL)	00 110 101	35	(HL)←(HL)−1	*	*	X	*	X	V	1	–	3	11
	DEC (IX+d)	11 011 101 00 110 101 dd ddd ddd	DD 35 d	(IX+d)←(IX+d)−1	*	*	X	*	X	V	1	–	6	23
	DEC (IY+d)	11 111 101 00 110 101 dd ddd ddd	FD 35 d	(IY+d)←(IY+d)−1	*	*	X	*	X	V	1	–	6	23
GENELAL-PURPOSE ARITHMETIC AND MPU CONTROL	DAA	00 100 111	27	Decimal adjust accumulator	*	*	X	*	X	P	–	*	1	4
	CPL	00 101 111	2F	A←$\overline{A}$	–	–	X	1	X	–	1	–	1	4
	NEG	11 101 101 01 000 100	ED 44	A←0−A	*	*	X	*	X	V	1	*	2	8
	CCF	00 111 111	3F	CY←$\overline{CY}$	–	–	X	X	X	–	0	*	1	4
	SCF	00 110 111	37	CY←1	–	–	X	0	X	–	0	1	1	4
	NOP	00 000 000	00	no operation	–	–	X	–	X	–	–	–	1	4
	HALT	01 110 110	76	MPU Halted	–	–	X	–	X	–	–	–	1	4
	DI	11 110 011	F3	IFF←0	–	–	X	–	X	–	–	–	1	4
	EI	11 111 011	FB	IFF←1	–	–	X	–	X	–	–	–	1	4
	IM 0	11 101 101 01 000 110	ED 46	Set interrupt mode 0	–	–	X	–	X	–	–	–	2	8
	IM 1	11 101 101 01 010 110	ED 56	Set interrupt mode 1	–	–	X	–	X	–	–	–	2	8
	IM 2	11 101 101 01 011 110	ED 5E	Set interrupt mode 2	–	–	X	–	X	–	–	–	2	8
16-BIT ARITHMETIC	ADD HL,t	00 tt1 001	09+t×10	HL←HL+t	–	–	X	X	X	–	0	*	3	11
	ADC HL,t	11 101 101 01 tt1 010	ED 4A+t×10	HL←HL+t+CY	*	*	X	X	X	V	0	*	4	15
	SBC HL,t	11 101 101 01 tt0 010	ED 42+t×10	HL←HL−t−CY	*	*	X	X	X	V	1	*	4	15
	ADD IX,p	11 011 101 00 pp1 001	DD 09+p×10	IX←IX+p	–	–	X	X	X	–	0	*	4	15
	ADD IY,s	11 111 101 00 ss1 001	FD 09+s×10	IY←IY+s	–	–	X	X	X	–	0	*	4	15
	INC t	00 tt0 011	03+t×10	t←t+1	–	–	X	–	X	–	–	–	1	6
	INC IX	11 011 101 00 100 011	DD 23	IX←IX+1	–	–	X	–	X	–	–	–	2	10
	INC IY	11 111 101 00 100 011	FD 23	IY←IY+1	–	–	X	–	X	–	–	–	2	10
	DEC t	00 tt1 011	0B+t×10	t←t−1	–	–	X	–	X	–	–	–	1	6
	DEC IX	11 011 101 00 101 011	DD 2B	IX←IX−1	–	–	X	–	X	–	–	–	2	10
	DEC IY	11 111 101 00 101 011	FD 2B	IY←IY−1	–	–	X	–	X	–	–	–	2	10
ROTATE	RLCA	00 000 111	07		–	–	X	0	X	–	0	*	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

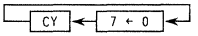
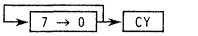
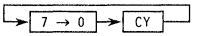
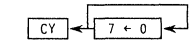
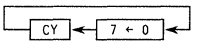
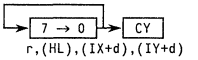
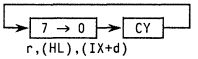
p	pp
BC	00
DE	01
IX	10
SP	11

s	ss
BC	00
DE	01
IY	10
SP	11

Note : ss is any of the register pairs BC, DE, HL, SP. PP is any of the register pairs BC, DE, IX, SP.
rr is any of the register pairs BC, DE, IY, SP.

120489

TMPZ84C013A Instruction Set (6/9)

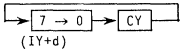
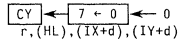
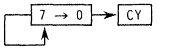
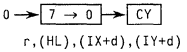
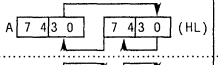
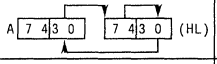
ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RLA	00 010 111	17		–	–	X	0	X	–	0	*	1	4
	RRCA	00 001 111	0F		–	–	X	0	X	–	0	*	1	4
	RRA	00 011 111	1F		–	–	X	0	X	–	0	*	1	4
	RLC r	11 001 011 00 000 rrr	CB 00+r		*	*	X	0	X	P	0	*	2	8
	RLC (HL)	11 001 011 00 000 110	CB 06		*	*	X	0	X	P	0	*	4	15
	RLC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 000 110	DD CB d 06		*	*	X	0	X	P	0	*	6	23
	RLC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 000 110	FD CB d 06		*	*	X	0	X	P	0	*	6	23
	RL r	11 001 011 00 010 rrr	CB 10+r		*	*	X	0	X	P	0	*	2	8
	RL (HL)	11 001 011 00 010 110	CB 16		*	*	X	0	X	P	0	*	4	15
	RL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 010 110	DD CB d 16		*	*	X	0	X	P	0	*	6	23
	RL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 010 110	FD CB d 16		*	*	X	0	X	P	0	*	6	23
	RRC r	11 001 011 00 001 rrr	CB 08+r		*	*	X	0	X	P	0	*	2	8
	RRC (HL)	11 001 011 00 001 110	CB 0E		*	*	X	0	X	P	0	*	4	15
	RRC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 001 110	DD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RRC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 001 110	FD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RR r	11 001 011 00 011 rrr	CB 18+r		*	*	X	0	X	P	0	*	2	8
	RR (HL)	11 001 011 00 011 110	CB 1E		*	*	X	0	X	P	0	*	4	15
	RR (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 011 110	DD CB d 1E		*	*	X	0	X	P	0	*	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C013A Instruction Set (7/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RR (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 011 110	FD CB d 1E	 (IY+d)	*	*	X	0	X	P	0	*	6	23
	SLA r	11 001 011 00 100 rrr	CB 20+r		*	*	X	0	X	P	0	*	2	8
	SLA (HL)	11 001 011 00 100 110	CB 26		*	*	X	0	X	P	0	*	4	15
	SLA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 100 110	DD CB d 26	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	SLA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 100 110	FD CB d 26		*	*	X	0	X	P	0	*	6	23
	SRA r	11 001 011 00 101 rrr	CB 28+r		*	*	X	0	X	P	0	*	2	8
	SRA (HL)	11 001 011 00 101 110	CB 2E		*	*	X	0	X	P	0	*	4	15
	SRA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 101 110	DD CB d 2E	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	SRA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 101 110	FD CB d 2E		*	*	X	0	X	P	0	*	6	23
	SRL r	11 001 011 00 111 rrr	CB 38+r		*	*	X	0	X	P	0	*	2	8
	SRL (HL)	11 001 011 00 111 110	CB 3E		*	*	X	0	X	P	0	*	4	15
	SRL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 111 110	DD CB d 3E	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23
	SRL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 111 110	FD CB d 3E		*	*	X	0	X	P	0	*	6	23
	RLD	11 101 101 01 101 111	ED 6F		*	*	X	0	X	P	0	–	5	18
	RRD	11 101 101 01 100 111	ED 67		*	*	X	0	X	P	0	–	5	18
BIT SET RESET AND TEST	BIT b,r	11 001 011 01 bbb rrr	CB 40+b×8+r	$Z \leftarrow \overline{r_b}$	X	*	X	1	X	X	0	–	2	8
	BIT b,(HL)	11 001 011 01 bbb 110	CB 46+b×8	$Z \leftarrow \overline{(HL)_b}$	X	*	X	1	X	X	0	–	3	12

*1

*1

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Note : *1 : Rotate digit left and right between the accmulator and location (HL).
The content of the upper half of the accumulator is unaffected.
The notation $(HL)_b$ indicates bit $_b$ (0 to 7) within the contents of the HL register pair.
The notation r_b indicates bit $_b$ (0 to 7) within the r register.

120489

TMPZ84C013A Instruction Set (8/9)

ITEM/CLASSIFICATION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CYCLES	No. OF STATES
BIT SET RESET AND TEST	BIT b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 01 bbb 110	DD CB d 46+b×8	$Z \leftarrow \overline{(IX+d)_b}$	X	*	X	1	X	X	0	–	5	20
	BIT b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 01 bbb 110	FD CB d 46+b×8	$Z \leftarrow \overline{(IY+d)_b}$	X	*	X	1	X	X	0	–	5	20
	SET b,r	11 001 011 11 bbb rrr	CB C0+b×8+r	$r_b \leftarrow 1$	–	–	X	–	X	–	–	–	2	8
	SET b,(HL)	11 001 011 11 bbb 110	CB C6+b×8	$(HL)_b \leftarrow 1$	–	–	X	–	X	–	–	–	4	15
	SET b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 11 bbb 110	DD CB d C6+b×8	$(IX+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
	SET b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 11 bbb 110	FD CB d C6+b×8	$(IY+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
	RES b,r	11 001 011 10 bbb rrr	CB 80+b×8+r	$r_b \leftarrow 0$	–	–	X	–	X	–	–	–	2	8
	RES b,(HL)	11 001 011 10 bbb 110	CB 86+b×8	$(HL)_b \leftarrow 0$	–	–	X	–	X	–	–	–	4	15
	RES b,(IX+d)	11 011 101 11 001 011 dd ddd ddd 10 bbb 110	DD CB d 86+b×8	$(IX+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
	RES b,(IY+d)	11 111 101 11 001 011 dd ddd ddd 10 bbb 110	FD CB d 86+b×8	$(IY+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
JUMP	JP mn	11 000 011 nn nnn nnn mm mmm mmm	C3 n m	PC←mn	–	–	X	–	X	–	–	–	3	10
	JP c,mn	11 ccc 010 nn nnn nnn mm mmm mmm	C2+c×8 n m	PC←mn (Only when condition is met)	–	–	X	–	X	–	–	–	3	10
	JR $+e	00 011 000 aa aaa aaa	18 a	PC←$+e	–	–	X	–	X	–	–	–	3	12
	JR C,$+e	00 111 000 aa aaa aaa	38 a	If C=0, continue If C=1, PC←$+e	–	–	X	–	X	–	–	–	2 3	7 12
	JR NC,$+e	00 110 000 aa aaa aaa	30 a	If C=0, PC←$+e If C=1, continue	–	–	X	–	X	–	–	–	3 2	12 7
	JR Z,$+e	00 101 000 aa aaa aaa	28 a	If Z=0, continue If Z=1, PC←$+e	–	–	X	–	X	–	–	–	2 3	7 12
	JR NZ,$+e	00 100 000 aa aaa aaa	20 a	If Z=0, PC←$+e If Z=1, continue	–	–	X	–	X	–	–	–	3 2	12 7
	DJNZ $+e	00 010 000 aa aaa aaa	10 a	B←B−1, If B=0, continue B←B−1, If B<>0, continue	–	–	X	–	X	–	–	–	2 3	8 13
	JP (HL)	11 101 001	E9	PC←HL	–	–	X	–	X	–	–	–	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

e represents the extension in the relative addressing mode, a = e − 2. e is a signed two's complement number in the range of $-126 \leq e \leq 129$

Note :
- a = e − 2 in the Opcode provides an effective address of PC + e as PC is incremented by 2 before the addition of e.
- $ indicates the reference to the location counter value of the current segment.
- The notation $(HL)_b$, $(IX + d)_b$ indicates bit $_b$ (0 to 7) within the contents of the register pair.
- The notation r_b indicates bit $_b$ (0 to 7) within the r register.
- a = e-2 in the op-code provides effective address of PC + e as PC is incremented by 2 prior to the addition of e.

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign Negative

270489

TMPZ84C013A Instruction Set (9/9)

ITEM/ CLASSIFICATION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CYCLES	No. OF STATES
JUMP	JP (IX)	11 011 101 11 101 001	DD E9	PC←(IX)	–	–	X	–	X	–	–	–	2	8
	JP (IY)	11 111 101 11 101 001	FD E9	PC←(IY)	–	–	X	–	X	–	–	–	2	8
CALL AND RETURN	CALL mn	11 001 101 nn nnn nnn mm mmm mmm	CD n m	$(SP-1)\leftarrow PC_H$, $(SP-2)\leftarrow PC_L$ PC←mn SP←SP−2	–	–	X	–	X	–	–	–	5	17
	CALL c,mn	11 ccc 100 nn nnn nnn mm mmm mmm	C4+cX8 n m	If condition c is met, same as CALL mn. If condition c is not met,continue	– –	– –	X X	– –	X X	– –	– –	– –	5 3	17 10
	RET	11 001 001	C9	$PC_L\leftarrow(SP)$, $PC_H\leftarrow(SP+1)$ SP←SP+2	–	–	X	–	X	–	–	–	3	10
	RET c	11 ccc 000	C0+c×8	If condition c is met, same as RET. If condition c is not met,continue	– –	– –	X X	– –	X X	– –	– –	– –	3 1	11 5
	RETI	11 101 101 01 001 101	ED 4D	Return from interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RETN	11 101 101 01 000 101	ED 45	Return from non-maskable interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RST j	11 kkk 111	C7+k×8	$(SP-1)\leftarrow PC_H$, $(SP-2)\leftarrow PC_L$ $PC_H\leftarrow 0$, $PC_L\leftarrow$j, SP←SP−2	–	–	X	–	X	–	–	–	3	11
INPUT AND OUTPUT	IN A,(n)	11 011 011 nn nnn nnn	DB n	A←(n) n→A0~A7, A→A8~A15	–	–	X	–	X	–	–	–	3	11
	IN r,(C)	11 101 101 01 rrr 000	ED 40+rx8	r←(C) If r=110, only the flags will be affected.	*	*	X	*	X	P	0	–	3	12
	INI	11 101 101 10 100 010	ED A2	(HL)←(C), B←B−1, HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	INIR	11 101 101 10 110 010	ED B2	(HL)←(C), B←B−1, HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	IND	11 101 101 10 101 010	ED AA	(HL)←(C), B←B−1, HL←HL−1	X	*M	X	X	X	X	1	X	4	16
	INDR	11 101 101 10 111 010	ED BA	(HL)←(C), B←B−1, HL←HL−1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUT (n),A	11 010 011 nn nnn nnn	D3 n	(n)←A n→A0~A7, A→A8~A15	–	–	X	–	X	–	–	–	3	11
	OUT (C),r	11 101 101 01 rrr 001	ED 41+rX8	(C)←r	–	–	X	–	X	–	–	–	3	12
	OUTI	11 101 101 10 100 011	ED A3	(C)←(HL), B←B−1, HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	OTIR	11 101 101 10 110 011	ED B3	(C)←(HL), B←B−1, HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUTD	11 101 101 10 101 011	ED AB	(C)←(HL), B←B−1, HL←HL−1	X	*M	X	X	X	X	1	X	4	16
	OTDR	11 101 101 10 111 010	ED BB	(C)←(HL), B←B−1, HL←HL−1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16

j	kkk
00H	000
08H	001
10H	010
18H	011
20H	100
28H	101
30H	110
38H	111

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

*1 [B<>0] ←[B=0]

*1 [B<>0] ←[B=0]

*1 [B<>0] ←[B=0]

*1 C→A0~A7, B→A8~A15

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-Carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign negative

Note :
- *M If the result of B − 1 is zero, the Z flag is set, otherwise it is reset.
- A0 through A15 indicate the address bus.
- [ ] indicates the total condition of the number of cycles and states indicated by arrow.

120489

TMPZ84C013A Instruction Map (1/7)

MPU Instruction Table (I)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	LD BC, mn	LD (BC), A	INC BC	INC B	DEC B	LD B, n	RLCA	EX AF,AF	ADD HC,BC	LD A, (BC)	DEC BC	INC C	DEC C	LD C, n	RRCA
1	DJNZ e	LD DE, mn	LD (DE), A	INC DE	INC D	DEC D	LD D,N	RLA	JE e	ADD HL,DE	LD A, (DE)	DEC DE	INC E	DEC E	LD E, n	RRA
2	JR NZ, e	LD HL, mn	LD (mn),HL	INC HL	INC D	DEC H	LD H, n	DAA	JR Z e	ADD HL,HL	LD HL,(mn)	DEC HL	INC L	DEC L	LD L, n	CPL
3	JR NC, e	LD SP, mn	LD (mn),A	INC SP	INC (HL)	DEC (HL)	LD (HL), n	SCF	JR C e	ADD HL,SP	LD A,(mn)	DEC SP	INC A	DEC A	LD A, n	CCF
4	LD B, B	LD B, C	LD B, D	LD B, E	LD B, H	LD B, L	LD B, (HL)	LD B, A	LD C, B	LD C, C	LD C, D	LD C, E	LD C, H	LD C, L	LD C, (HL)	LD C, A
5	LD D, B	LD D, C	LD D, D	LD D, E	LD D, H	LD D, L	LD D, (HL)	LD D, A	LD E, B	LD E, C	LD E, D	LD E, E	LD E, H	LD E, L	LD E, (HL)	LD E, A
6	LD H, B	LD H, C	LD H, D	LD H, E	LD H, H	LD H, L	LD H, (HL)	LD H, A	LD L, B	LD L, C	LD L, D	LD L, E	LD L, H	LD L, L	LD L, (HL)	LD L, A
7	LD (HL), B	LD (HL), C	LD (HL), D	LD (HL), E	LD (HL), H	LD (HL), L	HALT	LD (HL), A	LD A, B	LD A, C	LD A, D	LD A, E	LD A, H	LD A, L	LD A, (HL)	LD A, A
8	ADD A, B	ADD A, C	ADD A, D	ADD A, E	ADD A, H	ADD A, L	ADD A, (HL)	ADD A, A	ADC A, B	ADC A, C	ADC A, D	ADC A, E	ADC A, H	ADC A, L	ADC A, (HL)	ADC A, A
9	SUB B	SUB C	SUB D	SUB E	SUB H	SUB L	SUB (HL)	SUB A	SBC A, B	SBC A, C	SBC A, D	SBC A, E	SBC A, H	SBC A, L	SBC A, (HL)	SBC A, A
A	AND B	AND C	AND D	AND E	AND H	AND L	AND (HL)	AND A	XOR B	XOR C	XOR D	XOR E	XOR H	XOR L	XOR (HL)	XOR A
B	OR B	OR C	OR D	OR E	OR H	OR L	OR (HL)	OR A	CP B	CP C	CP D	CP E	CP H	CP L	CP (HL)	XOR A
C	RET NZ	POP BC	JP NZ, mn	JP mn	CALL NZ, mn	PUSH BC	ADD A, n	RST 00H	RET C	RET	JP Z, mn	①	CALL Z, mn	CALL mn	ADC A, n	RST 08H
D	RET NC	POP DE	JP NC, mn	OUT (n), A	CALL NC, mn	PUSH DE	SUB n	RST 10H	RET C	EXX	JP C, mn	IN A, (n)	CALL C, mn	③	SBC A, n	RST 18H
E	RET PO	POP HL	JP PO, mn	EX (SP),HL	CALL PO, mn	PUSH HL	AND n	RST 20H	RET PE	JP (HL)	JP PE, mn	EX DE,HL	CALL PI, mn	②	XOR n	RST 28H
F	RET P	POP AF	JP P, mn	DI	CALL P, mn	PUSH AF	OR n	RST 30H	RET M	LD SP, HL	JP M, mn	EI	CALL M, mn	④	CP n	RST 38H

120489

Note ①~④: Multi-Opcode Instructions (ref. Table (II)~(VII))

TMPZ84C013A Instruction Map (2/7)

① Byte 1 "CB"

Instruction Table (II) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	RLC B	RLC C	RLC D	RLC E	RLC H	RLC L	RLC (HL)	RLC A	RRC B	RRC C	RRC D	RRC E	RRC H	RRC L	RRC (HL)	RRC A
1	RL B	RL C	RL D	RL E	RL H	RL L	RL (HL)	RL A	RR B	RR C	RR D	RR E	RR H	RR L	RR (HL)	RR A
2	SLA B	SLA C	SLA D	SLA E	SLA H	SLA L	SLA (HL)	SLA A	SRA B	SRA C	SRA D	SRA E	SRA H	SRA L	SRA (HL)	SRA A
3									SRL B	SRL C	SRL D	SRL E	SRL H	SRL L	SRL (HL)	SRL A
4	BIT 0, B	BIT 0, C	BIT 0, D	BIT 0, E	BIT 0, H	BIT 0, L	BIT 0,(HL)	BIT 0, A	BIT 1, B	BIT 1, C	BIT 1, D	BIT 1, E	BIT 1, H	BIT 1, L	BIT 1, (HL)	BIT 1, A
5	BIT 2, B	BIT 2, C	BIT 2, D	BIT 2, E	BITT 2, H	BIT 2, L	BIT 2, (HL)	BIT 2, A	BIT 3, B	BIT 3, C	BIT 3, D	BIT 3, E	BIT 3, H	BIT 3, L	BIT 3, (HL)	BIT 3, A
6	BIT 4, B	BIT 4, C	BIT 4, D	BIT 4, E	BIT 4, H	BIT 4, L	BIT 4, (HL)	BIT 4, A	BIT 5, B	BIT 5, C	BIT 5, D	BIT 5, E	BIT 5, H	BIT 5, L	BIT 5, (HL)	BIT 5, A
7	BIT 6, B	BIT 6, C	BIT 6, D	BIT 6, E	BIT 6, H	BIT 6, L	BIT 6, (HL)	BIT 6, A	BIT 7, B	BIT 7, C	BIT 7, D	BIT 7, E	BIT 7, H	BIT 7, L	BIT 7, (HL)	BIT 7, A
8	RES 0, B	RES 0, C	RES 0, D	RES 0, E	RES 0, H	RES 0, L	RES 0, (HL)	RES 0, A	RES 1, B	RES 1, C	RES 1, D	RES 1, E	RES 1, H	RES 1, L	RES 1, (HL)	RES 1, A
9	RES 2, B	RES 2, C	RES 2, D	RES 2, E	RES 2, H	RES 2, L	RES 2, (HL)	RES 2, A	RES 3, B	RES 3, C	RES 3, D	RES 3, E	RES 3, H	RES 3, L	RES 3, (HL)	RES 3, A
A	RES 4, B	RES 4, C	RES 4, D	RES 4, E	RES 4, H	RES 4, L	RES 4, (HL)	RES 4, A	RES 5, B	RES 5, C	RES 5, D	RES 5, E	RES 5, H	RES 5, L	RES 5, (HL)	RES 5, A
B	RES 6, B	RES 6, C	RES 6, D	RES 6, E	RES 6, H	RES 6, L	RES 6, (HL)	RES 6, A	RES 7, B	RES 7, C	RES 7, D	RES 7, E	RES 7, H	RES 7, L	RES 7, (HL)	RES 7, A
C	SET 0, B	SET 0, C	SET 0, D	SET 0, E	SET 0, H	SET 0, L	SET 0, (HL)	SET 0, A	SET 1, B	SET 1, C	SET 1, D	SET 1, E	SET 1, H	SET 1, L	SET 1, (HL)	SET 1, A
D	SET 2, B	SET 2, C	SET 2, D	SET 2, E	SET 2, H	SET 2, L	SET 2, (HL)	SET 2, A	SET 3, B	SET 3, C	SET 3, D	SET 3, E	SET 3, H	SET 3, L	SET 3, (HL)	SET 3, A
E	SET 4, B	SET 4, C	SET 4, D	SET 4, E	SET 4, H	SET 4, L	SET 4, (HL)	SET 4, A	SET 5, B	SET 5, C	SET 5, D	SET 5, E	SET 5, H	SET 5, L	SET 5, (HL)	SET 5, A
F	SET 6, B	SET 6, C	SET 6, D	SET 6, E	SET 6, H	SET 6, L	SET 6, (HL)	SET 6, A	SET 7, B	SET 7, C	SET 7, D	SET 7, E	SET 7, H	SET 7, L	SET 7, (HL)	SET 7, A

120489

TMPZ84C013A Instruction Map (3/7)

② Byte 1 "ED"

Instruction Table (III) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4	IN B, (C)	OUT (C), B	SBC HL,BC	LD (mn),BC	NEG	RETN	IM0	LD I, A	IN C, (C)	OUT (C), C	ADC HL,BC	LD BC,(mn)		RETI		LD R, A
5	IN D, (C)	OUT (C), D	SBC HL,DE	LD (mn),DE			IM1	LD A, I	IN E, (C)	OUT (C), E	ADC HL,DE	LD DE,(mn)			IM2	LD A, R
6	IN H, (C)	OUT (C), H	SBC HL,HL	LD (mn),HL				RRD	IN L, (C)	OUT (C), L	ADC HL,HL	LD HL,(mn)				RLD
7			SBC HL,SP	LD (mn),SP					IN A, (C)	OUT (C), A	ADC HL,SP	LD SP,(mn)				
8																
9																
A	LDI	CPI	INI	OUTI					LDD	CPD	IND	OUTD				
B	LDIR	CPIR	INIR	OTIR					LDDR	CPDR	INDR	OTDR				
C																
D																
E																
F																

120489

TMPZ84C013A Instruction Map (4/7)

③ Byte 1 "DD"

Instruction Table (Ⅳ) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0										ADD IX,BC						
1										ADD IX,DE						
2		LD IX, mn	LD (mn), IX	INC IX						ADD IX, IX	LD IX,(mn)	DEC IX				
3					INC (IX + d)	DEC (IX + d)	LD (IX + d) , n			ADD IX, SP						
4							LD B, (IX + d)								LD C, (IX + d)	
5							LD D, (IX + d)								LD E, (IX + d)	
6							LD H, (IX + d)								LD L, (IX + d)	
7	LD (IX + d) , B	LD (IX + d) , C	LD (IX + d) , D	LD (IX + d) , E	LD (IX + d) , H	LD (IX + d) , L		LD (IX + d) , A							LD A, (IX + d)	
8							ADD (IX + d)								ADC 1, (IX + d)	
9							SUB (IX + d)								SBC A, (IX + d)	
8							AND (IX + d)								XOR (IX + d)	
9							OR (IX + d)								CP (IX + d)	
C												⑤				
D																
E		POP IX		EX (SP), IX		PUSH IX				JP (IX)						
F										LD SP, IX						

120489

Note ⑤: Special 2 byte Opcode Instructions (ref. Table (Ⅵ))

TMPZ84C013A Instruction Map (5/7)

④ Byte 1 "FD"

Instruction Table (Ⅴ) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0										ADD IY,BC						
1										ADD IY,DE						
2		LD IY, nn	ID (mn), IY	INC IY						ADD IY, IY	LD IY, (mn)	DEC IY				
3					INC (IY + d)	DEC (IY + d)	LD (IY + d) , n			ADD IY, SP						
4							LD B, (IY + d)								LD C, (IY + d)	
5							LD D, (IY + d)								LD E, (IY + d)	
6							LD H, (IY + d)								LD L, (IY + d)	
7	LD (IY + d) , B	LD (IY + d) , C	LD (IY + d) , D	LD (IY + d) , E	LD (IY + d) , H	LD (IY + d) , L		LD (IY + d) , A							LD A, (IY + d)	
8							ADD (IY + d)								ADDC A, (IY + d)	
9							SUB (IY + d)								SUBC A, (IY + d)	
8							AND (IY + d)								XOR (IY + d)	
9							OR (IY + d)								CP (IY + d)	
C												⑥				
D																
E		POP IY		EX (SP), IY		PUSH IY				JP (IY)						
F										LD SP, IY						

120489

Note ⑥: Special 2 byte Opcode Instructions (ref. Table (Ⅶ))

TMPZ84C013A Instruction Map (6/7)

⑤ Byte 1 "DD"

Byte 2 "CB"

Instruction Table (VI) (Special case of 2-byte Opcode : Byte 3)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0							RLC (IX + d)								RRC (IX + d)	
1							RL (IX + d)								RR (IX + d)	
2							SLA (IX + d)								SRA (IX + d)	
3															SRL (IX + d)	
4							BIT 0, (IX + d)								BIT 1, (IX + d)	
5							BIT 2, (IX + d)								BIT 3, (IX + d)	
6							BIT 4, (IX + d)								BIT 5, (IX + d)	
7							BIT 6, (IX + d)								BIT 7, (IX + d)	
8							RES 0, (IX + d)								RES 1, (IX + d)	
9							RES 2, (IX + d)								RES 3, (IX + d)	
A							RES 4, (IX + d)								RES 5, (IX + d)	
B							RES 6, (IX + d)								RES 7, (IX + d)	
C							SET 0, (IX + d)								SET 1, (IX + d)	
D							SET 2, (IX + d)								SET 3, (IX + d)	
E							SET 4, (IX + d)								SET 5, (IX + d)	
F							SET 6, (IX + d)								SET 7, (IX + d)	

120489

TMPZ84C013A Instruction Map (7/7)

⑥ Byte 1 "FD"

Byte 2 "CB"

Instruction Table (VII) (Special case of 2-byte Opcode : Byte 3)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0							RLC (IY + d)								RRC (IY + d)	
1							RL (IY + d)								RR (IY + d)	
2							SLA (IY + d)								SRA (IY + d)	
3															SRL (IY + d)	
4							BIT 0, (IY + d)								BIT 1, (IY + d)	
5							BIT 2, (IY + d)								BIT 3, (IY + d)	
6							BIT 4, (IY + d)								BIT 5, (IY + d)	
7							BIT 6, (IY + d)								BIT 7, (IY + d)	
8							RES 0, (IY + d)								RES 1, (IY + d)	
9							RES 2, (IY + d)								RES 3, (IY + d)	
A							RES 4, (IY + d)								RES 5, (IY + d)	
B							RES 6, (IY + d)								RES 7, (IY + d)	
C							SET 0, (IY + d)								SET 1, (IY + d)	
D							SET 2, (IY + d)								SET 3, (IY + d)	
E							SET 4, (IY + d)								SET 5, (IY + d)	
F							SET 6, (IY + d)								SET 7, (IY + d)	

120489

3.3. CGC OPERATIONS

This subsection describes the system configuration, functions, and basic operations of the clock generator/controller (CGC).

3.3.1 Block Diagram

Figure 3.3.1 shows the block diagram of CGC.

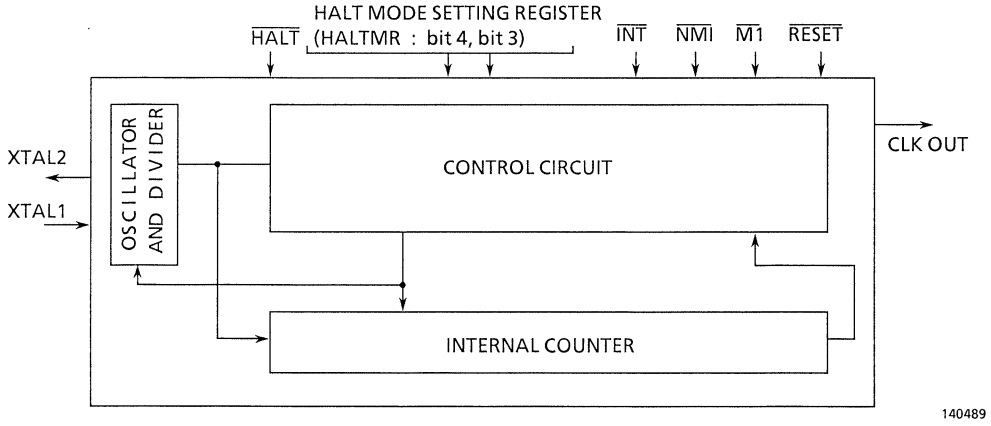


Figure 3.3.1 Block Diagram

3.3.2 CGC System Configuration

The internal configuration of the CGC is shown in Figure 3.3.1. The waveform from the external oscillator oscillated by the internal oscillator and divided by the divider is converted into the square wave for clock. The clock is controlled by the control circuit and the counter to be sent to the outside the CGC. The following describes the CGC's main components and their functions.

(1) Clock Generation

(2) Operation Modes

[1] Clock Generation

The CGC contains an oscillation circuit. By connecting oscillator to external pins (XTAL1 and XTAL2), the required clock can be generated easily. The CGC provides the clock whose frequency is 1/2 of the osicillation frequency. Figure 3.3.2 shows an example of oscillator connection.

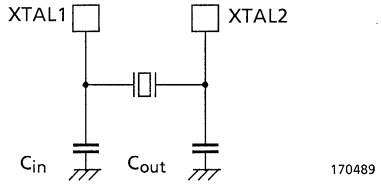


Figure 3.3.2 (a) Example of Crystal Connection

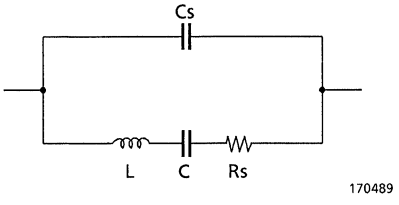


Figure 3.3.2 (b) Oscillator Equivalent Circuit

(1) For the quartz crystal oscillator, use the MR8000-C20 (oscillation frequency 8 MHz) or MR12000-C20 (oscillation frequency 12 MHz) manufactured by Tokyo Denpa Company Ltd., or the equivalent;

Product No.	Holder Type	Frequency (MHz)	Cin (pF)	Cout (pF)	Quartz Crystal Parameter (Typ.)			Drive Level (mW)	Condition Load Capacitance (pF)
					C_1 (pF)	C_0 (pF)	R_1 (Ω)		
MR8000-C20	HC-49 -U (TR-49)	8	22	33	—	4.00	30.0	—	—
MR8000-C14		8	20	20	0.0189	3.87	6.0	0.5	12.67
MR12000-C20		12	33	33	—	4.00	25.0	—	—
MR12000-C14		12	20	20	0.0190	3.81	6.9	0.5	12.55
MR16000-C14		16	20	20	0.0197	4.00	5.7	0.5	12.20

190589

Note : The load capacitance in the condition does not include any stray capacitance.

(2) For the ceramic resonater, use the CSA8.00MT, CST8.00MT (oscillation frequency 8 MHz) or CSA12.00MT, CST12.00MT (osillation frequency 12 MHz) manufactured by Murata MFG Co., Ltd.

Product No.	Frequency (MHz)	Cin (pF)	Cout (pF)
CSA8.00MT	8	30	30
CST8.00MT	8		
CSA12.0MT	12	30	30
CST12.0MT	12		

190589

Note : The CST8.00MT and CST12.0MT need no outer capacitance.

[2] Operation Modes

The CGC has the capability to control 4 operation modes;Run, Idle 1, Idle 2, and Stop. Any one of them can be selected through the mode setting register (#F0:bit4, bit3:HALTMR). These modes become valid when the MPU executes a HALT instruction. Fetching a HALT instruction, the MPU sets the $\overline{HALT}$ signal to "0", indicating that it has been put in the halt state. After the execution of the HALT instruction, the CGC performs the operation in the specified mode. Table 3.3.1 shows the operations in each mode.

Table 3.3.1 CGC Operation Modes

Halt mode setting register (#F0:HALTMR)		Operational Mode	Description
Bit 4	Bit 3		
0	0	Idle 1 Mode	Only the internal oscillator operates, stopping the supply of clock outside. The clock output (CLKOUT) is held at "0".
0	1	Idle 2 Mode	The internal oscillator continues operating with the supply of clock outside (CLKOUT) continued. When pins CLKOUT and CLKIN are connected, only the supply of clock (CLKOUT) to the CTC is continued.
1	0	Stop Mode	All internal operations are stopped. The clock output (CLKOUT) is held at "0".
1	1	Run Mode	The supply of clock outside is continued.

140489

The restart from the clock stop state in Idle 1, Idle 2 (these two modes are referred to as Idle mode hereafter), or Stop mode is performed by reset ($\overline{RESET}$ signal) or acknowledge of maskable interrupt ($\overline{INT}$ signal) or non-maskable interrupt ($\overline{NMI}$ signal).

[3] Warm-up Time for Restart (from Stop mode)

Releasing the halt state by interrupt acknowledge, the MPU begins executing interrupt processing. Therefore, when restarting the clock by the $\overline{NMI}$ or $\overline{INT}$ restart signal in the Stop mode, the oscillation must be fully stabilized before supplied outside. The CGC provides, by means of the internal counter, the warm-up time enough for the clock to stabilize frequency. The warm-up ends on the rising edge of the internal counter output dividing the oscillation frequency to start clock output. The warm-up time is equal to the time derived by dividing the frequency of the externally attached oscillator by 2^{14}.

Figure 3.3.3 shows the block diagram of the internal counter. Table 3.3.2 shows the relationship between the oscillation frequency and the warm-up time.

In the restart by the $\overline{RESET}$ signal, no warm-up is performed for the quick operation at power-on. Therefore, expand the width of the $\overline{RESET}$ signal adequately to provide the warm-up time.

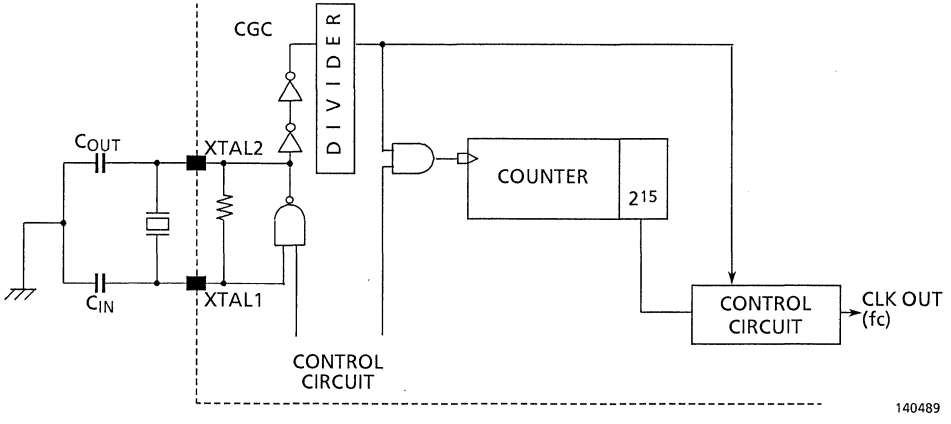


Figure 3.3.3 Block Diagram of Internal Counter

* fc = fxTAL/2

Table 3.3.2 Warm-up Time

Counter output	Warm-up Time		
		fXTAL = 12MHz	fXTAL = 8MHz
2^{15}	2^{14}/fc	2.7 ms	4 ms

140489

3.3.3 CGC Status Transition Diagram and Basic Timing

The following describes the status transition and basic timing to be provided when the CGC operates.

[1] Status Transition Diagram

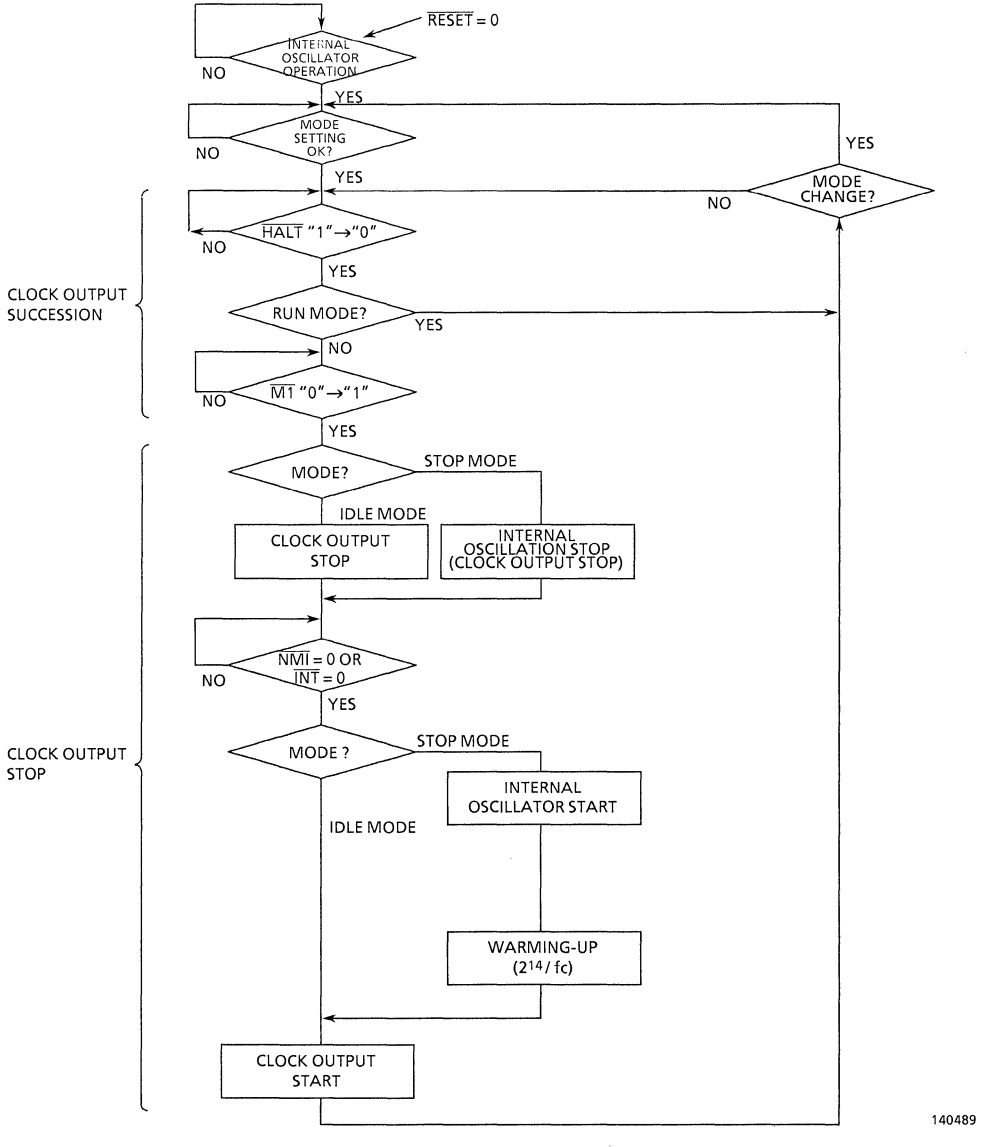


Figure 3.3.4 Status Transition Diagram

[2] Basic Timing

The following describes the CGC basic timing when the CGC clock output pin (CLKOUT) and clock input pin (CLKIN) are connected.

(1) Operation at execution of HALT instruction

The following describes the basic timing in each mode to be provided when the MPU executes a HALT instruction. The MPU sets the $\overline{\text{HALT}}$ signal to "0" synchronized with the falling edge of clock state T4 of the HALT instruction Opcode fetch cycle (M1). This signal indicates to the CGC that the MPU is going to enter into the halt states.

(a) Run mode (#F0:bit 4 = 1, bit 3 = 1:HALTMR)

Figure 3.3.5 shows the basic timing in the Run mode. In the Run mode, the CGC continues supplying the clock to the outside even when the MPU is in the halt state. Therefore, the MPU continues executing NOPs during the halt state. The systems which need memory address refresh use this mode.

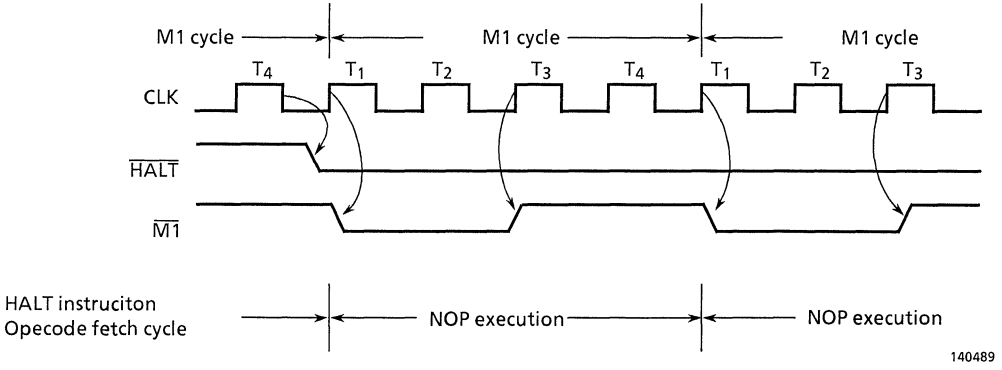


Figure 3.3.5 Basic Timing in Ruin Mode

(b) Idle 1 mode (#F0:bit 4=0, bit 3=0:HALTMR), idle 2 mode (#F0:bit 4=0, bit 3=0:HALTMR), and Stop mode (#F0:bit 4=1, bit 3=0)

Figure 3.3.6 shows the basic timing in the Idle modes and Stop mode. In these modes, the clock output is stopped with clock state T4 being "0" by the $\overline{\text{HALT}}$ signal and the $\overline{\text{M1}}$ signal which follows the HALT instruction.

However, in the Stop mode, the CGC's internal oscillator also stops.

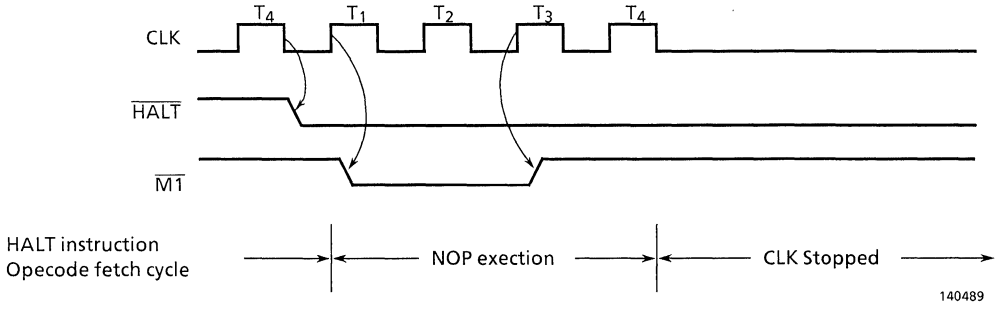


Figure 3.3.6 Basic Timing in Idle and Stop Modes

(2) Clock output restart from each mode

The clock stopped state in the Idle or Stop mode is cleared by setting any of the following signals to "0" (for the system restart operation, see Subsection 3.3.4) :

- $\overline{INT}$ (level trigger input)
- $\overline{NMI}$ (edge trigger input)
- $\overline{RESET}$ (level trigger input)

(a) Clock output restart from Idle mode

Figure 3.3.7 (a) shows the basic timing for the sequence of the output restart from the clock stopped state in the Idle 1 mode. In the restart in the Idle 1 mode, the clock output is restarted in a relatively short delay time because the internal oscillator operates even in the clock stopped state.

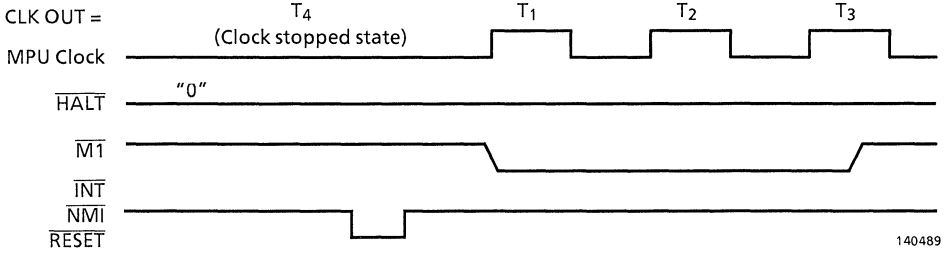
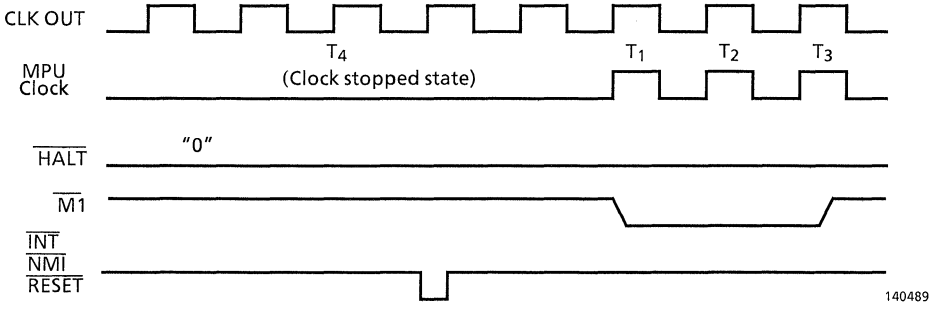


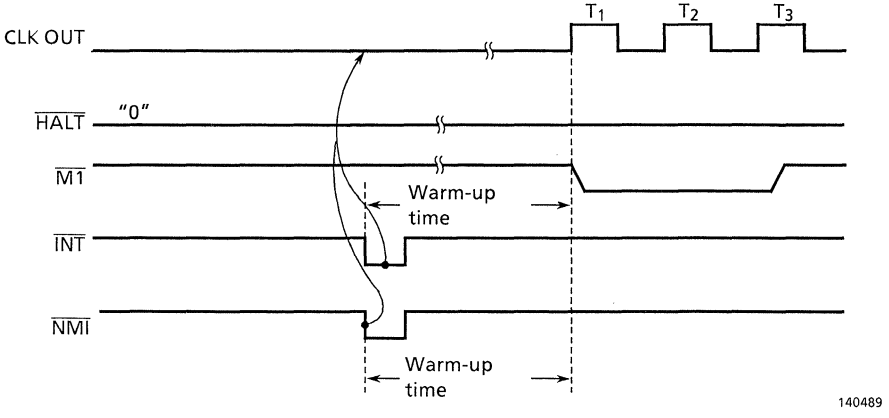
Figure 3.3.7 (a) Basic Timing for Sequence of Restart from Clock stopped State (Idle 1 MOde)

(b) Clock output restart from Stop mode

Figure 3.3.8 shows the basic timing for the sequence of the restart from the clock stopped state in the Stop mode. When restarting by setting the $\overline{INT}$ or $\overline{NMI}$ signal to "0", the warm-up time is automatically created by the internal counter. In the restart by the $\overline{RESET}$ signal, oscillation restarts without warm-up.



Figre 3.3.7 (b) Basic Timing for Sequence of Restart from Clock stopped State (Idle 2 Mode)



Figre 3.3.8 Basic Timing for Sequence of Restart from Clock stopped State (Stop Mode)

3.3.4 Relationship with MPU

The following describes the relationship between the CGC and the MPU mainly in terms of the halt clear operation.

[1] RESET Signal

Figure 3.3.9 shows an example of the timing for the restart from the Stop mode on the TMPZ84C013A using $\overline{\text{RESET}}$ signal for both the MPU and CGC . To reset the MPU, the $\overline{\text{RESET}}$ signal must be set to "0" for at least 3 stable clocks. When the $\overline{\text{RESET}}$ signal goes "1", the MPU releases the halt state after a dummy cycle of 2T clock states to start executing instructions from address 0000H.

To restart the clock output by the $\overline{\text{RESET}}$ signal in the Stop mode, the internal counter to determine the warm-up time does not operate.
Therefore, if the MPU does not restart correctly due to the unstable clock output immediately after the restart of the internal oscillator, or the unstability of the crystal at power-on, the $\overline{\text{RESET}}$ signal must be held at "0" for a time long enough for the MPU to be reset securely.

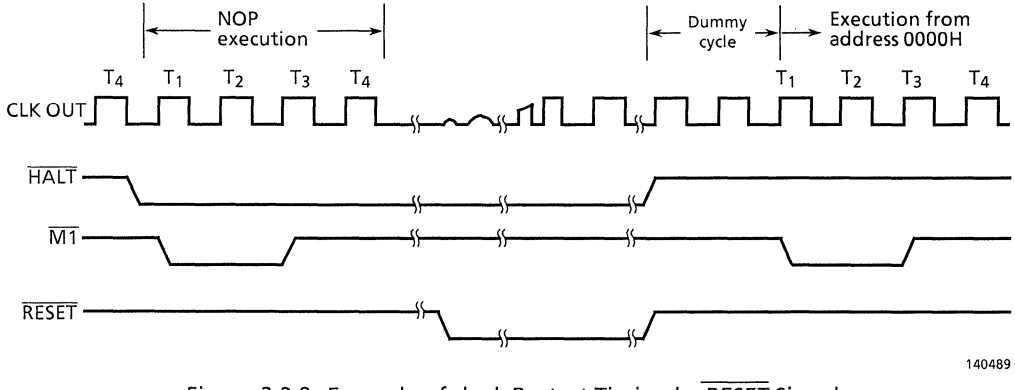


Figure 3.3.9 Example of clock Restart Timing by $\overline{\text{RESET}}$ Signal

[2] Releasing Halt State by Interrupt Signal

The CGC restarts the clock output from the Idle or Stop mode by the input of $\overline{\text{INT}}$ or $\overline{\text{NMI}}$ signal. By this clock, the MPU starts operating. However, when the CGC restarts the clock output, the MPU is still in the halt state executing NOPs. To clear the halt state, the interrupt signal must be entered into the MPU (in the case of the $\overline{\text{INT}}$ signal) for at least one instruction. The MPU interrupt is detected on the rising edge of the last clock of each instruction (NOP for the halt state).

(1) When using non-maskable interrupt (NMI)

MPU's non-maskable interrupt is edge trigger input. The MPU contains the flip-flop to detect an interrupt. The state of this internal NMI flip-flop is sampled on the rising edge of the last clock of each instruction. Therefore, when a short active low ("0") pulse has been inserted before the interrupt detection timing, the interrupt is acknowledged. The $\overline{NMI}$ input of the TMPZ84C013A is connected to the $\overline{NMI}$ input of the MPU via the CGC, performing the same operations as above. (See Figure 3.3.11)

(2) When using maskable interrupt (INT)

With a maskable interrupt, the maskable interrupt enable flip-flop (IFF) must be set to "1" by program before the $\overline{INT}$ input signal is detected "0". Even if the CGC accepts the $\overline{INT}$ signal to restart supply of the clock, no interrupt is acknowledged unless the $\overline{INT}$ signal is kept inserted until one instruction (NOP) has been executed. Figure 3.3.10 shows the timing for clearing the halt state by the interrupt signal.

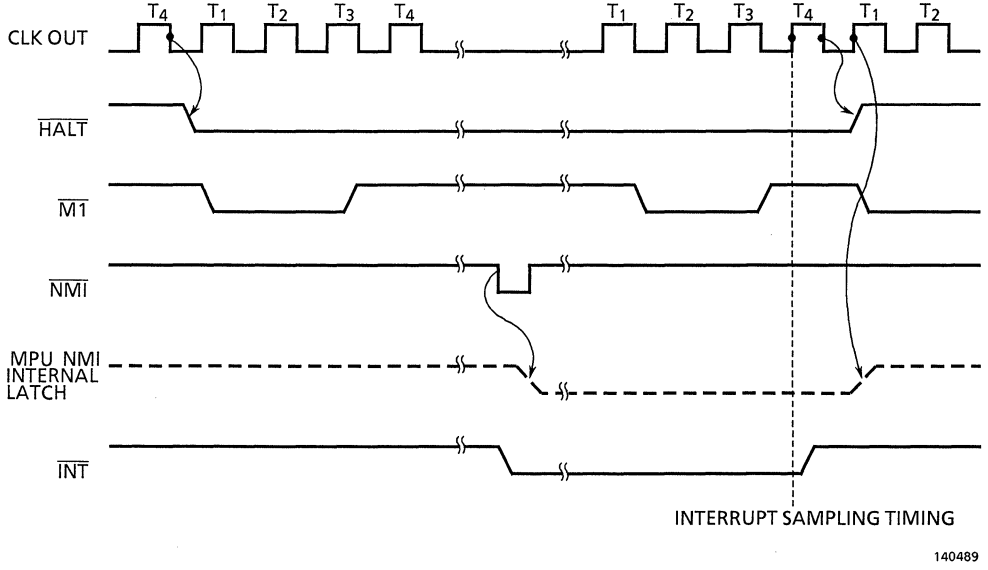


Figure 3.3.10 Timing for Clearing Halt State by Interrupt Signal

[3] Connecting CGC to MPU on TMPZ84C013A

Figure 3.3.11 shows the connection between the CGC and the MPU on the TMPZ84C013A.

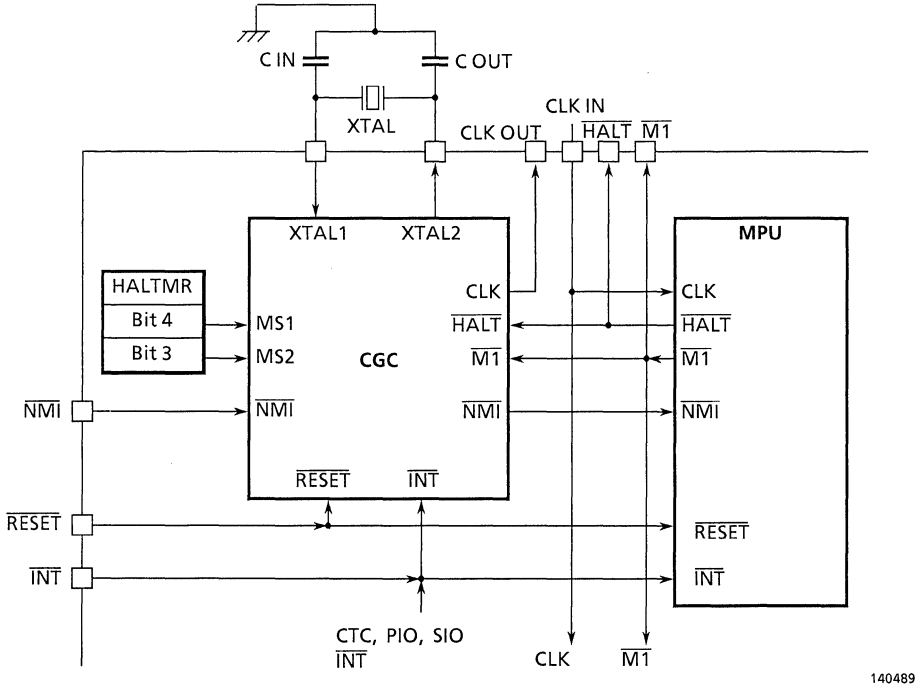


Figure 3.3.11 Connection Between CGC and MPU

3.4 CTC OPERATIONAL DESCRIPTION

The CTC has 4 independent channels. To these channels, addresses are allocated on the TMPZ84C013A's I/O map, permitting the read/write of the channels in the MPU's I/O cycle. (See Figure 3.4.1) This subsection mainly describes the CTC operation to be performed after accessed.

3.4.1 CTC Block Diagram

Figure 3.4.1 shows the block diagram of the CTC.

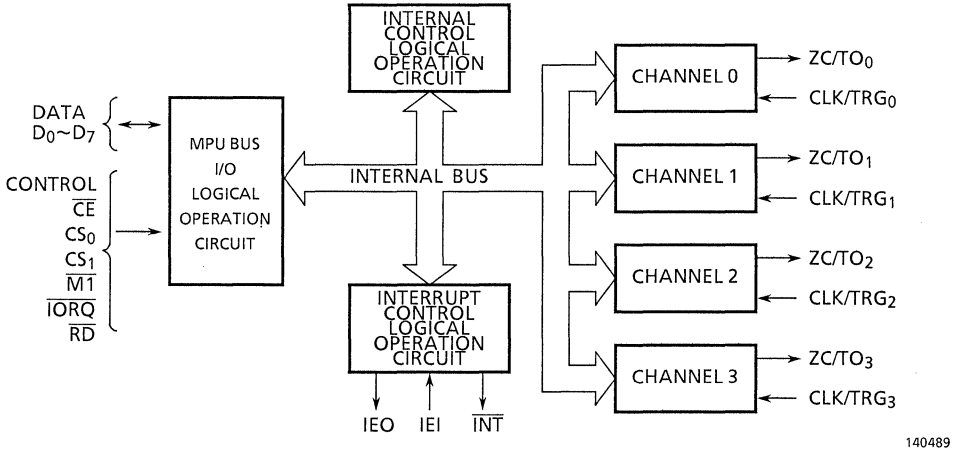


Figure 3.4.1 Block Diagram of CTC

3.4.2 CTC System Configuration

The CTC system consists of the following 4 logic circuits:

(1) MPU bus I/O logic circuit

(2) Internal control logic circuit

(3) Interrupt control logic circuit

(4) Four independent counter/timer channel logic circuits

[1] MPU Bus I/O Logic Circuit

This circuit transfers data between the MPU and the CTC.

[2] Internal Control Logic Circuit

This circuit controls the CTC operational functions such as the CTC selecting chip enable, reset, and read/write circuits.

[3] Interrupt Control Logic Circuit

This circuit performs the MPU interrupt related processing such as priority determination. The order of priority with other LSIs is determined according to the physical location in daisy chain connection.

[4] Counter/Timer Channel Logic Circuit

This circuit consists of the following 2 registers and 2 counters.
Figure 3.4.2 shows the configuration of this circuit.

- Time-constant register (8 bits)
- Channel control register (8 bits)
- Down-counter (8 bits)
- Prescaler (8 bits)

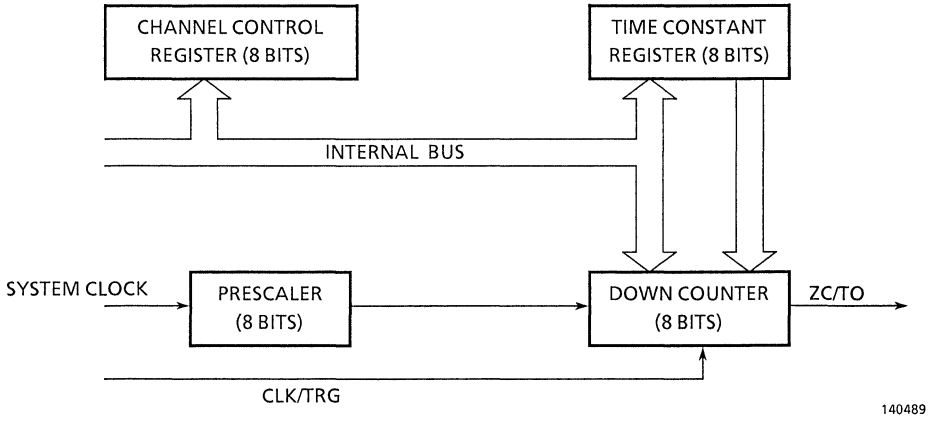


Figure 3.4.2 Configuration of Counter/Timer Channel Logic Circuit

(1) Time-constant register

This register holds the time constant to be written in the down counter. When the CTC is initialized or the down-counter has reached zero, the time constant is loaded into down-counter. The time constant is set immediately after the MPU has written the channel control word in the channel control register. For a time constant, an integer from 1 to 256 can be used.

(2) Channel control register

This register is used to choose the channel mode or condition according to the channel control word sent from the MPU.

(3) Down-counter

The contents of the time-constant register are loaded into the down counter. In the counter mode, these contents are decremented at each edge of the external clock;in the timer mode, they are decremented for each prescaler clock output. The contents of the time-constant register are loaded at initialization or when the down-counter has reached zero.

The contents of the down-counter can be read any time. Also, the system can be programmed so that an interrupt request is generated each time the down-counter has reached zero.

(4) Prescaler

The prescaler, used only in the timer mode, divides the system clock by 16 or 256. The dividing number is programmed by channel control word. The output of the prescaler becomes the clock input to the down-counter.

3.4.3 CTC Basic Operations

[1] Reset

The state of the CTC is unstable after it is powered on. To stabilize the CTC, the low level signal needs to be applied to the $\overline{\text{RESET}}$ pin. On any channel, the channel control word and time-constant data must be written to be started before it is started in the counter or timer mode. To program the system to enable interrupts, the interrupt vector word must be written in the interrupt controller. When these data have been written in the CTC, it is ready to start.

[2] Interrupt

The CTC can cause an interrupt when the MPU is operating in the mode 2. The CTC interrupt can be programmed for each channel. Each time the channel's down-counter has reached zero, the CTC outputs the interrupt request signal ($\overline{\text{INT}}$). When the MPU accepts the CTC's interrupt request, the CTC outputs the interrupt vector. Based on this interrupt vector, the MPU specifies the start address of the interrupt processing routine and calls it to start interrupt processing.

The MPU specifies the start address of the interrupt processing routine by the interrupt vector output from the CTC, so that the user can change the vector value to call any desired address.

The interrupt processing is terminated when the MPU executes an RETI instruction. The CTC has the circuit which decodes the RETI instruction. By constantly monitoring the data bus the CTC can detect the termination of the interrupt processing.

The order of interrupt priority with the Z80 peripheral LSIs is determined by the daisy chain connection. That is, the peripheral LSIs are connected one after another and the one physically near the MPU is given a higher priority. The priority of the Z80 peripheral LSIs (CTC, PIO, and SIO) contained in the TMPZ84C013A is determined by the contents of the interrupt priority register (#F4:bits 2 through 0). Inside the CTC, channel 0 is given the highest priority, followed by channels 1, 2 and 3 in this order.

The CTC and other peripheral LSIs on the TMPZ84C013A have the signal lines IEO and IEI. Connect the IEO of a higher peripheral LSI to the IEI of a lower perpheral LSI. Connect the IEI of the highest peripheral LSI to VCC. Leave the IEO of the peripheral LSI unused. In this connection, the CTC interrupt is caused under the following conditions:

- When both IEI and IEO are high, no interrupt is caused. At this time, the $\overline{INT}$ signal is high. An interrupt can be requested in this state.
- When the CTC outputs the interrupt request signal ($\overline{INT}$), the IEO of the CTC becomes low. When the MPU accepts the interrupt, the $\overline{INT}$ goes high again.
- When the IEI goes low, the IEO also goes low.
- While the IEI is low, no interrupt can be requested.
- When the IEI goes low while an interrupt is being serviced, the interrupt processing is aborted.

[3] Operation Modes

The CTC operates in either the counter mode or the timer mode. Mode is selected by writing the channel control word.

(1) Counter mode

In the counter mode, the number of edge of the pulses applied to the channel's CLK/TRG pin is counted. When pulses have been input, the contents of the down-counter are decremented synchronizing with the rising edge of the next system clock. The pulse's rising edge or falling edge to be counted can be specified by the channel control word.

When the contents of the down-counter has reached zero, the high level pulse is output from the ZC/TO pin. When the interrupt is enabled by the channel control word, the $\overline{INT}$ pin goes low and an interrupt is requested. When the contents of the down-counter has reached zero, the time constant data written in the time constant register is automatically loaded into the down-counter. To load a new time constant value into the down-counter, write the data to the time constant register, and it is loaded into the down-counter after the current count operation is terminated.

(2) Timer mode

In the timer mode, the time intervals which are integral multiples of the system clock period. A timer interval is measured according to the system clock. The system clock is supplied to the prescaler which divides it by a factor of 16 or 256. The output of the prescaler provides the clock to decrement the down-counter by 1. The time constant data is automatically loaded into the down-counter each time it has reached zero as in the counter mode. When the contents of the down-counter has reached zero, the high level pulse is output from the ZC/TO pin.

This pulse period is given by the following expression:

tc*P*TC

Where, tc = System clock period
P = Prescaler value (16 or 256)
TC = Time constant data (256 for 00H)

The user can select, by means of the channel control word, to start the timer automatically or to start the timer on the edge of the pulse at CLK/TRG pin. In case the user select the CLK/TRG pin, the user can also select the rising edge or falling edge of the pulse.

3.4.4 CTC Status Transition Diagram and Basic Timing

[1] Transition Diagram

Figure 3.4.3 shows the CTC status transition diagram.

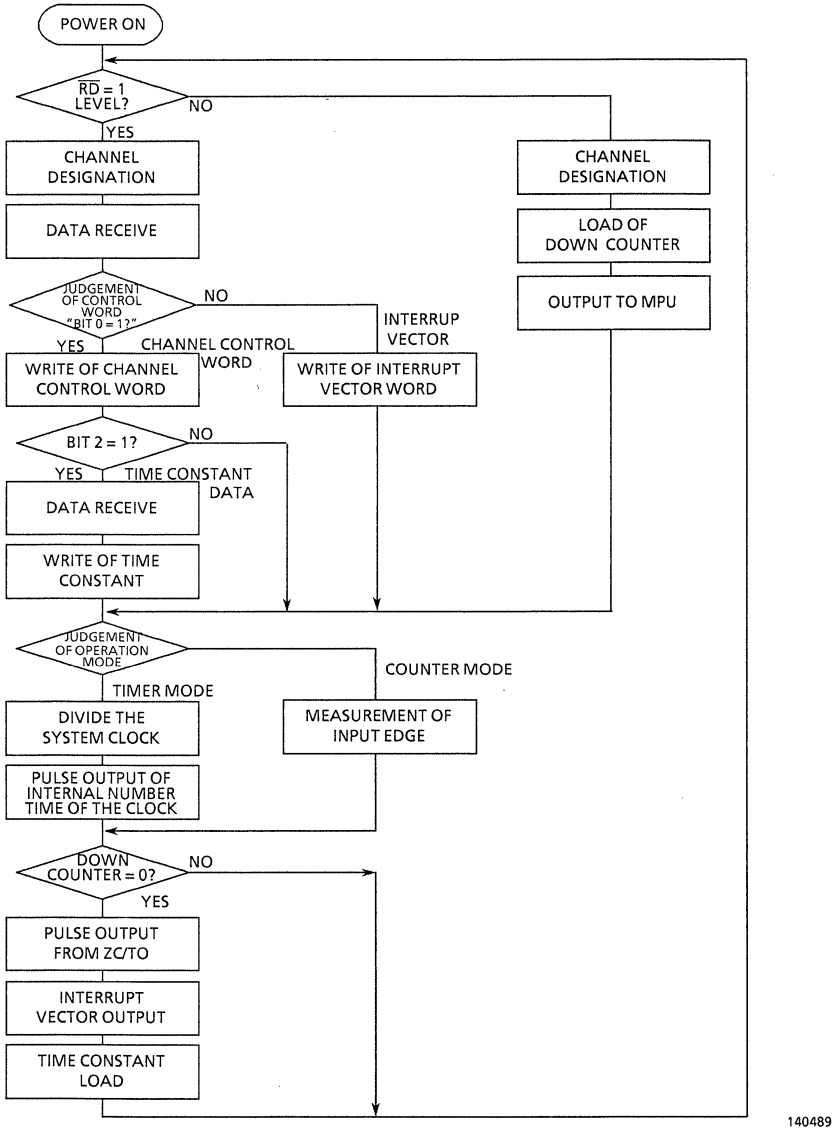


Figure 3.4.3 (a) CTC Transition Diagram (a)

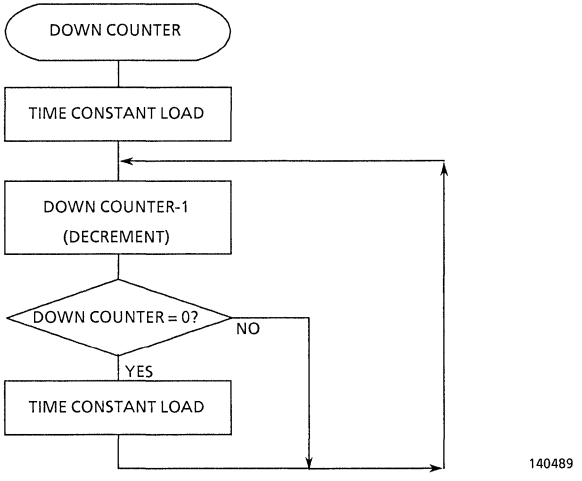


Figure 3.4.3 (b) CTC Transition Diagram (b)

[2] Basic Timing

(1) Write cycle

The write cycle is used to write a channel control word, an interrupt vector, or a time constant. The MPU drives the $\overline{IORQ}$ pin of the CTC low in the subsequent system clock cycle T2 to start the write cycle.
It is required to make the $\overline{M1}$ pin of the CTC high to indicate that the write cycle is on.

At the start of the cycle, the channel is specified by CS1 (A1) or CSO (A0) of the CTC. Thus, the CTC's internal registers are ready to accept data in system clock T3. Tw is the state to be automatically inserted by the MPU.

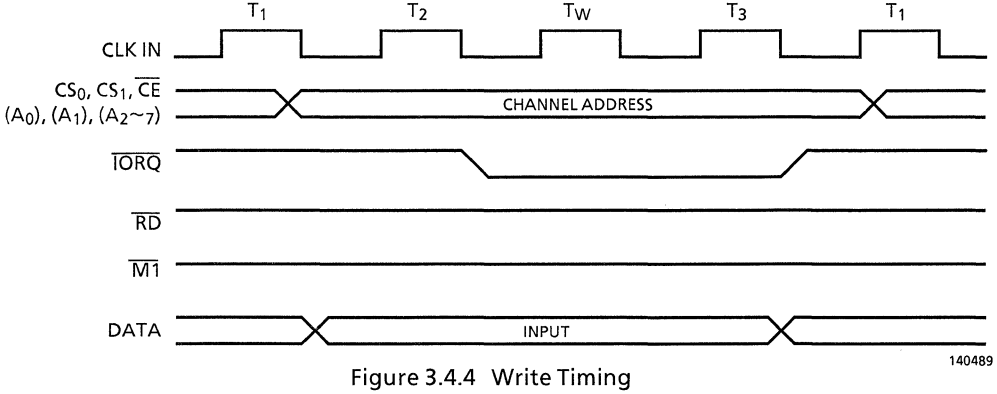


Figure 3.4.4 Write Timing

(2) Read cycle

The read cycle is used to read the contents of the down-counter. During clock cycle T2, the MPU initiates a read cycle by driving the $\overline{RD}$ and $\overline{IORQ}$ pins low. It is required to make the $\overline{M1}$ pin high to indicate that the read cycle is on. At the start of the read cycle, the channel is specified by CS1 (A1) or CS0 (A0) of the CTC.

On the rising edge of system clock TW, the contents of the down-counter at the time of the rising edge of T2 are put on the data bus. TW is the wait state to be automatially inserted by the MPU.

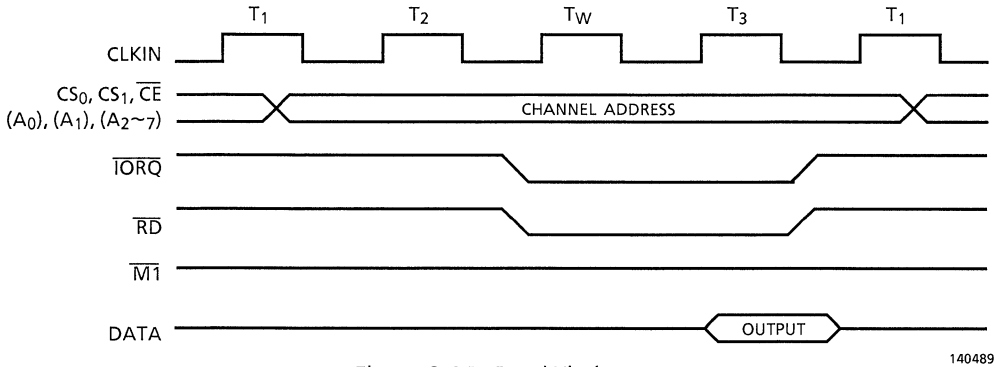


Figure 3.4.5 Read Timing

[3] Counter mode

In the counter mode, the down-counter is decremented synchronizing with the system clock, at the edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the down-counter is decremented one system clock later. When the down-counter has reached zero, a high level pulse is output from the ZC/TO pin.

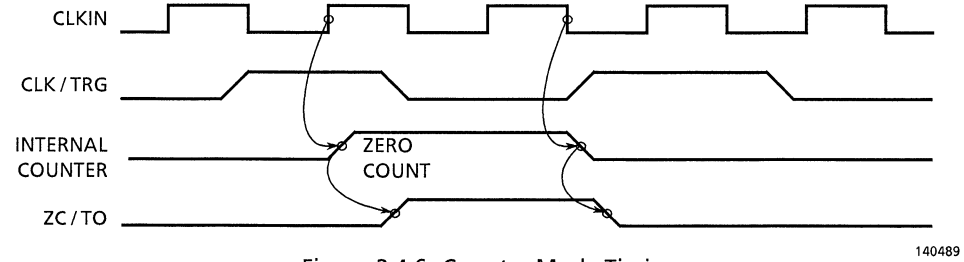


Figure 3.4.6 Counter Mode Timing

[4] Timer mode

The timer starts operating at the second rising edge of the system clock from the rising edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the timer starts one system clock cycle later.

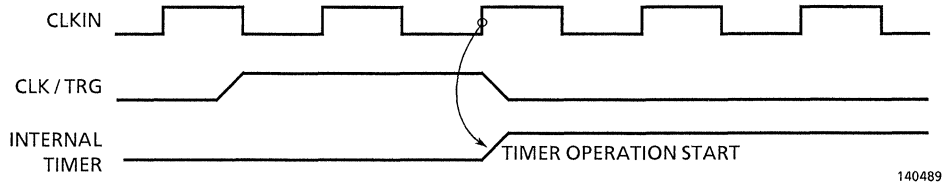


Figure 3.4.7 Timer Mode Timing

[5] Interrupt acknowledge cycle

Having received the interrupt request signal ($\overline{INT}$) from the CTC, the MPU drives the CTC's $\overline{M1}$ pin and $\overline{IORQ}$ pin low to provide the acknowledge signal. The $\overline{IORQ}$ pin goes low 2.5 system clocks later than the $\overline{M1}$ pin. To stabilize the signal lines (IEI and IEO) in daisy chain connection, the interrupt request cannot be changed on each channel while the $\overline{M1}$ pin is low. The $\overline{RD}$ pin is held high to make distinction between the instruction fetch cycle and the interrupt acknowledge cycle. While the $\overline{RD}$ pin is high, the CTC's interrupt control circuit determines the interrupt-requesting channel of highest priority. When the CTC's IEI is high and the $\overline{M1}$ pin and $\overline{IORQ}$ pin go low, the interrupt vector is output from the interrupt requesting channel of highest priority on the data bus. At this time, 2 system clock cycles are automatically inserted by the MPU as a wait state to maintain the stabilization of the daisy chain connection.

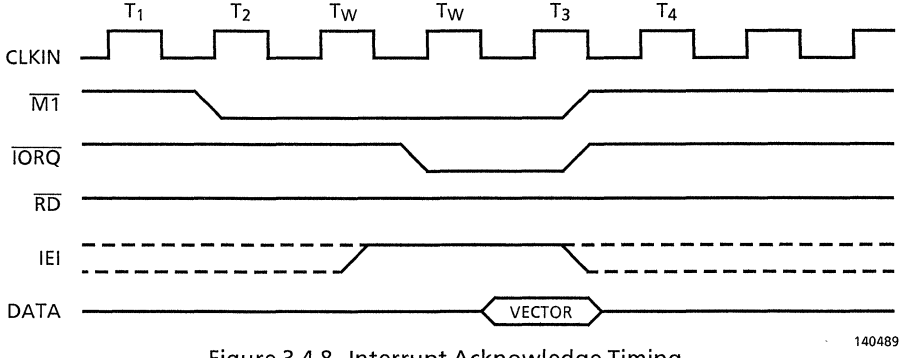


Figure 3.4.8 Interrupt Acknowledge Timing

[6] Return from interrupt processing

Return from the interrupt processing is performed when the MPU executes the RETI instruction. This RETI instruction must be used at the end of the interrupt processing routine. When this instruction is executed by the MPU, the CTC's IEI and IEO return to the state active before the interrupt has been serviced.

The RETI instruction is a 2-byte instruction. Its code is EDH 4DH. The CTC decodes this instruction to check if there is the next interrupt request channel.

In the daisy chain structure, the interrupting LSI's IEI and IEO are held high and low respectively at the time the instruction code EDH has been decoded.

The code following EDH is 4DH, only the peripheral LSI which has sent the last interrupt vector (that is, the LSI whose IEI is high and IEO is low) returns from the interrupt processing. This restarts the processing of the suspended interrupt of the peripheral LSI of the next higher priority.

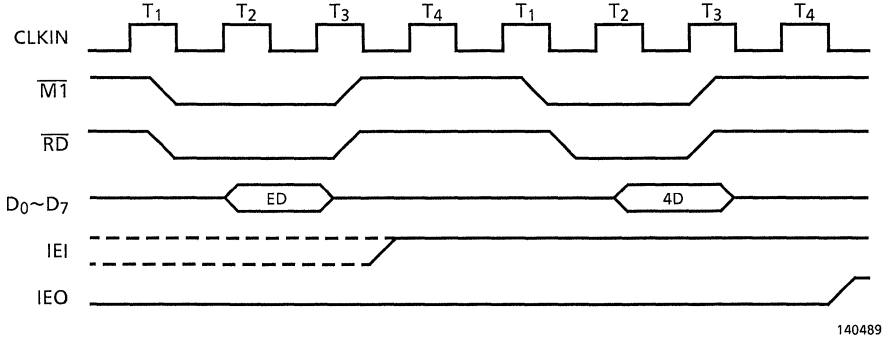


Figure 3.4.9 Interrupt Return Timing

3.4.5 CTC Operational Procedure

To operate the CTC in the counter mode or the timer mode, the channel control word and time-constant data must be written in the CTC. To enable interrupts by the channel control word, the interrupt vector must be written in the CTC.

[1] I/O Address and Channel Control Word

To write the channel control word in the CTC, the channel must be specified by the corresponding channel I/O address. Table 3.4.1 shows the channel I/O addresses.

Table 3.4.1 Channel I/O Addresses

Channel	I/O address
0	#10
1	#11
2	#12
3	#13

140489

The channel control word to be written in the CTC consists of 8 bits. The system data bus D0 through D7 correspond to bit 0 through 7 respectively. Figure 3.4.10 shows the meaning of each bit. Table 3.4.2 shows the function of each bit.

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt	Counter/ timer	Prescaler	Edge	Trigger	Time constant	Reset	1

140489

Figure 3.4.10 Channel Control Word

For the channel control word, D0 must be always 1.

Table 3.4.2 Meanings and Function of Channel Control Words (1/3)

Bit	Meaning and function	
	0	1
Bit 7 (D7)	Disables channel interrupt	Enables channel interrupt. In either counter or timer mode, the interrupt is requested every time the down-counter has reached zero.When this bit is set to "1", the interrupt vector must be written in the CTC before the down-counter starts. When the channel control word whose D7 bit is "1" is written in an already operating channel, the interrupt occurs only when the down-counter has reached zero for the first time after the writing of the new channel control word.

140489

(2/3)

Bit	Meaning and function	
	0	1
Bit 6 (D6)	Puts the channel in the timer mode. Puts the system clock into the prescaler and outputs the divided signal to the down-counter.	Puts the channel in the counter mode. The down-counter is decremented for each edge trigger applied to the CLK/TRG pin. In the counter mode, the prescaler is not used.
Bit 5 (D5)	Used only in the timer mode. The prescaler is set to divide the system clock by 16.	Used only in the timer mode. The prescaler is set to divide the system clock by 256.
Bit 4 (D4)	In the timer mode, the timer operation starts on the falling edge of the trigger PULSE (CLK/TRG) . In the counter mode, the down-counter is decremented on the falling edge of the external clock pulse (CLK/TRG)	In the timer mode, the timer operation starts on the rising edge of the trigger pulse (CLK/TRG). In the counter mode, the down-counter is decremented on the rising edge of the trigger pulse (CLK/TRG).
Bit 3 (D3)	Used only in the timer mode. The timer oparation starts on the rising edge following 2 system clocks after the time constant has been loaded into the down-counter.	Used only in the timer mode. The timer operation is started by the external trigger pulse. That is, the timer starts operating at the trigger pulse entered after the rising edge following 2 system clocks after the time constant has been loaded into the down-counter. If the interval between the system clock and the trigger pulse satisfies the setup time, the prescaler starts operating on the second rising edge. Otherwise, the prescaler starts operating on the rising edge following 3 system clocks. If the trigger pulse is entered before the time constant is loaded the same effect as when bit 3 = "0" takes place.
Bit 2 (D2)	Indicates that the channel control word is not followed by the writing of time constant. When the channel is in the reset state, this bit cannot be set to "0" in the first channel control word . To change states with the time constant unchanged, the channel control word with this bit set to "0" must be entered.	Indicates that the channel control word is followed by the writing of time constant. If the time constant is written during down-counter operation, the new time constant is set to the time constant register with proceeding the current counting. When the down-counter has reached zero, the new time constant is loaded into the down-counter.

140489

(3/3)

Bit	Meaning and function	
	0	1
Bit 1 (D1)	Continues the current channel operation	Stops the down-counter operation. When this bit is set to "1", the channel operation stops but all the channel control register bits remain unchanged. When bit 2 = "1" and bit 1 = "1", the channel operation remains stopped until a new time constant is written. Channel restart is set up after the new time constant is programmed. The channel is restarted acoording to the state of bit 3. When bit 2 = "0" and bit 1 = "1", the channel operation does not start until a new channel control word is written.

140489

[2] Time-Constant Data

In either the time mode or the counter mode, the time-constant data must be loaded into the time constant register. When bit 2 (D2) of the channel control word is "1", the time constant is loaded into the time constant register immediately after the channel control word is written. A time-constant value must be an integer in a range of 1 to 256. When the 8 bits of a time constant are all "0"s, such a time constant is assumed to be 256. Figure 3.4.11 shows the bit configuration of time-constant data.

D7	D6	D5	D4	D3	D2	D1	D0
TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0

140489

Figure 3.4.11 Time-Constant Data

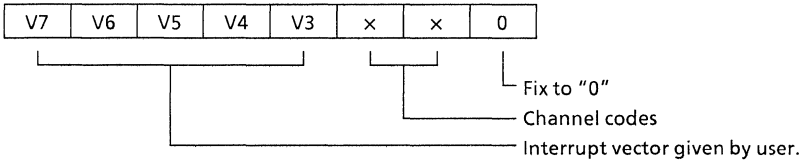
[3] Interrupt Vector

In interrupt in the MPU mode-2, the interrupting channel must give an interrupt vector to the MPU. An interrupt vector is written in the channel-0 interrupt vector register with bit 0 (D0) ="0". The vector is written in the same way as the channel control word is written on channel 0. However, bit 0 (D0) of the vector should always be "0". Bit 7 (D7) through bit 3 (D3) are user-defined values. Bit 2 (D2) and bit 1 (D1) are automatically given and contain the code of the interrupt-requesting channel having the highest priority. Table 3.4.3 shows the channel codes. Figure 3.4.12 shows the interrupt vector bit configuration.

Table 3.4.3 Channel Codes

Bit 2 (D2)	Bit 1 (D1)	Channel number	
0	0	0	(high)
0	1	1	↑ Priority
1	0	2	↓
1	1	3	(low)

140489



140489

Figure 3.4.12 Interrupt Vector

3.4.6 Using CTC

[1] Counter Mode

The following describes how to use the CTC by referring to a program using channel 0 with interrupt disabled.

(a) The counter programming procedure is shown in Figure 3.4.13.

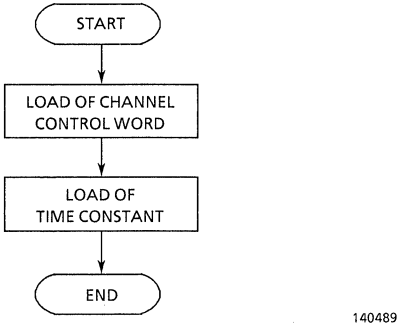


Figure 3.4.13 Counter Programming Procedure

(b) The block diagram for converting the 100 kHz system clock into the 10 kHz equivalent is shown in Figure 3.4.14.

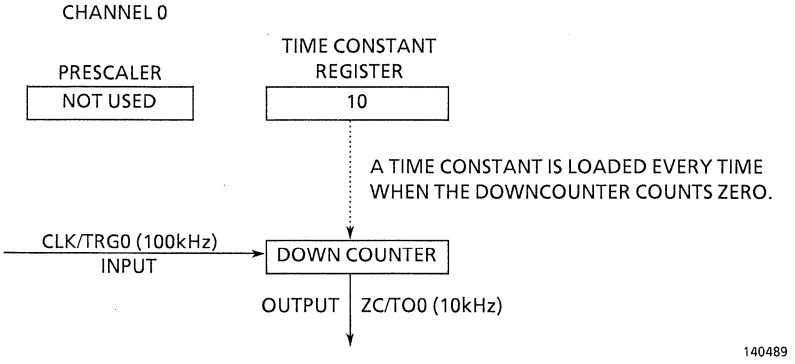


Figure 3.4.14 Down-Counter Block Diagram

(c) The channel control word configuration is shown in Figure 3.4.15.

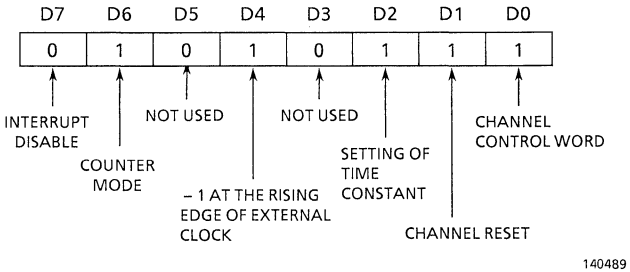


Figure 3.4.15 Channel Control Word Configuration

[2] Timer Mode

(a) The timer programming procedure with interrupt disabled is shown in Figure 3.4.16.

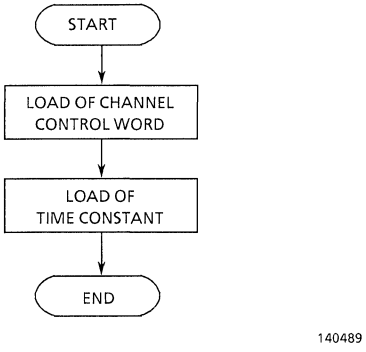


Figure 3.4.16 Timer Programming Procedure

(b) The block diagram for converting the 4 MHz system clock into the 1 kHz equivalent is shown in Figure 3.4.17.

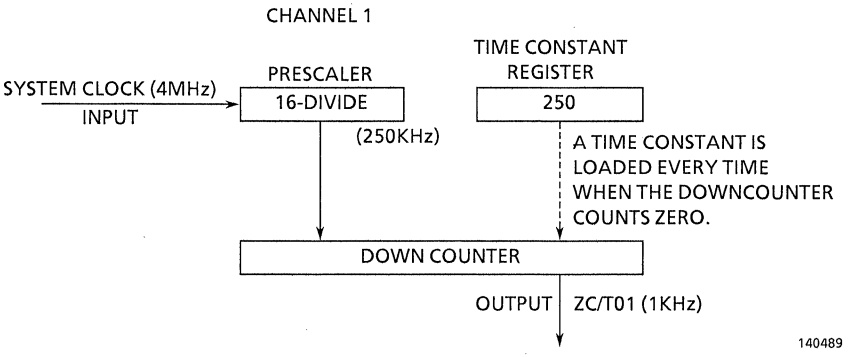


Figure 3.4.17 Timer block Diagram

(c) The channel control word is shown in Figure 3.4.18.

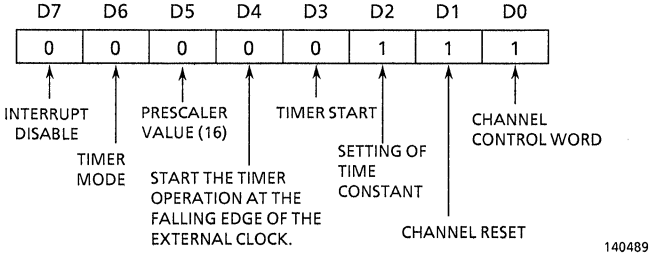


Figure 3.4.18 Channel Control Word

3.5 SIO OPERATIONAL DESCRIPTION

The SIO has two independent, programmable full-duplex serial ports. These ports are assigned addresses on the TMPZ84C013A's I/O map. This subsection mainly describes the operations that take place after accessing the SIO.

3.5.1 SIO Block Diagram

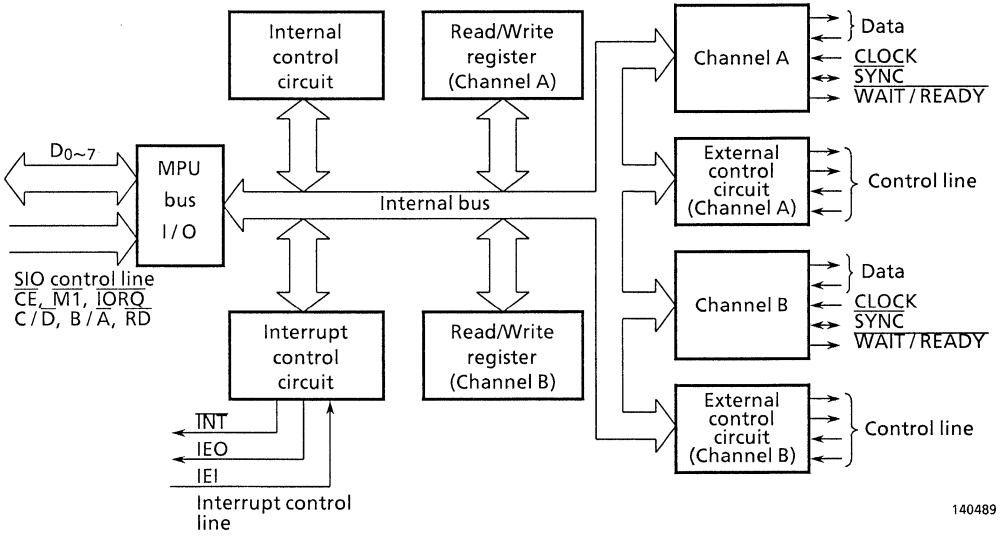
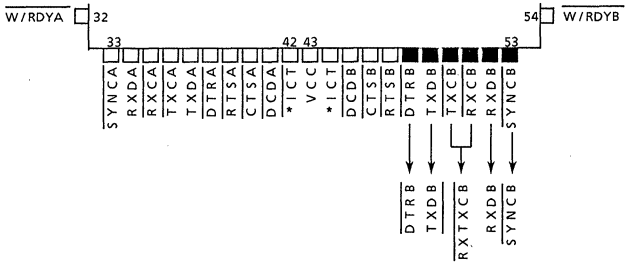


Figure 3.5.1 SIO Block Diagram

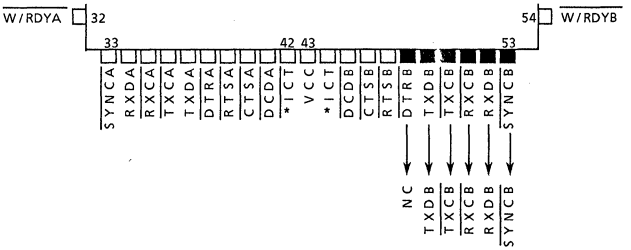
3.5.2 SIO System Configuration

As shown in Figure 3.5.1, the SIO consists of the MPU bus interface, the internal controller, the interrupt controller, and two independently operating full-duplex channels. Each channel has the read register, the write register, and the external controller which controls the connection with peripheral LSIs or external devices.

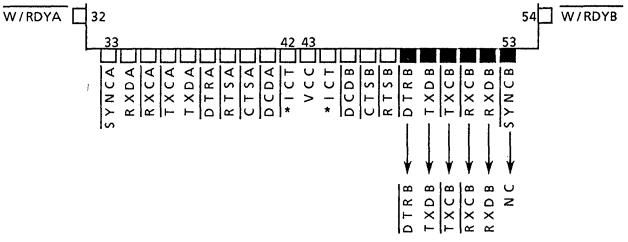
The TMPZ84C013A contains all the functions and pins of the 40-pin, DIP-type TMPZ84C40A (SIO/0), TMPZ84C41A (SIO/1), and TMPZ84C42A (SIO/2). However, when using the SIO of the TMPZ84C013A, the SIO/0, SIO/1, or SIO/2 must be used alone. The pin assignments are as shown in Figure 3.5.2.



(a) SIO/0



(b) SIO/1



(c) SIO/2

140489

Figure 3.5.2 SIO Pin Assignments

Table 3.5.1 shows the types and functions of the SIO registers. Each channel has 8 write registers and 3 read registers.

(1) Communication data path

Figure 3.5.3 shows the communication path of each channel's transfer data.

1 Receive operation

The receiver has an 8-bit receive register and a 3-stage 8-bit buffer register in FIFO configuration. This saves time in high-speed data block transfers. The receivers also have the receive error FIFO which holds the status information such as parity and framing errors. The receive data follow different paths according to the operation mode and character length as shown in Figure 3.5.3.

Table 3.5.1 (a) Write Registers

Register	Function
Write register 0 (WR0)	Resets CRC. Sets pointers of registers, and commands.
Write register 1 (WR1)	Sets the interrupt mode.
Write register 2 (WR2)	Sets the vector to be transmitted at interrupt. (Channel B only)
Write register 3 (WR3)	Provides the parameters to control the receiver.
Write register 4 (WR4)	Provides the parameters to control the receiver and transmitter.
Write register 5 (WR5)	Controls the transmitter.
Write register 6 (WR6)	Sets the sync character or the SDLC address field.
Write register 7 (WR7)	Sets the sync character or the SDLC flag.

140489

Table 3.5.1 (b) Read Registers

Register	Function
Read register 0 (RR0)	Indicates the receive/transmit buffer state and the pin state.
Read register 1 (RR1)	Indicates the error status and the end-of-frame code.
Read register 2 (RR2)	Indicates the interrupt vector contents. (Channel B only)

140489

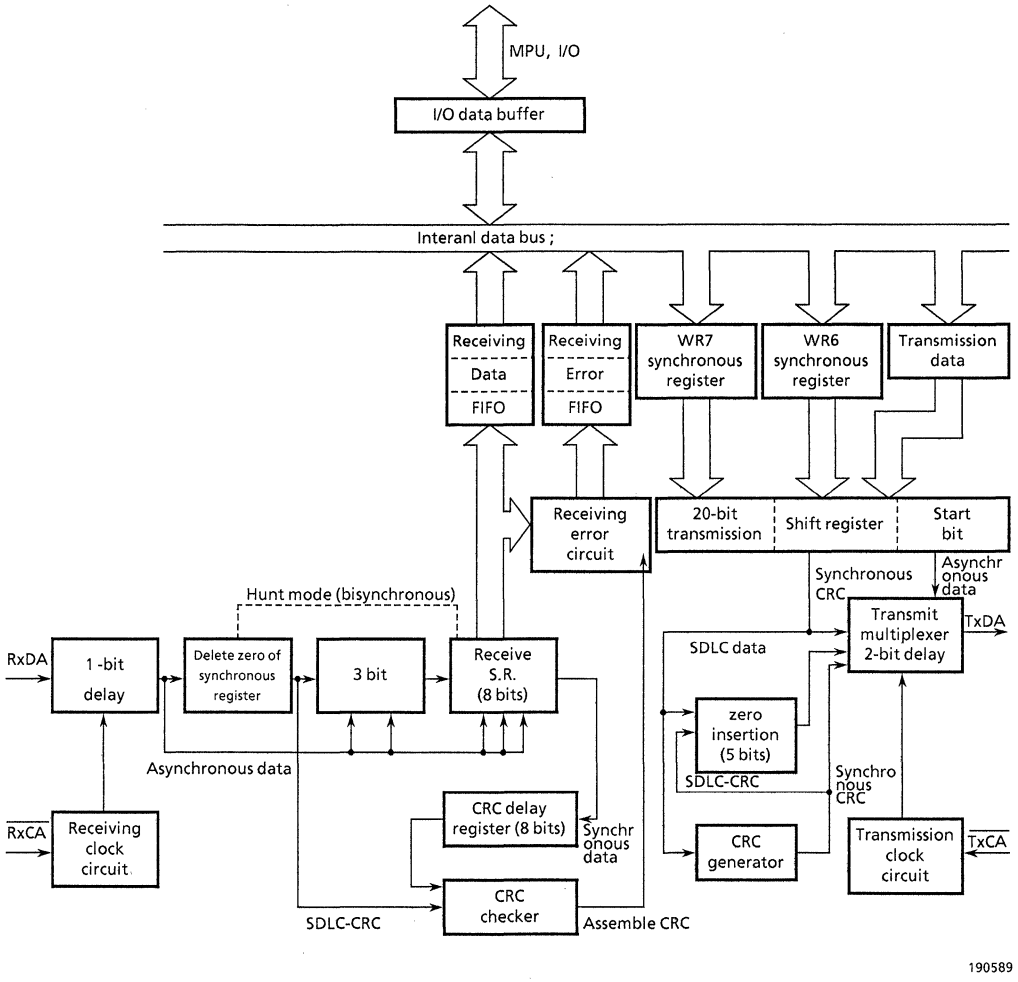


Figure 3.5.3 Transfer Data Path (Channel A)

- Asynchronous mode

  In the asynchronous mode, the receive data enters the 3-bit buffer if the character length is 7 or 8 bits or the 8-bit receive shift register if the character length is 5 or 6 bits.

- Synchronous mode

  In the synchronous mode, the data path depends on the receive processing phase at the time. The receiver operation starts from the hunt phase. In this mode, the receiver searches the receiver data for the bit pattern which matches the specified sync character. If the SIO is set in the monosync mode, the receiver searches for the bit pattern which matches the sync character set in WR7; if the SIO is set in the bisync mode, the receiver searches for the bit pattern which matches two consecutive sync characters set in WR6 and WR7. When synchronization has been established, the subsequent data enter the 3-bit buffer by bypassing the sync register.

- SDLC mode

  In the SDLC mode, the sync register constantly monitors the receive data perform-ing zero deletion as required. When the sync register detects 5 "1"s consecutively in the receive data, the following bit is deleted if it is "0". If it is "1", the bit that follows is checked. If it is "0", it is assumed as a flag, if it is "1", it is assumed an abort sequence (7 consecutive "1"s).

  The reformatted data are put in the receive shift register via the 3-bit buffer. When synchronization has been established, the subsequent data follow the same path regardless of the character length.

2 Transmission

The transmitter has an 8-bit transmit data register and a 20-bit transmit shift register. The 20-bit transmit shift register holds the data from the WR6, WR7, and transmit data register.

- Asynchronous mode

  In the asynchronous mode, the data in the 20-bit transmit shift register are added with the start and stop bits to be sent to the transmit multiplexer.

- Synchronous mode

  In the synchronous mode, the WR6 and WR7 hold the sync character. The contents of these registers are sent to the 20-bit transmit register as the sync character at the transmission of data blocks or as the idle sync character if a transmitter underrun occurs in data block transmission.

- SDLC mode

  In the SDLC mode, the WR6 holds the station address and the WR7 holds the flag. The flag (WR7) is sent to the 20-bit transmit register at the start and end of each frame. For each of the other data fields, one "0" follows five consecutive "1"s.

(2) I/O functions

To transfer data from/to the MPU, the SIO must be set in the polling, interrupt, or block transfer mode.

- Polling

  To operate the SIO in the polling mode, all interrupts mode must be disabled. In the polling mode, the MPU reads the status bits D0 and D2 in each channel's RR0 to check for reception or transmission.

- Interrupts

  There are 3 types of SIO interrupt: transmit interrupt, receive interrupt, and external/status interrupt. These interrupts can be enabled by program. The receive interrupt is further divided into the following three:

  - Interrupt on the first received character
  - Interrupt on all received characters
  - Interrupt on special receive conditions

  Higher priority is given to channel A than channel B. On the same channel, higher priority is given to reception, transmission, and external/status in this order.

  The SIO provides the daisy-chained interrupt priority control feature and the interrupt vector generating feature. Further, it provides the "status affected vector" feature. This feature outputs 4 interrupts depending on the interrupt source.

- Block transfer

  The SIO has the block transfer mode to adapt to the MPU"s block transfer and the DMA controller. For block transfer, the $\overline{\text{W/RDY}}$ line is used. For the MPU's block transfer, this line is used as the wait line; for the DMA block transfer, it is used as the ready line. The SIO's ready output indicates to the DMA controller that the data is ready to transfer. The SIO's wait output indicates to the MPU that the SIO is not ready for data transfer and therefore requesting the extension of the output cycle.

3.5.3 SIO Basic Operations

(1) Asynchronous mode

For data transfer in the asynchronous mode, the character length, clock rate, and interrupt mode must be set. These parameters are written in the write registers. Note that WR4 must be set before the other registers are set.

Data transfer does not start until the transmit enable bit is set. When the auto enable bit is set, the SIO starts transmission upon the $\overline{CTS}$ pin's going "0", allowing the programmer to send a message to the SIO without waiting for the $\overline{CTS}$ signal. Figure 3.5.4 shows the data format of the asynchronous mode.

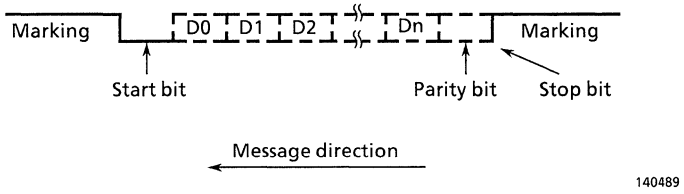


Figure 3.5.4 Data Format of Asynchoronous Mode

1 Transmission

Serial data are output from the TxD pin. Its transfer clock rate can be set to one of 1, 1/16, 1/32, and 1/64 times the clock rate to be supplied to the transmit clock input ($\overline{TxC}$). The serial data are output on the falling edge of $\overline{TxC}$.

2 Reception

The receive operation in the asynchronous mode starts when the receive enable bit (D0 of WR3) is set. When the receive data input RxD is set to "0" for the duration of at least 1/2 bit time, the SIO interprets it as the start bit, sampling the input data at the middle of the bit time. The sampling is performed on the rising edge of the $\overline{RxC}$ signal.

When the receiver receives the data whose character length is not 8 bits, it converts the data into the one composed of the necessary bits, the parity bit and the unused bit set to "1".

Example : a 6-bit character

"								"
1	P	D5	D4	D3	D2	D1	D0	

When the external/status interrupt is enabled and a break state is detected in the receive data, the interrupt is generated and the break/abort status bit (D7 of RR0) is set and the SIO monitors the transmit data until the break state is cleared. The interrupt is also generated when the $\overline{DCD}$ signal is in the inactive state for more than the specified pulse width. The DCD status bit is set to "1".

In the polling mode, the MPU must refer the receive character valid bit (D0 of RR0) to read the data. This bit is automatically reset when the receive buffer is read. In the polling mode, the transmit buffer status must be checked before writing data in the transmitter to avoid overwrite.

(2) Synchronous mode

There are 3 kinds of character synchronization : monosync, bisync, and external sync. In each of these synchronous modes, the times 1 clock rate is used for both transmission and reception. The receive data is sampled on the rising edge of the receive clock input ($\overline{RxC}$). The transmit data changes on the falling edge of the transmit clock input.

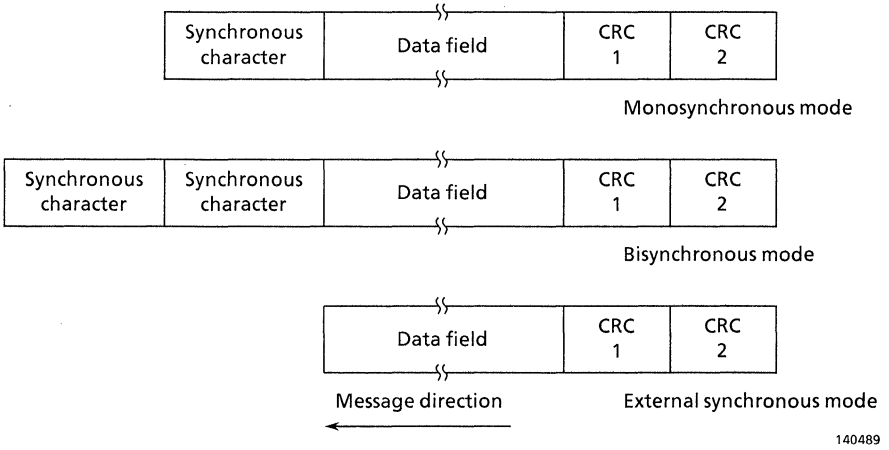


Figure 3.5.5 Data Format of Synchronous Mode

1 Monosync

In this mode, synchronization is established when a match with the sync character (8 bits) set to WR7 is found, enabling data transfer.

2 Bisync

In this mode, synchronization is established when a match with 2 consecutive sync characters set to WR6 and WR7 is found, enabling data transfer. In this mode as well as the monosync mode $\overline{SYNC}$ is active during the receive clock period in which the sync character is being detected.

3 External sync

In this mode, synchronization is performed externally. When synchronization is established, it is indicated by the $\overline{SYNC}$ pin. The $\overline{SYNC}$ input must be kept to "0" until the character synchronization is lost. Character assembly starts from the rising edge of the $\overline{RxC}$ after the falling of the $\overline{SYNC}$.

After reset, the SIO enters the hunt phase to search for the sync character. If synchronization is lost, the SIO sets the enter-hunt-phase-bit (D4 of WR3) to reenter the hunt phase.

- Transmission

(a) Data transfer using interrupt

When the transmit interrupt is enabled, the interrupt is caused upon the transmit buffer's being emptied. For the interrupt processing, other data are written in the transmitter. If these data are not ready for some reason, the transmit underrun condition occurs.

(b) Bisync mode

In the bisync mode, if the transmitter runs out of data during transmission, supply characters are inserted. This is done in two methods. In one method, sync characters are inserted. In the other, characters generated so far are transmitted followed by sync characters. Either of these methods can be selected by the reset transmit underrun/EOM command in WR0.

(c) End of transmission

Break can be performed by setting bit D4 of WR5. When break is performed, the data in the transmit buffer and the shift register are lost. When the external/status interrupt is enabled, the SIO generates the interrupt depending on the transmitter state and outputs the vector. This mode can be used for block transfer.

- Reception

(a) Interrupt on the first received character

This mode is used for ordinary block transfer. In this mode, the SIO generates the interrupt only for the first character;subsequently, it does not generate the interrupt unless special receive conditions are satisfied.

To initialize these settings, command 4 of WR0 (to be enabled by the next receive interrupt) must be set in advance.

(b) Interrupt on all received characters

In this mode, the SIO generates the interrupt for all characters coming into the receive buffer. When the status affect vector has been set, a special vector is generated on a special receive condition.

(c) Special receive condition interrupt

This interrupt occurs when any of the above interrupts is selected. The special receive conditions include parity error, receive overrun error, framing error, and end-of-frame (SDLC). These error status bits are latched, so that they must be reset after they are read. They can be reset by command 6 of WR0 (error reset).

(3) SDLC mode

The SIO supports both the SDLC and HDLC protocols. They resemble each other, so that only the SDLC mode is explained here.

Figure 3.5.6 shows the data format in the SDLC mode. In the SDLC mode, one data block is called a frame and the message in it is put between the open flag and the close flag. The address field in the frame contains the address of a secondary station. Checking this address, the SIO receives or ignores the frame.

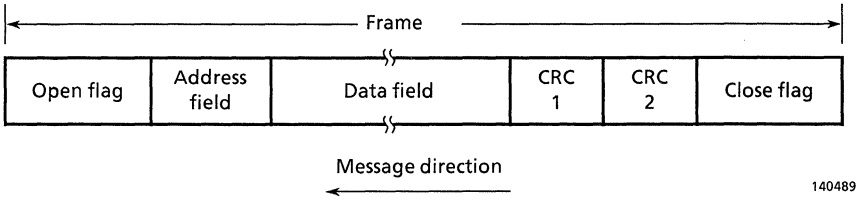


Figure 3.5.6 Data Format in SDLC Mode

- Transmission

(a) Data transfer using interrupt

When the transmit interrupt has been set, the interrupt occurs each time the transmit buffer becomes empty. In the SDLC mode, data are sent to the SIO by this interrupt.

(b) Data transfer using wait/ready

The wait function in the wait/ready capability is used to make the MPU extend the output cycle when the SIO's transmit buffer is not empty. The ready function indicates to the DMA that the SIO's transmit buffer is empty and therefore ready to receive data. If no data has been written in the transmit shift register before transmission, the SIO goes in the underrun state. This capability permits data transfer to the SIO.

(c) Transmit underrun/EOM

The SIO automatically ends the SDLC frame if there is no data to be transmitted to the transmit data buffer. To implement this, the SIO sends a 2-byte CRC when there is no data to send, then the SIO transmits one or more flags. After reset, the transmit underrun/EOM status bit is set to prevent the CRC character from being inserted when there is no data to be sent. Using this function, the SIO starts frame transmission. Here, the transmit underrun/EOM reset command must be set in advance between the transmission of the first data and the data end. Thus, the SIO goes in the reset state at the end of each message with the CRC character being sent automatically.

(d) CRC generation

For CRC calculation, the CRC generator must be reset before transmission (bits D6 and D7 of WR0). CRC calculation starts when the address field is written in the SIO (WR6). The transmit CRC enable bit (D0 of WR5) must be set before the address field is written.

(e) End of transmission

When the transmitter is disabled during transmission, the data currently transmitted is all transmitted to its end. The subsequent data is put in the marking state. When the transmitter is disabled, characters remain in the buffer. However, the abort sequence is made active when the abort command is written in the command register, deleting all data.

- Reception

As in the transmit mode, several parameters must be preset in the receive mode. The address field is written in WR7 and the flag character in WR7. Receiving the open flag, the receiver compares the contents of the following address field with the address set in WR6 or the global address ("1111 1111"). If the contents of the address field in frame matches either of these address, the SIO starts reception.

(a) Interrupt on the first received character

This mode is generally used for the block transfer using the wait/ready capability. In this mode, the SIO generates the interrupt only on the first character. The status flag of this interrupt is latched, so that command 4 (to be enabled by the next received character) of WR0 must be preset for re-initialization. When the external/status interrupt is set, an interrupt occurs every time the $\overline{DCD}$ changes. This interrupt also occurs when the special receive condition is satisfied.

(b) Interrupt on all received characters

In this mode, the SIO generates an interrupt on all received characters. When the status affect vector has been set, the SIO generates a special vector on the special receive condition interrupt.

(c) Special receive condition interrupt

Using the special receive condition, the interrupt on the first received character or the interrupt on all received characters must be selected in advance. The receive overrun status of the special receive condition interrupt is latched. The status bit can be reset by the error reset command (WR0 command).

(d) CRC check

The receive CRC check is reset when the open flag at the head of a frame is received. CRC calculation is performed on the subsequent characters up to the close flag. In the SDLC mode, the transmit CRC is inverted, so that a special check sequence is used. The check must end with "0001 1101 0000 1111." Since SIO handles the CRC character as a data, the MPU must discard it after reading it.

(e) End of transmission

When the SIO receives the close flag, the end-of-frame-bit is set to indicate that the close flag has been received. When the status affect vector has been set, the special receive condition interrupt occurs and the interrupt vector is output. Any frame can be aborted by abort transmission. When the external/status interrupt has been set, the interrupt occurs and the break/abort bit in RR0 is set.

3.5.4 SIO Status Transition Diagram and Basic Timing

[1] Status Transition Diagram

Figure 3.5.7 shows the SIO status transition diagram.

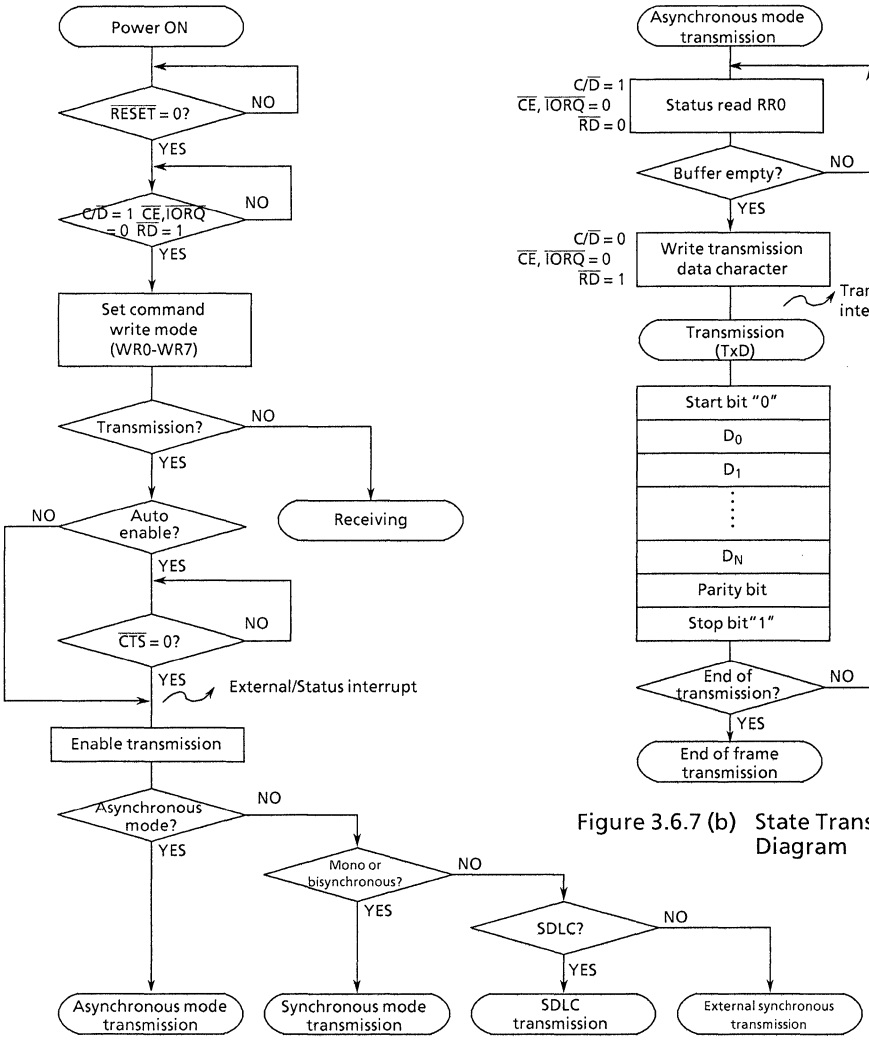


Figure 3.6.7 (b) State Transition Diagram

Figure 3.5.7 (a) SIO Status Transition Diagram

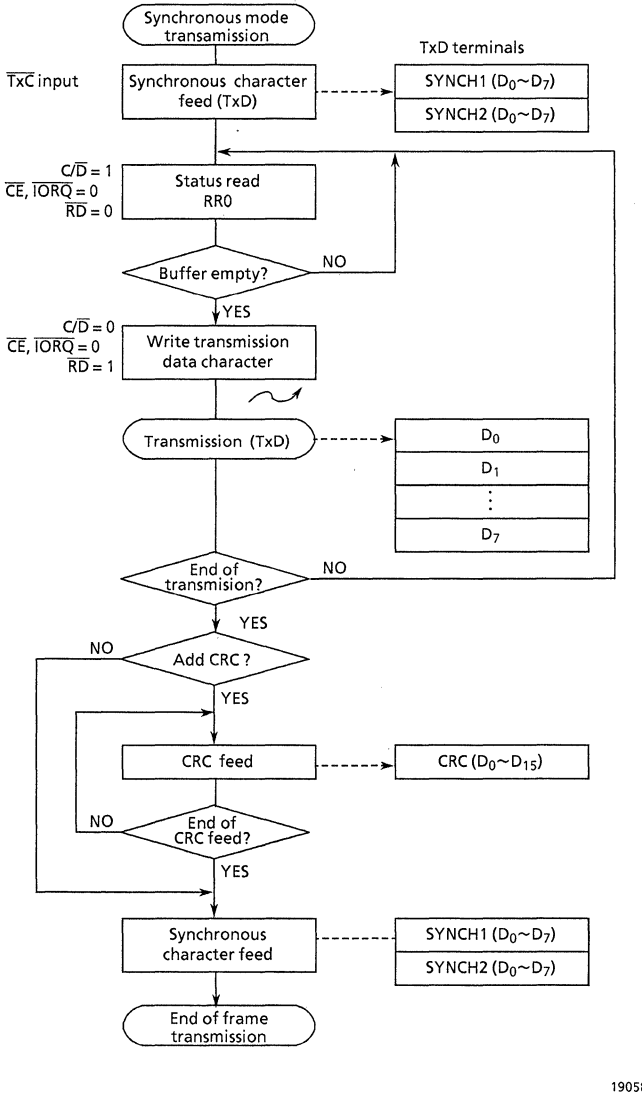


Figure 3.6.7 (c) State Transition Diagram

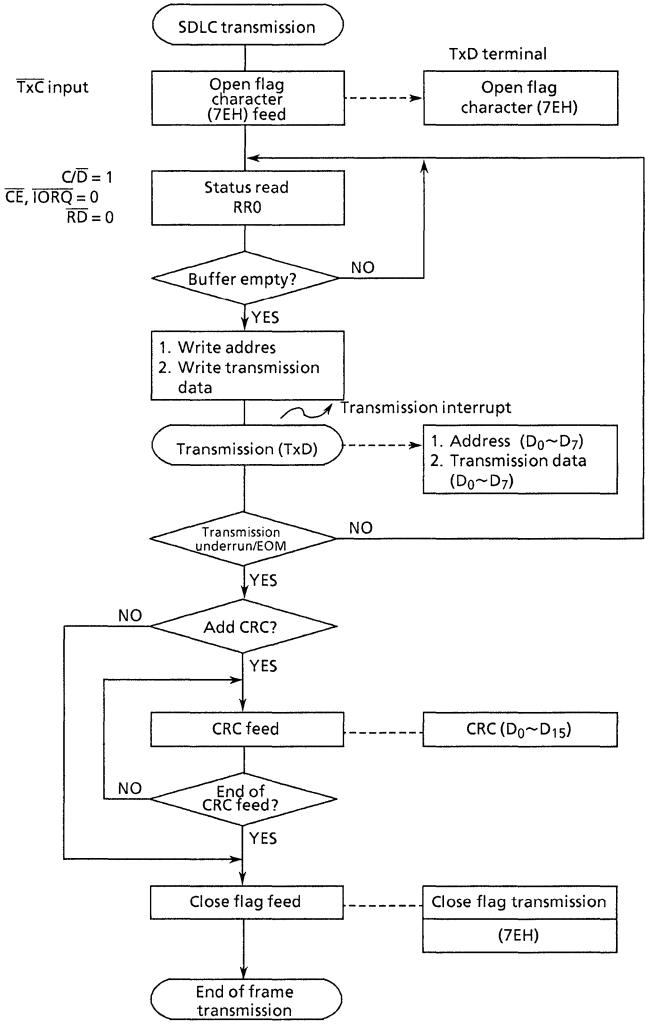


Figure 3.6.7 (d) SIO Status Transition Diagram

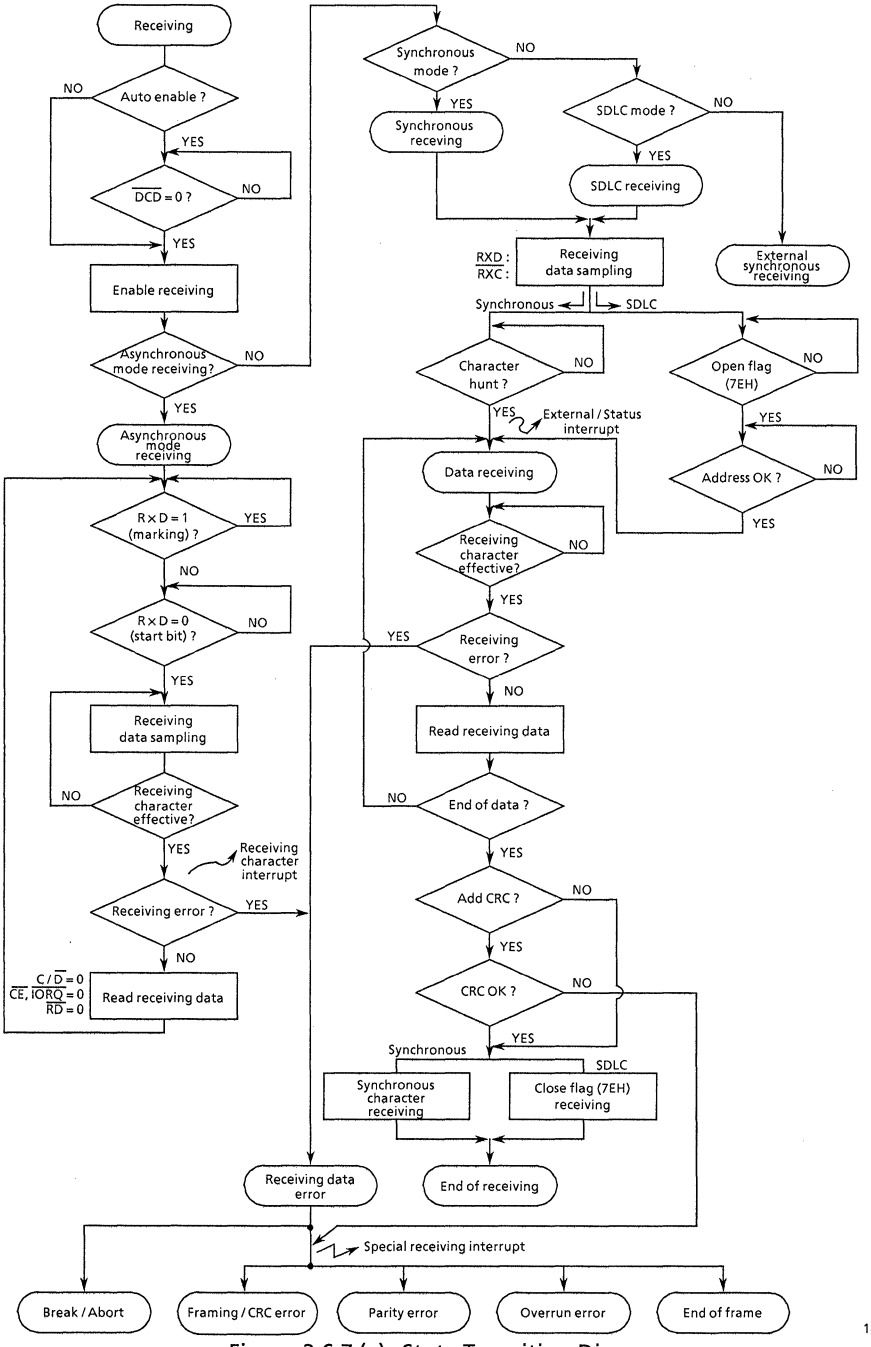


Figure 3.6.7 (e) State Transition Diagram

[2] Basic Timing

Figure 3.5.8 shows the timing in which data or a command is written from the MPU to the SIO. Figure 3.5.9 shows the timing in which data is read from the SIO to the MPU. Figure 3.5.10 shows the interrupt acknowledge timing in which the MPU gives an interrupt response to the SIO's interrupt request to set the $\overline{IORQ}$ pin to "0" several clocks after setting the $\overline{M1}$ pin to "0" as the acknowledge signal. To maintain the interrupt serviced state in daisy chain structure, the interrupt request state cannot be changed while $\overline{M1}$ is active.

Figure 3.5.11 shows the timing in which the return from interrupt is performed.

Figure 3.5.12 shows how the daisy chain structure works. First, suppose that the SIO is servicing interrupt. When the PIO issues an interrupt request immediately before the first byte "EDH" of the RETI instruction is decoded with $\overline{M1}$ being active, "IEO" of the PIO goes "0". However, when "EDH" is decoded, the PIO's interrupt request is not acknowledged. Therefore, the PIO's "IEO" returns to "1". When the second byte "4DH" is decoded, the SIO's "IEO" returns to "1". Therefore, the "IEI" and "IEO" of each peripheral LSI at this point of time all go "1", or out of the interrupt serviced state. The PIO keeps the $\overline{INT}$ pin at "0" until this state is set. Then, the interrupt is serviced starting with the peripheral device of the higher priority.

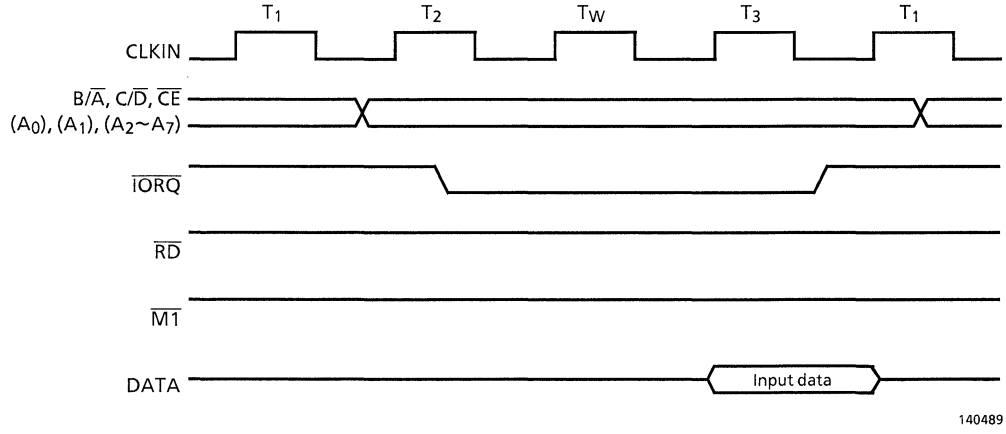


Figure 3.5.8 Write Timing

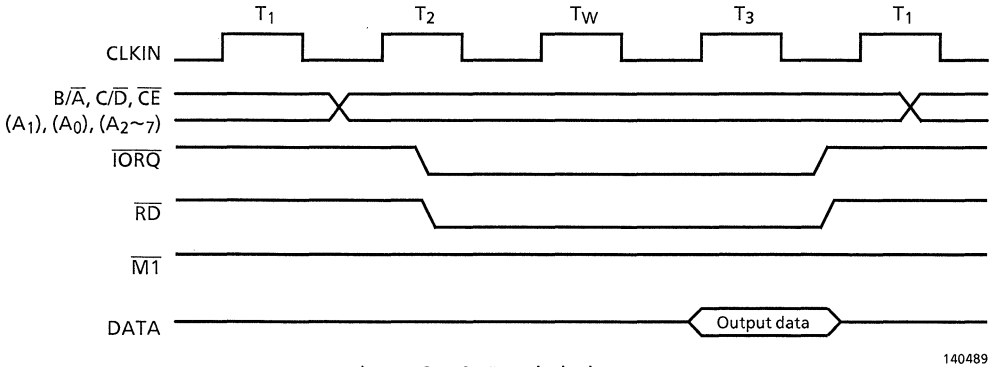


Figure 3.5.9 Read Timing

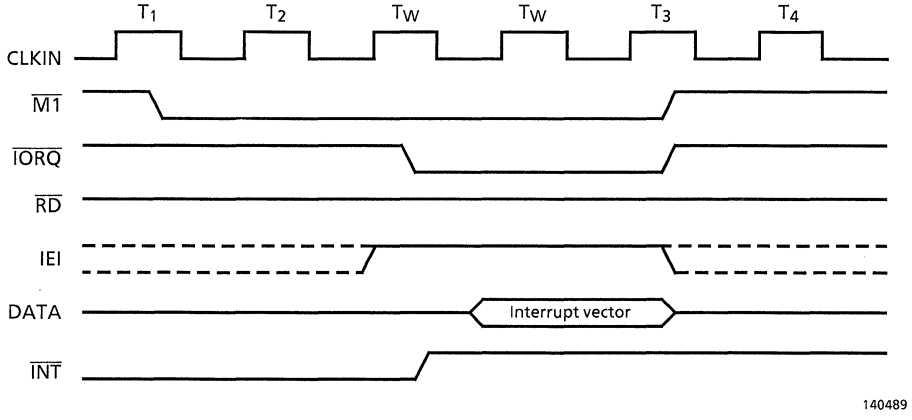


Figure 3.5.10 Interrupt Acknowledge Timing

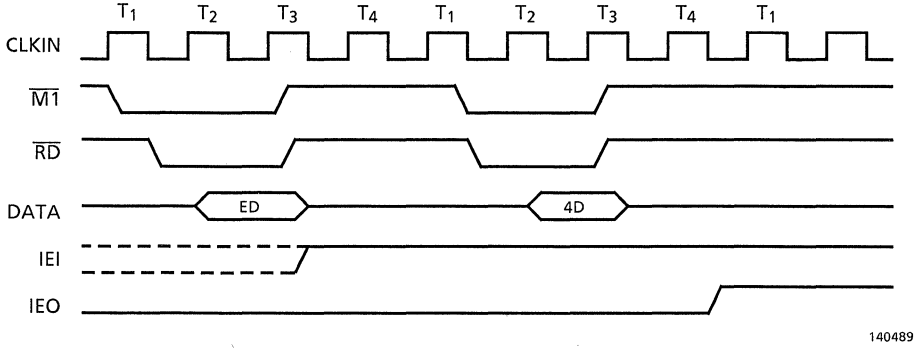
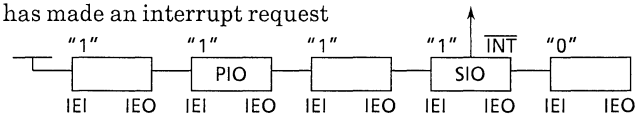
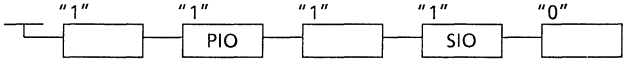


Figure 3.5.11 Return Timing from Interrupt

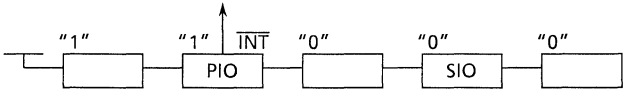
1 The SIO has made an interrupt request



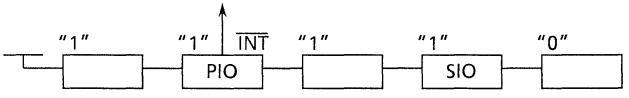
2 The SIO is servicing the interrupt.



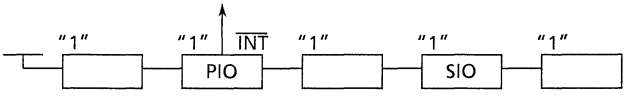
3 The PIO has made an interrupt request immediately before "EDH" is decoded by the SIO. By the PIO's interrupt request, PIO's IEO is set to "0".



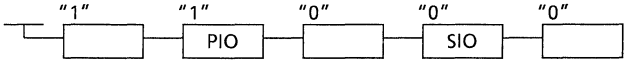
4 Because "EDH" has been decoded, the PIO's interrupt request is not acknowledged. Therefore, PIO's IEO returns to "1".



5 Because "4DH" has been decoded, the SIO's IEO is set to "1".



6 The PIO's interrupt request is acknowledged and the PIO's IEO is set to "0".



140489

Figure 3.5.12 Daisy Chain at Execution of RETI Instruction

3.5.5 SIO Operational Procedure

The following mainly describes the meaning of each bit of the write and read registers. Special attention should be directed to the fact that the parameters of the write register (WR4) should be set before the others.

Some registers can use only a signal channel. The I/O addresses listed in Table 3.5.2 must be specified to write the control word and read/write data on the SIO.

Table 3.5.2 I/O Addresses

I/O function	I/O address
Channel A data	#18
Channel A command	#19
Channel B data	#1A
Channel B command	#1B

140489

[1] Write Registers

1 WR 0; Write register 0

Table 3.5.3 Configuration of Write Register 0

D7	D6	D5	D4	D3	D2	D1	D0
CRC reset code		Primary command bit			Register pointer bit		

140489

Bits D0 through D2: Register pointer bits

These bits specify the register on which read/write is performed by the next byte. When read/write is completed, the register pointer points to WR0.

Bits D3 through D5: Basic command bits

- Command 0 (=000): No operation

  This command only sets the register pointer without making the SIO operate. It is used to invalidate the command in the command chain for the SIO or hold the location at which a command is inserted in the command chain if required.

- Command 1 (=001): Abort sequence generation

  This command is used to generate the abort sequence (7 or more consecutive "1"s). Note that command 1 is used only in the SDLC.

- Command 2 (=010): External/status interrupt reset

  Once an external interrupt or a status interrupt has occurred, the status bit of RRO is latched. This command is issued to enable the RR0's status bit in order to enable the interrupt again.

- Command 3 (=011): Channel reset

  This command performs generally the same operation as when the $\overline{\text{RESET}}$ pin is set. The difference is that reset is performed only on a single channel. The command for channel A resets the interrupt priority circuit as well.

- Command 4 (=100): Enable the interrupt at the next character reception.

  This command is used to enable an interrupt when the end of data block has been detected followed by the reception of the next block.

- Command 5 (=101): Reset transmit interrupt pending

  If the transmit buffer becomes empty in the transmit interrupt enable mode, an interrupt occurs. This command is used to disable the transmit interrupt when there is no data in the transmit buffer.

- Command 6 (=110): Error reset

  The error (parity or overrun error) caused in block transfer is latched in bits D4 and D5 of RR1. This commands is used to clear these bits.

- Command 7 (=111): Return from interrupt

  This command performs the same operation as the operation required to execute the RETI instruction on the SIO's data bus. Therefore, non-Z80 MPUs (that is, systems using no RETI instruction) can use the daisy chain in the SIO. This command is available only on channel A.

Bits D6 and D7: CRC reset code

These 2 bits allow the programmer to select between the receive CRC checker reset, the transmit CRC generator reset, and the transmit underrun/EOM reset.

Table 3.5.4 List of Reset Command Codes

Rest command	D7	D6
No operation	0	0
Reset the receive CRC checker	0	1
Reset the transmit CRC generator	1	0
Reset the transmit underrun/EOM	1	1

140489

2 WR 1; Write register 1

Table 3.5.5 Configuration of Write Register 1

D7	D6	D5	D4	D3	D2	D1	D0
Enable	Wait/ready Select function	Select receiving/ transmission	Receiving interrupt mode		Status -affect vector	Enable transmission interrupt	Enable external/ status interrupt

140489

Bit D0: External/status interrupt enable

When this bit is set, an interrupt is generated at the start of sync character transmission even if the execution is terminated upon detection of break/abort, the $\overline{DCD}$, $\overline{CTS}$ or $\overline{SYNC}$ signal has changed, or the transmit underrun/EOM latch is set.

Bit D1: Transmit interrupt enable

When this bit is set, a transmit interrupt is generated upon the transmit buffer becoming empty.

Bit D2: Status affect vector

When this bit is set, bits D1 through D3 (V1 through V3) of WR2 is changed. When this bit is not set, the same interrupt vector as the contents of WR2 issued. Note that this bit is available only on channel B.

Bits D3 and D4: Receive interrupt mode

These bits are used to select a receive interrupt mode.

Bits D5 through D7: Selection wait/ready functions

These 3 bits are used to select a $\overline{W/RDY}$ pin function. The wait or the ready function is selected by program and they are not used simultaneously. The meaning of these bits are:

- When D5 is set to "1", it indicates that the $\overline{W/RDY}$ pin responds to the receive buffer; when D5 is reset to "0", it indicates that the pin responds to the transmit buffer.
- When D6 is set to "1", the $\overline{W/RDY}$ pin functions as the $\overline{READY}$ pin; when D6 is reset to "0", the pin functions as the $\overline{WAIT}$ pin.
- When D7 is set to "1", the wait/ready function is enabled; when D7 is reset to "0", the function is disabled.

For example, when D7, D6, and D5 are "1", "1", and "0" respectively, and the transmit buffer is full, the $\overline{\text{READY}}$ pin goes "1"; when the transmit buffer is empty, the pin goes "0".

Table 3.5.6 shows the summary of the above description of bits D3 and D4 and D5 through D7.

Table 3.5.6 List of Receive Interrupt Mode Codes

Receive interrupt mode	D4	D3
Receive interrupt disable	0	0
Interrupt on first received character or special receive condition*	0	1
Interrupt on received character or special receive condition*	1	0
Interrupt on received character or special receive condition* (except for parity error)	1	1

140489

* Special receive conditions:
- End of frame (in SDLC mode only)
- Receive overrun error
- Parity error
- Framing error

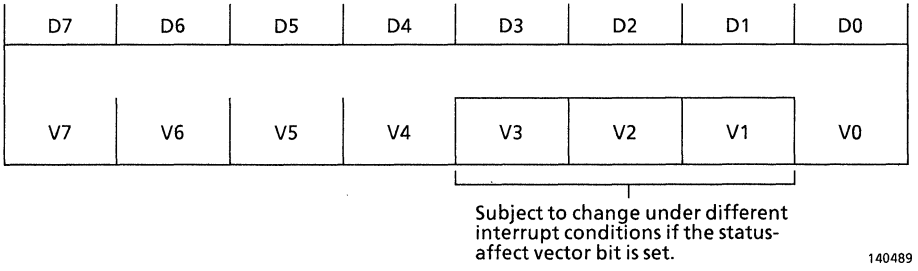
Table 3.5.7 Wait/Ready Select Function (D5 through D7)

Pin state			Buffer state	D7	D6	D5
Pin (Function)		Pin output				
DISABLE	$\overline{\text{WAIT}}$	Floating	–	0	0	–
	$\overline{\text{READY}}$	High	–		1	
ENABLE	$\overline{\text{WAIT}}$	Low	The transmit buffer is full and the SIO data port is selected.	1	0	0
		Floating	The transmit buffer is empty.			
	$\overline{\text{READY}}$	High	The transmit buffer is full.		1	
		Low	The transmit buffer is empty.			
	$\overline{\text{WAIT}}$	Floating	The receive buffer is full.		0	1
		Low	The receive buffer is empty and the SIO data port is selected.			
	$\overline{\text{READY}}$	Low	The receive buffer is full.		1	
		High	The receive buffer is empty.			

140489

3 WR 2; Write register 2

Table 3.5.8 Configuration of Write Register 2



This write register is the interrupt vector register. When bit D2 of WR1 (B channel) is not set, the interrupt vector is issued. When bit D2 of WR1 (B channel) is set, bits D1 through D3 (V1 through V3) are changed depending on the interrupt generation condition. This time, the contents of WR2 remain unchanged. Because WR2 is available only on channel B, WR2 must be programmed even if only channel A of the SIO is used.

Table 3.5.9 shows the WR2 bit states in the interrupt condition with the status affect vector being set.

Table 3.5.9 Channel Interrupt Condition Codes

Channel	Interrupt condition	V3	V2	V1
B	Transmit buffer empty	0	0	0
	Change of external/status	0	0	1
	Received character available	0	1	0
	Special receive condition*	0	1	1
A	Transmit buffer empty	1	0	0
	Change of external/status	1	0	1
	Received character available	1	1	0
	Special receive condition*	1	1	1

140489

* Special receive conditions:
- End of frame (in SDLC mode only)
- Receive overrun error
- Parity error
- Framing error

4 WR 3; Write register 3

Table 3.5.10 Configuration of Write Register 3

D7	D6	D5	D4	D3	D2	D1	D0
Receiving bit /character		Auto enable	Enter hunt phase	Enable receiving CRC	Address search mode	Prohibit synchronous character load	Enable receiving

140489

Bit D0: Receive enable

When this bit is set, the receive operation starts. Because this bit is used to start the receive operation, it must be set after the receive-associated programming has been all completed.

Bit D1: Sync character load inhibit

When this bit is set in the sync mode, the sync character is not loaded into the receive buffer. This bit is used to remove the sync character and idle sync from the received characters.

Bit D2: Address search mode

When this bit is set in the SDLC mode, any message having a programmed address or an address other than the global address (FFH) is not received by WR6. Therefore, the receive interrupt does not occur unless an address match occurs.

Bit D3: Receive CRC enable

When this bit is set, CRC calculation starts at the start of the last data transfer from the receive shift register to the receiver buffer.

Bit D4: Enter hunt Phase

When the establishment of synchronization is required, set this bit to enter the SIO into the hunt phase. The hunt phase is automatically cleared upon establishment of synchronization.

Bit D5: Auto enable

When this bit is set, the transmitter is enabled at the time the $\overline{CTS}$ pin is "0". When the $\overline{DCD}$ pin is "0", the receiver is enabled.

Bits D6 and D7: Receive character length

These bits are used to specify the number of receive bits which make up one character (character length). Table 3.5.11 shows the number of bits per character.

Table 3.5.11 Receive Character Length Codes

Bits/character	D7	D6
5	0	0
7	0	1
6	1	0
8	1	1

140489

5 WR 4; Write register 4

Table 3.5.12 Configuration of Write Register 4

D7	D6	D5	D4	D3	D2	D1	D0
Clock mode		Synchronous mode		Stop bit		Parity Even/$\overline{\text{Odd}}$	Enable

140489

Bit D0: Parity enable

When this bit is set, 1-bit transmit data is added to the number of bits specified by D6 and D7 of WR3 and the data is received in the resulting number of bits. If a character length other than 8 bits is selected, the added parity bit is set to the MSB side to be transferred to the receive data FIFO. When the 8-bit character length is selected, the parity bit is not transferred to the receive data FIFO.

Bit D1: Parity even/odd

This bit is used to determine whether to perform transfer and check in even or odd parity. (Even parity = "1", odd parity = "0")

Bit D2 and D3: Stop bit length

These bits are used to select the stop bit length in the asynchronous mode. In the synchronous mode, both D2 and D3 must be set to "0".

Table 3.5.13 Stop Bit Length Codes

top bit	D3	D2
Sync mode	0	0
1 stop bit/character	0	1
1.5 stop bits/character	1	0
2 stop bits/character	1	1

140489

Bits D4 and D5: Sync mode

These bits are used to select the sync mode.

Table 3.5.14 Sync Mode Codes

Sync mode	D5	D4
8-bit sync mode	0	0
16-bit sync mode (bisync mode)	0	1
SDLC mode (flag character; 7EH)	1	0
External sync mode	1	1

140489

Bits D6 and D7: Clock mode

These bits are used to select the factor between the transmit/receive clock and the data transfer rate. In the synchronous mode, the $\times 1$ clock mode must be set. In the asynchronous mode, the transmit side and the receive side must have the same factor.

Table 3.5.15 Clock Mode Codes

Clock mode (data transfer rate)	D7	D6
× 1 data transfer rate	0	0
× 16 data transfer rate	0	1
× 32 data transfer rate	1	0
× 64 data transfer rate	1	1

140489

6 WR 5; Write register 5

Table 3.5.16 Configuration of Write Register 5

D7	D6	D5	D4	D3	D2	D1	D0
DTR	Transmit bit /character		Break transmission	Enable transmission	CRC-16 /SDLC	RTS	Enable CRC transmission

140489

Bit D0: Transmit CRC enable

When this bit is set at the time the transmit data is loaded from the transmit data buffer into the transmit shift register, the CRC calculation is performed on that data. If this bit is not set, the CRC calculation and transmission are not performed in the transmit underrun state in the synchronous or SDLC mode.

Bit D1: Request to send

When this bit is set, the $\overline{RTS}$ pin goes "0". When this bit is not set, the $\overline{RTS}$ pin goes "1". In the asynchronous mode, the $\overline{RTS}$ pin goes "1" when the transmit buffer becomes empty. In the synchronous or SDLC mode, this bit state is followed by the $\overline{RTS}$ pin state.

Bit D2: CRC-16 / SDLC

When this bit is set, the CRC-16 polynomial ($X^{16}+X^{15}+X^2+1$) is selected. When this bit is reset to "0", the CRC-CCITT polynomial ($X^{16}+X^{12}+X^5+1$) is selected.

Bit D3: Transmit enable

When this bit is set, the transmitter is enabled. Even if this bit is reset to "0" after the start of transmission, the sync character and the data being transmitted are transmitted to the last.

Bit D4: Transmit break

When this bit is set, transmitting any data forcibly puts the transmit data line (TxD pin) in the space state. When this bit is reset to "0", the TxD pin is put in the marking state.

Bits D5 and D6: Transmit character length

These bits indicate the character length of transmit data.

Table 3.5.17 Transmit Character Length Codes

Bits/character	D6	D5
Less than 5 bits	0	0
7 bits	0	1
6 bits	1	0
8 bits	1	1

140489

As shown in Table 3.5.17, for the transmission of less than 5 bits (4 bits or 3 bits) per character, D6 and D5 are "0" and "0", which do not indicate how many bits the transmit data consists of. To solve this problem, the data characters must be processed by the format shown in Table 3.5.18. Note that D indicates data.

Table 3.5.18 Data Transfer Format with Transmit Data Consisting of Less than 5 bits

Transmit bits/character	D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	0	0	0	D
2	1	1	1	0	0	0	D	D
3	1	1	0	0	0	D	D	D
4	1	0	0	0	D	D	D	D
5	0	0	0	D	D	D	D	D

140489

Bit D7: Data terminal ready

This bit indicates the $\overline{DTR}$ pin state. When this bit is set, the $\overline{DTR}$ pin goes "0", when it is reset, the $\overline{DTR}$ pin goes "1".

7 WR 6; Write register 6

Table 3.5.19 Configuration of Write Register 6

D7	D6	D5	D4	D3	D2	D1	D0
SYNC							
7	6	5	4	3	2	1	0

140489

This register is programmed as follows:

- In the external sync mode : Transmit sync character
- In the monosync mode : Transmit sync character
- In the bisync mode : First sync character
- In the SDLC mode : Slave station address

8 WR 7; Write register 7

Table 3.5.20 Configuration of Write Register 7

D7	D6	D5	D4	D3	D2	D1	D0
SYNC							
15 (7)	14 (6)	13 (5)	12 (4)	11 (3)	10 (2)	9 (1)	8 (0)

140489

This register is programmed as follows:

- In the monosync mode : Receive sync character
- In the bisync mode : Second sync character
- In the SDLC mode : Flag character (7EH)

This register is not used in the external sync mode.

[2] Read Registers

1 RR 0; Read register 0

Table 3.5.21 Configuration of Read Register 0

D7	D6	D5	D4	D3	D2	D1	D0
Break/ Abort	Trasmission underrun/ EOM	CTS	Synchronize /Hunt	DCD	Trasmission buffer empty	Interrupt pending	Receiving character effective

140489

Used with the external/status interrupt

Bit D0: Receive character available

This bit is set when the receive buffer holds characters of 1 byte or more. This bit is reset when the buffer becomes empty.

Bit D1: Interrupt pending

This bit is set when an interrupt occurs in the SIO regardless of the interrupt condition type. This bit is available only on channel A.

Bit D2: Transmit buffer empty

This bit is set when the transmit data buffer becomes empty or the SIO is reset. However, in the sync and SDLC modes where the CRC character is being transmitted, bit D2 is reset.

Bit D3: Data carrier detect

This bit indicates the $\overline{DCD}$ pin input state. This bit is latched when the external/status interrupt occurs.

Bit D4: Sync/hunt

The meaning of this bit depends on the operation mode:

(i) Asynchronous mode

Bit D4 indicates the SIO's $\overline{SYNC}$ pin state. When the $\overline{SYNC}$ pin state changes, the external/status interrupt occurs.

(ii) External sync mode

When synchronization has been established by the detection of external synchronization, the last bit of the sync character must be set to "0" at the second $\overline{RxC}$ falling edge from the rising edge of the received $\overline{RxC}$. That is, to set the $\overline{SYNC}$ input to "0" by the external circuit after the detection of synchronization, full 2 receive cycle clocks must be awaited.

When the $\overline{SYNC}$ input goes "0", the sync hunt bit is set. When synchronization is lost or the end of message is detected, the enter hunt phase bit is set.

(iii) Internal sync mode

In the monosync and bisync modes, bit D4 is initialized to "1" by the enter hunt phase command (D4 of WR3). This bit is reset when the SIO detects the sync character.

(iv) SDLC mode

Bit D4 is set when the receiver is disabled or the enter hunt phase command is issued. Then, when the frame open flag is detected, this bit is reset.

Bit D5: Clear to send

This bit indicates the opposite of the $\overline{CTS}$ pin input state.

Bit D6: Transmit underrun/EOM

This bit is set when the SIO is reset (including channel reset). Only the reset transmitter underrun/EOM latch command (WR0 bits D7, D6="1", "1") can reset this bit. When the transmit underrun state occurs, the external/status interrupt is generated. Bit D5 is also used to control transmission in the sync or SDLC mode.

Bit D7: Break/abort

In the asynchronous mode in reception, this bit indicates the break state detection. When the break state is detected, this bit is set, generating the external/status interrupt. This bit is reset by the external/status interrupt reset command.

After break, the external/status interrupt is generated again. In the SDLC mode, bit D7 is set when the abort sequence is detected, generating the external/status interrupt.

2 RR 1; Read register 1

Table 3.5.22 Configuration of Read Register 1

D7	D6	D5	D4	D3	D2	D1	D0
End of frame	framing error	Receiving overrun error	Parity error	Fraction			Feed all characters

140489

Bit D0: All sent

In the asynchronous mode, this bit is set when all characters are sent from the transmitter or there is no transmit data in the SIO. In the synchronous mode, this bit is always set.

Bits D1 through D3: Fraction codes

Normally, I field is an integral multiple of character length. If it is not, these bits show the number of fraction bits. These codes are effective only for the transmission for which the end of frame bit is set in the SDLC mode.

Example : Figure 3.5.13 shows examples of fractions in which the number of bits/character at the end of I field is 8 bits (1) and 4 bits (2).

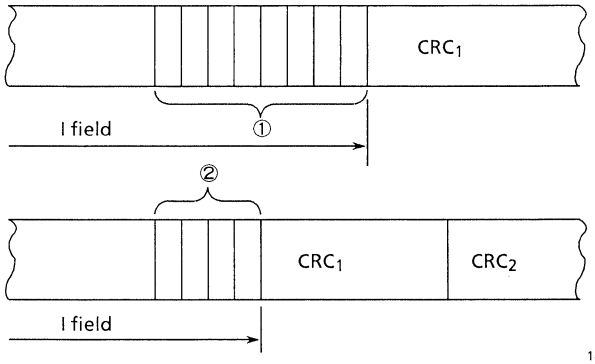


Figure 3.5.13 Examples of Fraction Bits in I Field

Table 3.5.23 (a) shows the fraction codes for the receive character whose character length is 8 bits.

Table 3.5.23 (a) Bit Patterns by Fraction Bits at End of I Field

Number of fraction bits at end of I field		D3	D2	D1
1 byte before	2 bytes before			
0	3	1	0	0
0	4	0	1	0
0	5	1	1	0
0	6	0	0	1
0	7	1	0	1
0	8	0	1	1
1	8	1	1	1
2	8	0	0	0

140489

The same table can also be provided for each character length when the receive character length of I field is other than 8 bits.

Table 3.5.23 (b) Bit Patterns by Number of Bits /Character (No Fractions)

Bits/character	D3	D2	D1
5 bits/Character	0	0	1
6 bits/Character	0	1	0
7 bits/Character	0	0	0
8 bits/Character	0	1	1

140489

Bit D4: Parity error

This bit is latched when the parity select bit (D0 of WR4) is set and a parity error is detected in the receive data. Latch can be cleared by the error reset command (WR0 bits D5, D4, D3 = "1", "1", "0").

Bit D5: Receive overrun error

The receive data FIFO holds up to 3 characters. When more characters are received without read out by the MPU, the excess character is set to the receive FIFO. When this character is read by the MPU, this receive overrun error is set. Once set, bit D5 latches that state. When the error reset command (command 6 of WR0 bits D3 through D5) is written, this bit is also reset.

Bit D6: CRC/framing error

In the asynchronous mode, this bit is set when a framing error is detected in the received character. Because this bit is not latched, it is always updated.

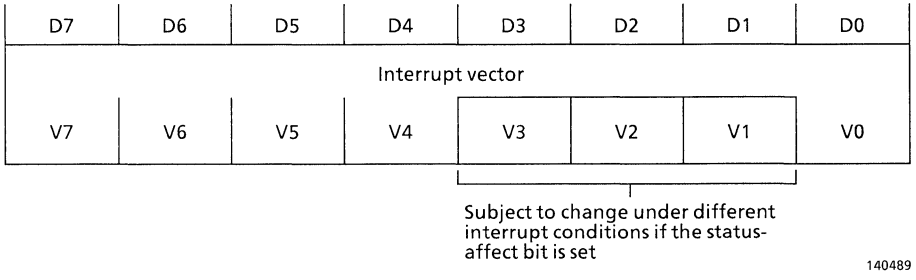
In the synchronous and SDLC modes, this bit indicates the transmitted CRC check result. This bit is reset when the error reset command (command 6 of WR0 bits D3 through D5) is written.

Bit D7: End of frame

This bit is set when the end flag is detected in the receive data and the CRC check and the fraction code are found normal. This bit is reset when the error reset command (command 6 of WR0 bits D3 through D5) is written. This bit is used only in the SDLC mode and is updated when the first character of the next frame is received.

3 RR 2; Read register 2

Table 3.5.24 Configuration of Read Register 2



When the status affect vector bit (D2 of WR1 (Channel B)) is set, bits V3 through V1 are changed depending on the interrupt condition at the time. The vector to be read is determined by the interrupt condition having the highest priority at the time of read. When the status affect vector bit is reset, the contents of this register are the same as those of WR2.

3.5.6 Using SIO

The following describes some system examples using the SIO. Figure 3.5.14 shows an inter-processor communication system. In this example, the MPU on the left side controls the data transfer with the modules on the right side. Both diagrams shown in Figure 3.5.14 (a) and (b) are communication systems. As shown, the SIO is used to interface with external devices in data communication. The greatest advantage of the SIO is the smaller number of data lines than parallel communication.

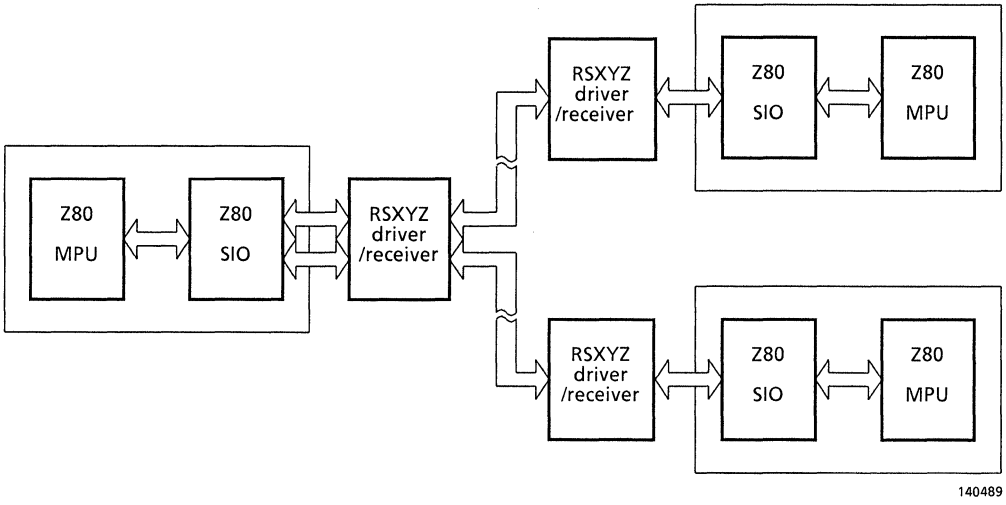


Figure 3.5.14 (a) Example of Data Communication Between Processors

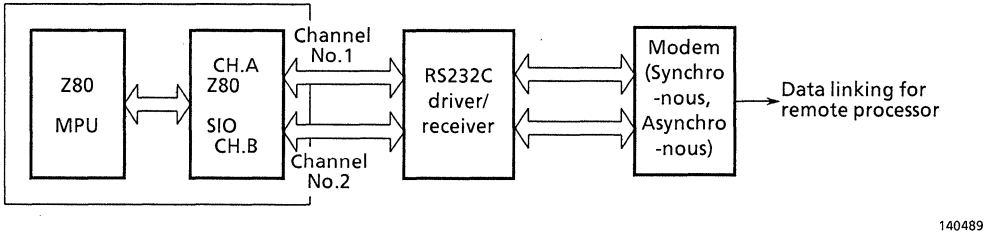


Figure 3.5.14 (b) Example of Data Communication Between Processor

3.6 STANDBY CAPABILITY

When a HALT instruction is executed, the TMPZ84C013A is put in one of the Run, Idle-1, Idle-2, or Stop mode depending on the contents of the halt mode setting register (#F0 : bit 4, bit 3 : HALTMR). (However, the TMPZ84C013A is put in the Run mode immediately after the reset operation by the $\overline{\text{RESET}}$ pin.) The halt mode setting register is set as follows. For the description and timing of each mode, see Subsection 3.3 "CGC Operations. "

The halt mode setting register is assigned to bits 4 and 3 of address F0 in the I/O address area. The halt mode is released by the interrupt (the nonmaskable interrupt by the $\overline{\text{NMI}}$ pin or the maskable interrupt by the $\overline{\text{INT}}$ pin) or by the reset through the $\overline{\text{RESET}}$ pin. A maskable interrupt is accepted when the MPU is in the EI state (in the state after the execution of EI instruction). A nonmaskable interrupt is accepted unconditionally. When an interrupt is accepted, the interrupt processing starts.

When the MPU is in the DI state (after the reset operation and the execution of DI instruction) with maskable interrupt, the TMPZ84C013A returns to the halt mode after executing a HALT instruction (actually a NOP instruction).

3.6.1 Setting Halt Mode

Duplicate control is provided to prevent the stop of the watchdog timer operation which may be caused by the halt mode setting error due to program runaway.

The halt mode is set by the halt mode setting register (HALTMR) and the halt mode control register (#F1 : bits 7 through 0 : HALTMCR). Figure 3.6.1 shows the contents of the halt mode control register (HALTMCR). Figure 3.6.2 shows the contents of the halt mode setting register (HALTMR).

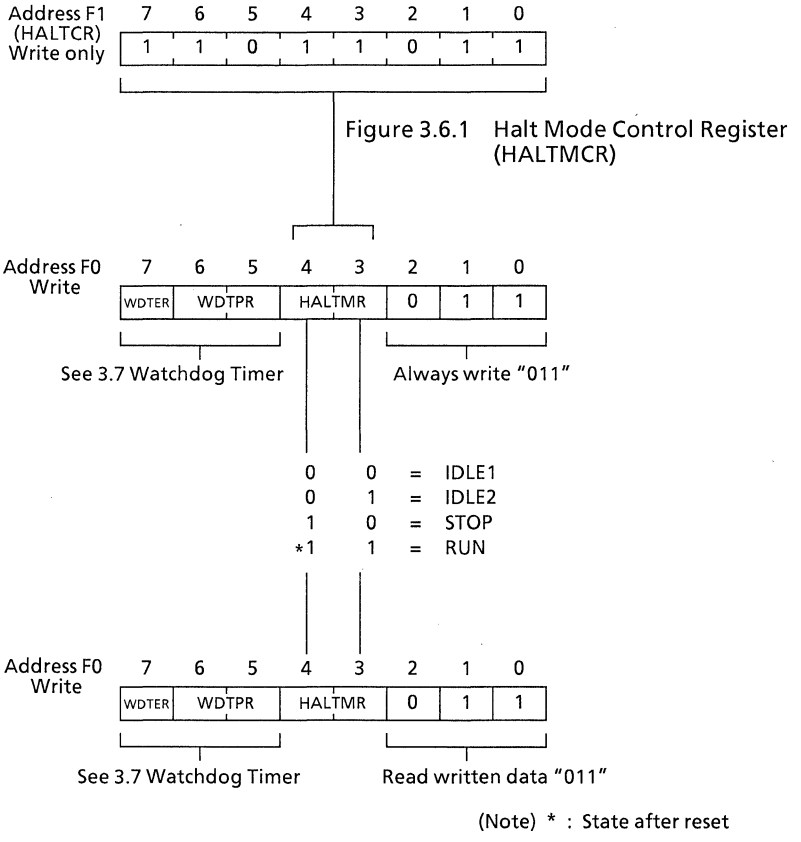


Figure 3.6.2 Halt Mode Setting Register (HALTMR)

Figure 3.6.3 shows the device states in the halt state with the CLKOUT pin connected to the CLKIN pin.

MODE	CGC	MPU	CTC	SIO	Watchdog Timer (WDT)	CLKOUT PIN
IDLE1	○	×	×	×	×	×
IDLE2	○	×	○	×	×	○
STOP	×	×	×	×	×	×
RUN	○	○	○	○	○	○

140489

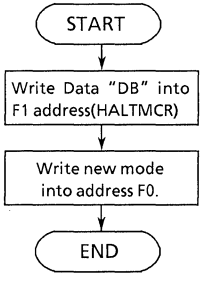
○……Operating (CLKOUT and CLKIN must be connected.)
×……Stop
Note: CLK OUT and CLK IN must be connected.

Figure 3.6.3 Device States in Halt State

For the halt mode in which the clock is supplied from the CLKIN pin (with the CGC oscillator unused), the Run mode must be used.

3.6.2 Halt Mode Setting Procedure

After reset, the halt mode is changed to the Run mode. Figure 3.6.4 shows the procedure to set a new mode.



140489

Figure 3.6.4 Setting Halt Mode

3.7 WATCHDOG TIMER

The watchdog timer (WDT) detects an operation error caused by the program runaway to return to the normal operation.

3.7.1 Block Diagram of Watchdog Timer

Figure 3.7.1 shows the block diagram of the watchdog timer.

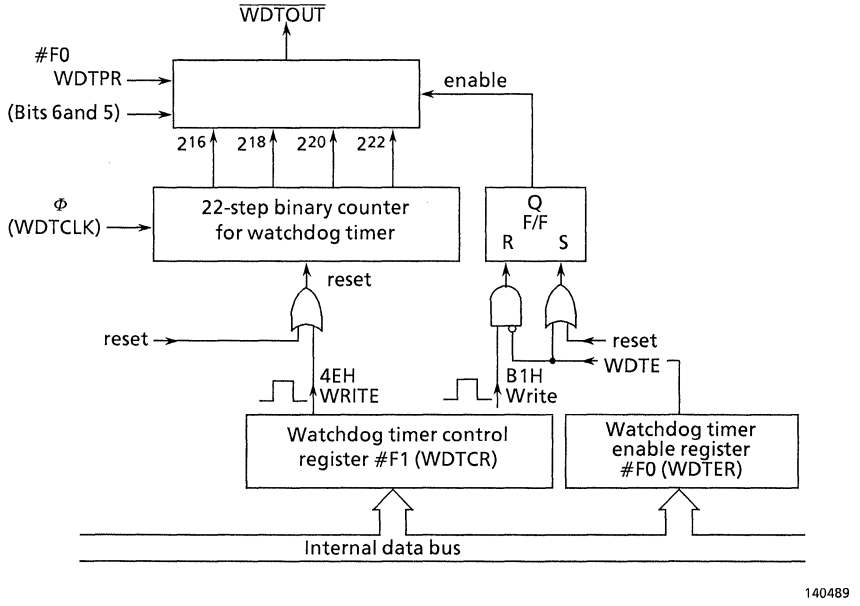


Figure 3.7.1 Block Diagram of Watchdog Timer

3.7.2 Setting watchdog Timer

(1) Enabling the watchdog timer

The watchdog timer can be set by the watchdog timer enable register (#F0 : bit 7 : WDTER) and the watchdog timer periodic register (#F0 : bit 6, bit 5 : WDTPR).

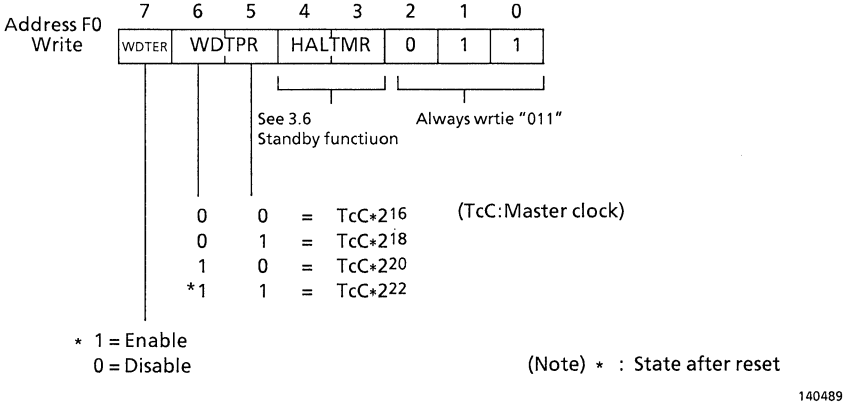


Figure 3.7.2 Enabling Watching Timer

(2) Disabling the watchdog timer

The watchdog timer can be disabled by disabling the watchdog timer enable register (WDTER) then writing data "B1" in the watchdog timer control register (#F1 : bit 7 through bit 0 : WDTCR).

This function has a duplicate structure to prevent the watchdog timer setting error, which may lead to the watchdog timer operation stop, caused by program runaway.

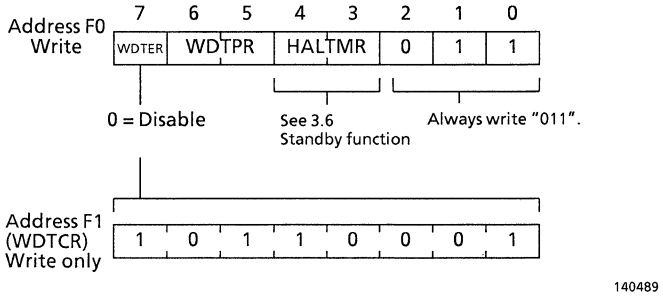


Figure 3.7.3 Disabling Watchdog Timer

(3) Clearing the watchdog timer

The watchdog timer can be cleared by writing data "4E" in the watchdog timer control register (WDTCR).

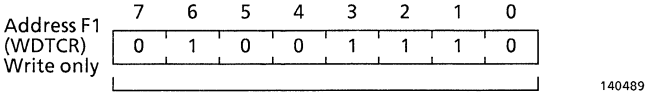


Figure 3.7.4 Cleaning The Watchdog Timer

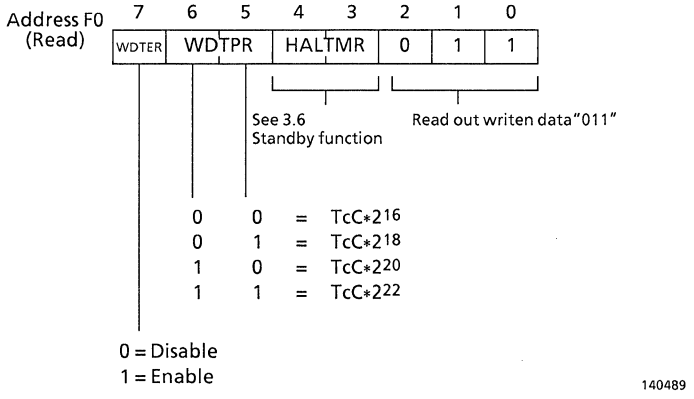


Figure 3.7.5 Reading Watchdog Timer Setting Register

3.7.3 Watchdog Timer Output

When the enabled watchdog timer is used, the "0" level signal is output to the $\overline{\text{WDTOUT}}$ pin after the duration of time specified in the watchdog timer periodic register (WDTPR). The output pulse width is one of the following two types depending on the $\overline{\text{WDTOUT}}$ pin connection :

(1) The $\overline{\text{WDTOUT}}$ connected to the $\overline{\text{RESET}}$ pin: The "0" level pulse of 5TcC (system clock) is output.

(2) The $\overline{\text{WDTOUT}}$ connected to a pin other than $\overline{\text{RESET}}$ pin : The "0" level pulse is kept output until the watchdog timer is cleared by software or reset by the $\overline{\text{RESET}}$ pin.

3.8 INTERRUPT PRIORITY

The programmable interrupt priority register (#F4 : bits 2 through 0 : INTPR) is provided to determine the interrupt priority for the CTC, SIO, in the TMPZ84C013A.

3.8.1 Setting Interrupt Priority

Figure 3.8.1 shows the register to determine the daisy chain interrupt priority for the CTC, SIO.

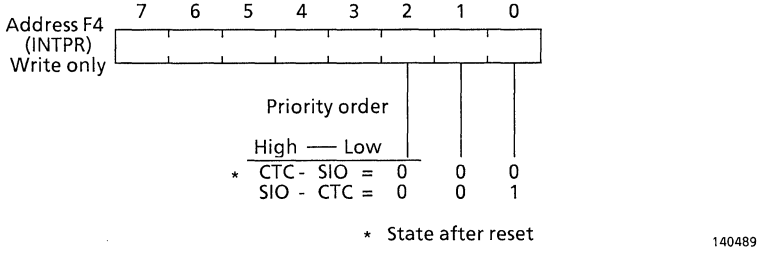


Figure 3.8.1 Interrupt Priority Register (INTPR)

[Example]

When "001" is written in address F4 (INTPR), the daisy chain interrupt priority is given as shown in Figure 3.8.2.

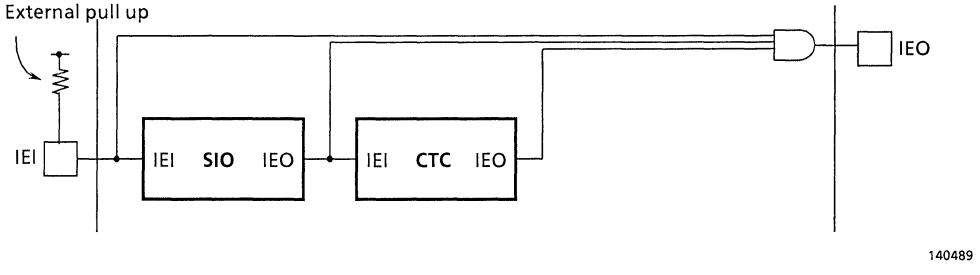


Figure 3.8.2 Daisy Chain Interrupt Priority

4. ELECTRICAL CHARACTERISTICS

4.1 MAXIMUM RATINGS

SYMBOL	ITEM		RATING
VCC	Vcc Supply Voltage with respect to Vss		– 0.5V to 7V
VIN	Input Voltage		– 0.5V to VCC + 0.5V
PD	Power Dissipation (6MHz VERSION : TA = 85°C) (8MHz VERSION : TA = 70°C)		250mW
TSOLDER	Soldering Temperature (Soldering Time 10sec)		260°C
TSTG	Storage Temperature		– 55°C to 125°C
TORP	Operating Temperature	6MHz VERSION	– 40°C to 85°C
		8MHz VERSION	– 10°C to 70°C

030489

4.2 DC ELECTRICAL CHARACTERISTICS

6MHz VERSION : TOPR = −40°C to +85°C, VCC = 5V ± 10%, VSS = 0V
8MHz VERSION : TOPR = −10°C to +70°C, VCC = 5V ± 5%, VSS = 0V

(1/2)

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT.
VILC	Clock Input Low Voltage (CLKIN)		– 0.3	—	0.6	V
VIHC	Clock Input High Voltage (CLKIN)		VCC-0.6	—	VCC + 0.3	V
VIL	Input Low Voltage (except XTAL1, $\overline{RESET}$)		– 0.5	—	0.8	V
VIH	Input High Voltage (except XTAL1, $\overline{RESET}$)		2.2	—	VCC	V
VILR	Input Low Voltage ($\overline{RESET}$)		– 0.5	—	0.45	V
VIHR	Input High Voltage ($\overline{RESET}$)		VCC-0.6	—	VCC	V
VOLC	Output Low Voltage (CLKOUT)	IOL = 2.0mA	—	—	0.6	V
VOHC	Output High Voltage (CLKOUT)	IOH = – 2.0mA	VCC-0.6	—	—	V
VOL	Output Low Voltage (except CLKOUT)	IOL = 2.0mA	—	—	0.4	V
VOH1	Output High Voltage 1 (except CLKOUT)	IOH = – 1.6mA	2.4	—	—	V
VOH2	Output High Voltage 2 (except CLKOUT)	IOH = – 250uA	VCC-0.8	—	—	V
ILI	Input Leakage Current	Vss ≦ VIN ≦ Vcc	—	—	± 10	μA
ILO	3-state Output Leakage Current in Float	Vss ≦ Vout ≦ VCC	—	—	± 10	μA

030489

(2/2)

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.	MAX.	UNIT.
ICCI	Power Supply Current	VCC = 5V, fCLK = (1) VIHC = VIHR = VIH = VCC-0.2V, VILC = VILR = VIL = 0.2V	AT-6	—	22	27	mA
			AT-8	—	31	35	
ICC2	Stand-by Supply Current (See Note(2))	VCC = 5V, fCLK = Stopped, VIHC = VIH = VIHR = VCC-0.2V, VILC = VIL = VILR = 0.2V	AT-6 /AT-8	—	0.5	50	μA
ICC3	Power Supply Current (IDLE1 Mode)	VCC = 5V, fCLK = (1), VIHC = VIH = VIHR = VCC-0.2V, VILC = VIL = VILR = 0.2V	AT-6	—	1.5	3	mA
			AT-8	—	2	4	mA
ICC4	Power Supply Current (IDLE2 Mode)	VCC = 5V, fCLK = (1), VIHC = VIH = VIHR = VCC-0.2V, VILC = VIL = VILR = 0.2V	AT-6	—	11	15	mA
			AT-8	—	15	20	mA

030489

Note :

(1) fCLK = 1/TcC (MIN)

(2) ICC2 Stand-by Supply Current is guaranteed only when the supplied clock is stopped at a low level during T4 state of the following machin Cycle (M1) next to OP code fetch Cycle of HALT instruction.
Except $\overline{SYNCA} = 0$ or $\overline{SYNCB} = 0$ state

4.3 AC ELECTRICAL CHARACTERISTICS (1) (IN ACTIVE STATE)

6MHz VERSION : TA = −40°C~85°C, VCC = 5V ± 10%, VSS = 0V
8MHz VERSION : TA = −10°C~70°C, VCC = 5V ± 5%, VSS = 0V

4.3.1 AC Characteristics of CPU (in Active State)

(1/3)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
1	TcC	Clock Cycle Time	162	—	DC	125	—	DC	ns
2	TwCh	Clock Pulse Width (High)	65	—	DC	50	—	DC	ns
3	TwCl	Clock Pulse Width (Low)	65	—	DC	50	—	DC	ns
4	TfC	Clock Fall Time	—	—	20	—	—	15	ns
5	TrC	Clock Rise Time	—	—	20	—	—	15	ns
6	TdCr (A)	Clock ↑ to Address Valid Delay	—	—	90	—	—	85	ns
7	TdA (MREQf)	Address Valid to $\overline{MREQ}$ ↓ Delay	35	—	—	20	—	—	ns
8	TdCf (MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay	—	—	70	—	—	60	ns
9	TdCr (MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay	—	—	70	—	—	60	ns
10	TwMREQh	$\overline{MREQ}$ pulse Width (High)	65	—	—	45	—	—	ns
11	TwMREQ1	$\overline{MREQ}$ pulse Width (Low)	135	—	—	100	—	—	ns
12	TdCf (MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay	—	—	70	—	—	60	ns
13	TdCf (RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay	—	—	80	—	—	70	ns
14	TdCr (RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay	—	—	70	—	—	60	ns
15	TsD (Cr)	Data Setup Time to Clock ↑	30	—	—	30	—	—	ns
16	ThD (RDr)	Data Hold Time to $\overline{RD}$ ↑	0	—	—	0	—	—	ns
17	TsWAIT (Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	60	—	—	50	—	—	ns
18	ThWAIT (Cf)	$\overline{WAIT}$ Hold Time after Clock ↓	10	—	—	10	—	—	ns
19	TdCr (M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay	—	—	80	—	—	70	ns
20	TdCr (M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay	—	—	80	—	—	70	ns
21	TdCr (RFSHf)	Clock ↑ to $\overline{RFSH}$ ↓ Delay	—	—	110	—	—	95	ns
22	TdCr (RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay	—	—	100	—	—	85	ns
23	TdCf (RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay	—	—	70	—	—	60	ns
24	TdCr (RDf)	Clock ↑ to $\overline{RD}$ ↓ Delay	—	—	70	—	—	60	ns

030489

(2/3)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
25	TsD (Cf)	Data Setup to Clock ↓ during M2, M3, M4 or M5 Cycles	40	—	—	30	—	—	ns
26	TdA (IORQf)	Address Stable prior to $\overline{IORQ}$ ↓	110	—	—	75	—	—	ns
27	TdCr (IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay	—	—	65	—	—	55	ns
28	TdCf (IORQr)	Clock ↓ to $\overline{IORQ}$ ↑ Delay	—	—	70	—	—	60	ns
29	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	25	—	—	5	—	—	ns
30	TdCf (WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay	—	—	70	—	—	60	ns
31	TwWR	$\overline{WR}$ Pulse Width	135	—	—	100	—	—	ns
32	TdCf (WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay	—	—	70	—	—	60	ns
33	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	−55	—	—	−55	—	—	ns
34	TdCr (WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay	—	—	60	—	—	55	ns
35	TdWRr (D)	Data Stable from $\overline{WR}$ ↑	30	—	—	15	—	—	ns
36	TdCf (HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓	—	—	260	—	—	225	ns
37	TwNMI	$\overline{NMI}$ Pulse Width	80	—	—	80	—	—	ns
38	TsBUSREQ (Cr)	$\overline{BUSREQ}$ Setup Time to Clock ↑	50	—	—	40	—	—	ns
39	ThBUSREQ (Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	10	—	—	10	—	—	ns
40	TdCr (BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay	—	—	90	—	—	80	ns
41	TdCf (BUSACKr)	Clockt ↓ to $\overline{BUSACK}$ ↑ Delay	—	—	90	—	—	80	ns
42	TdCr (Dz)	Clock ↑ to Data Float Delay	—	—	80	—	—	70	ns
43	TdCr (CTz)	Clock ↑ to Control Outputs Float Delay ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)	—	—	70	—	—	60	ns
44	TdCr (AZ)	Clock ↑ to Address Float Delay	—	—	80	—	—	70	ns
45	TdCr (A)	$\overline{MREQ}$ ↑, $\overline{IORQ}$ ↑, $\overline{RD}$ ↑, and $\overline{WR}$ ↑ to Address Hold Time	35	—	—	20	—	—	ns
46	TsRESET (Cr)	$\overline{RESET}$ to Clock ↑ Setup Time	60	—	—	45	—	—	ns
47	ThRESET (Cr)	$\overline{RESET}$ to Clock ↑ Hold Time	10	—	—	10	—	—	ns
48	TsINTf (Cr)	$\overline{INT}$ to Clock ↑ Setup Time	70	—	—	55	—	—	ns
49	TsINTr (Cr)	$\overline{INT}$ to Clock ↑ Hold Time	10	—	—	10	—	—	ns

190589

(3/3)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
50	TdM1f (IORQf)	$\overline{M1}\downarrow$ to $\overline{IORQ}\downarrow$ Delay	365	—	—	270	—	—	ns
51	TdCf (IORQf)	Clock $\downarrow$ to $\overline{IORQ}\downarrow$ Delay	—	—	70	—	—	60	ns
52	TdCr (IORQr)	Clock $\uparrow$ to $\overline{IORQ}\uparrow$ Delay	—	—	70	—	—	60	ns
53	TdCf (D)	Clock $\downarrow$ to Data Valid Delay	—	—	130	—	—	115	ns

030489

4.3.2 AC Characteristics of CGC (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
54	TcC CLK	Output clock cycle	—	162	—	—	125	—	ns
55	TwCh CLK	Output clock Width(High)	—	70	—	50	—	—	ns
56	TwCl CLK	Output clock Width (Low)	—	70	—	50	—	—	ns
57	TfC CLK	Output clock fall time	—	12	—	—	—	12	ns
58	TrC CLK	Output clock rise time	—	12	—	—	—	12	ns
59	TRST (INT) S	CLK out restart time by $\overline{INT}$ (STOP Mode)	—	2^{14} + 2.5TcC	—	—	2^{14} + 2.5TcC	—	ns
60	TRST (NMI) S	CLK out restart time by NMI (STOP Mode)	—	2^{14} + 2.5TcC	—	—	2^{14} + 2.5TcC	—	ns
61	TRST (INT) I	CLK out restart time by $\overline{INT}$ (IDLE 1/2 Mode)	—	2.5 *TcC	—	—	2.5 *TcC	—	ns
62	TRST (NMI) I	CLK out restart time by $\overline{NMI}$ (IDLE 1/2 Mode)	—	2.5 *TcC	—	—	2.5 *TcC	—	ns
63	TRST (RESET) I	CLK out restart time by $\overline{RESET}$ (IDLE 1/2 Mode)	—	TcC	—	—	TcC	—	ns

030489

4.3.3 AC Characteristics of CTC (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C013AF-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
64	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall (in case of generating only interrupt immediately before M1 cycle)	—	—	160	—	—	130	ns
65	TdIEI (IEOf)	Delay from IEI fall to IEO fall	—	—	70	—	—	50	ns
66	TdIEI (IEOr)	Delay from IEI rise to IEO rise (after ED decode)	—	—	150	—	—	120	ns
67	IsCLK ($\overline{INT}$)	CLK/TRG setup to TL ↑ for detection of interrupt tsCTR (c) Satisfied	TcC + 120 + T68 + T48	—	—	TcC + 100 + T68 + T48	—	—	ns
		tsCTR (c) not Satisfied	2TcC + 120 + T68 + T48	—	—	2TcC + 100 + T68 + T48	—	—	
68	TcCTR	CLK/TRG Frequency (counter mode)	2TcC	—	—	2TcC	—	—	ns
69	TrCTR	CLK/TRG rising time	—	—	40	—	—	30	ns
70	TfCTR	CLK/TRG falling time	—	—	40	—	—	30	ns
71	TwCTR1	CLK/TRG Pulse Width (Low)	120	—	—	90	—	—	ns
72	TwCTRh	CLK/TRG Pulse Width (High)	120	—	—	90	—	—	ns
73	TsCTR (Cs)	CLK/TRG ↑ to Clock ↑ Setup Time for Immediate Count (counter mode)	150	—	—	110	—	—	ns
74	TsCTR (CT)	CLK/TRG ↑ to Clock ↑ Setup Time for enabling of Prescaler on following clock ↑ (timer mode)	150	—	—	110	—	—	ns
75	TdC (ZC/TOr)	Clock ↑ to ZC/TO ↑ Delay	—	—	140	—	—	110	ns
76	TdC (ZC/TOf)	Clock ↓ to ZC/TO ↓ Delay	—	—	140	—	—	110	ns

030489

4.3.4 AC Characteristics of SIO (in Active State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
92	TsM1 (C)	$\overline{M1}$ ↑ to clock ↑ Setup time	75	—	—	50	—	—	ns
93	TsIEI (IO)	IEI ↓ to $\overline{IORQ}$ ↓ Setup time (INTACK cycle)	120	—	—	100	—	—	ns
94	TdM1 (IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (Interrupt before $\overline{M1}$)	—	—	160	—	—	120	ns
95	TdIEI (IEOr)	$\overline{IEI}$ ↓ to IEO ↑ Delay (after ED decode)	—	—	150	—	—	120	ns
96	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	—	—	50	ns
97	TdIO (W/RWf)	$\overline{IORQ}$ ↓ or $\overline{CE}$ ↓ to $\overline{W/RDY}$ ↓ Delay (Wait mode)	—	—	175	—	—	130	ns
98	TdC (W/RRf)	Clock ↑ to $\overline{W/RDY}$ ↓ Delay (Ready Mode)	—	—	100	—	—	80	ns
99	TdC (W/RWZ)	Clock ↓ to $\overline{W/RDY}$ float delay (Wait mode)	—	—	110	—	—	90	ns
100	TwPh	Pulse Width (High)	200	—	—	200	—	—	ns
101	TwPl	Pulse Width (Low)	200	—	—	200	—	—	ns
102	TcTxC	$\overline{TxC}$ cycle time	330	—	∞	250	—	∞	ns
103	TwTxCl	$\overline{TxC}$ Width (Low)	100	—	∞	80	—	∞	ns
104	TwTxCh	$\overline{TxC}$ Width (High)	100	—	∞	80	—	∞	ns
105	TdTxC (TxD)	$\overline{TxC}$ ↓ toTxD Delay (x1 mode)	—	—	220	—	—	180	ns
106	TdTxC (W/RRf)	$\overline{TxC}$ ↓ to$\overline{W/RDY}$ ↓ Delay (Ready mode)	5	—	9	5	—	9	CLK Periods
107	TcRxC	$\overline{RxC}$ cycle time	330	—	∞	250	—	∞	ns
108	TwRxCl	$\overline{RxC}$ Width (Low)	100	—	∞	80	—	∞	ns
109	TwRxCh	$\overline{RxC}$ Width (High)	100	—	∞	80	—	∞	ns
110	TsRxD (Rxc)	RxD to $\overline{RxC}$ ↑ Setup time (x1 mode)	0	—	—	10	—	—	ns
111	ThRxD (Rxc)	$\overline{RxC}$ ↑ to RxD Hold time (x1 mode)	100	—	—	80	—	—	ns
112	TdRxC (W/RRf)	$\overline{RxC}$ ↑ to $\overline{W/RDY}$ ↓ Delay (Ready mode)	10	—	13	10	—	13	CLK Periods
113	TdRxC (SYNC)	$\overline{RxC}$ ↑ to $\overline{SYNC}$ ↓ Delay (Output mode)	4	—	7	4	—	7	CLK Periods

030489

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
114	TsSYNC (RxC)	$\overline{SYNC}$ ↑ to $\overline{RxC}$ ↓ Setup (External SYNC modes)	−100	—	—	−100	—	—	ns
115	IsTxC	$\overline{TxC}$ Setup to TL ↑ for detection of interrupt	5*T1 +T48	—	9*T1 +T48	5*T1 +T48	—	9*T1 +T48	ns
116	IsRxC	$\overline{RxC}$ Setup to TL ↑ for detection of interrupt	10*T1 +T48	—	13*T1 +T48	10*T1 +T48	—	13*T1 +T48	ns

030489

4.3.5 AC Chracteristics of WDT (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			TMPZ84C013AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
117	TdC (WDTf)	Clock ↑ to $\overline{WDTOUT}$ ↓ Delay	—	—	160	—	—	120	ns
118	Tdc (WDTr)	Clock ↑ to $\overline{WDTOUT}$ ↑ Delay	—	—	165	—	—	125	ns
119	TcWDT	$\overline{WDTOUT}$ OUTPUT Period							
		WDT Mode0	—	$T1*2^{16}$	—	—	$T1*2^{16}$	—	ns
		WDT Mode1	—	$T1*2^{18}$	—	—	$T1*2^{18}$	—	
		WDT Mode2	—	$T1*2^{20}$	—	—	$T1*2^{20}$	—	
		WDT Mode3	—	$T1*2^{22}$	—	—	$T1*2^{22}$	—	

030489

Note1 : Timing Measurements are made at the following voltage.

Input VIH=2.4V, VIL=0.4V, VIHC=VCC−0.6V, VILC=0.6V

Output VOH=2.2V, VOL=0.8V, (Except CLK OUT)

CL=100pF

4.4 AC TIMING CHARTS (1) (IN ACTIVE STATE)

4.4.1 AC Timing Charts of CPU (in Active State)

Figure 4.4.1 through 4.4.8 show the basic timing charts. The circled numbers in these charts correspond to the numbers in the number column of the AC Electrical Characteristics Tables.

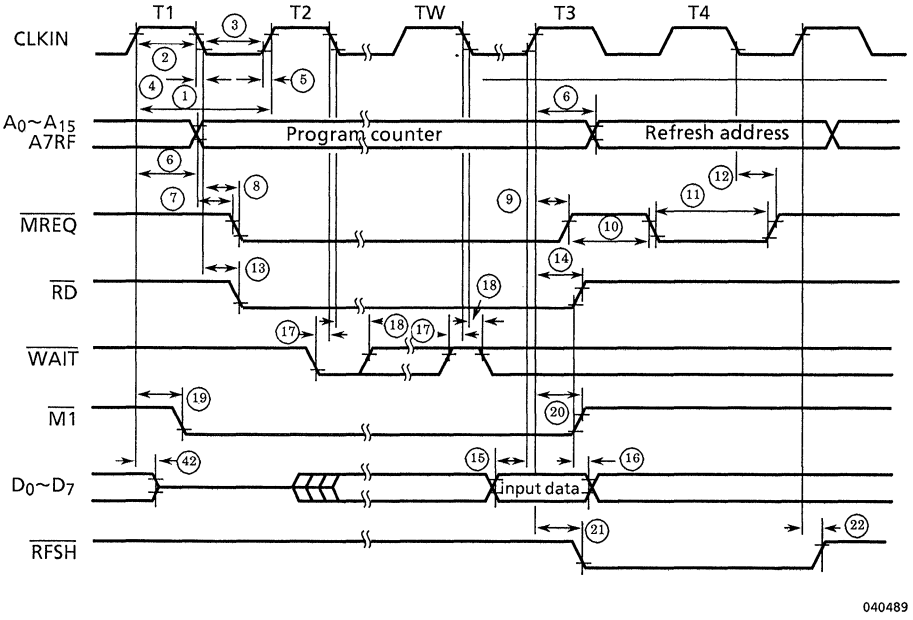


Figure 4.4.1 Opcode Fetch Cycle

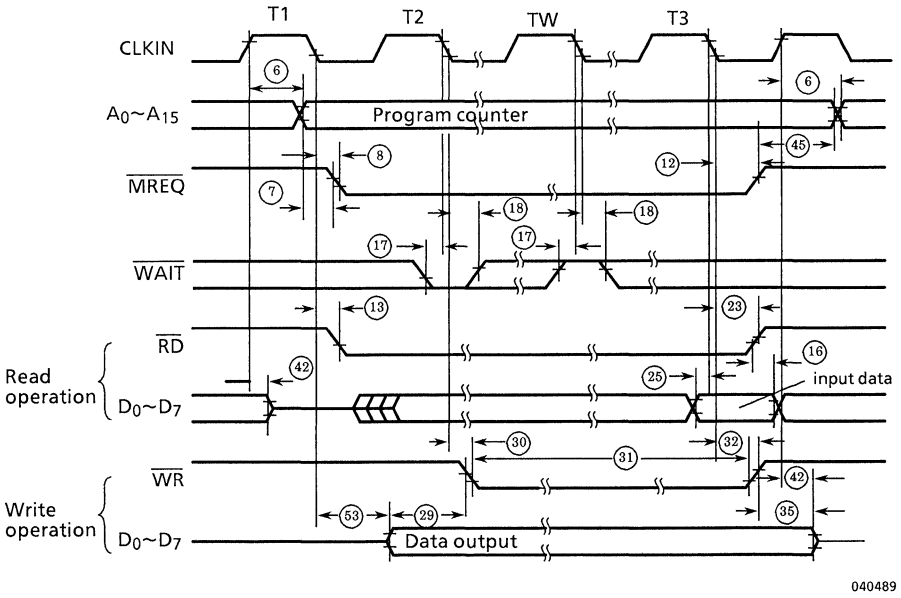
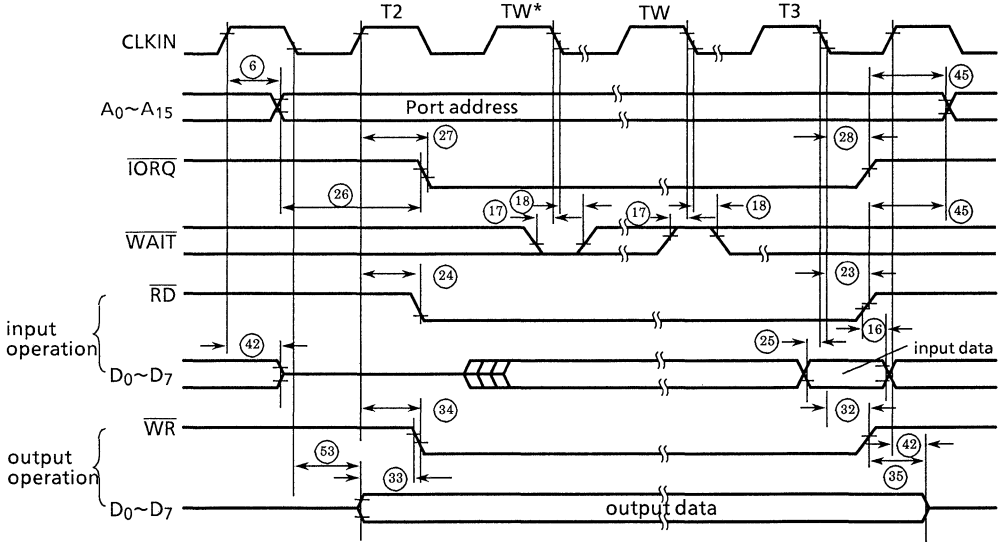


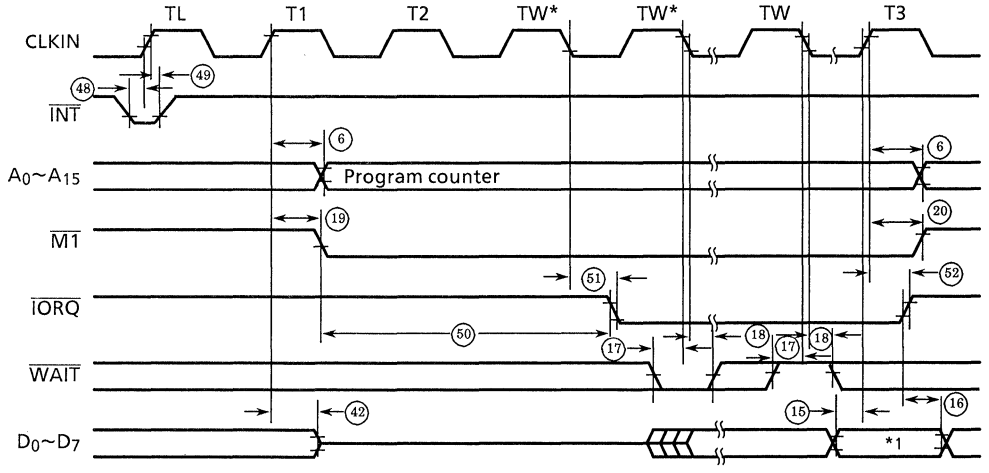
Figure 4.4.2 Memory Read/Write Cycle



Note : 1-wait state (TW*) is inserted automatically by MPU.

040489

Figure 4.4.3 I/O Cycle



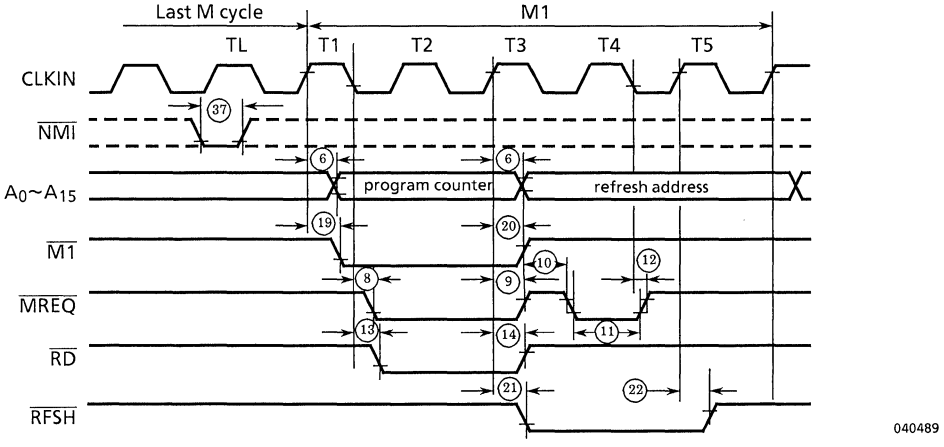
Notes :

1. TL : Last state of instruction
2. 2-wait state (TW*) is inserted automatically by MPU

*1 VALID DATA

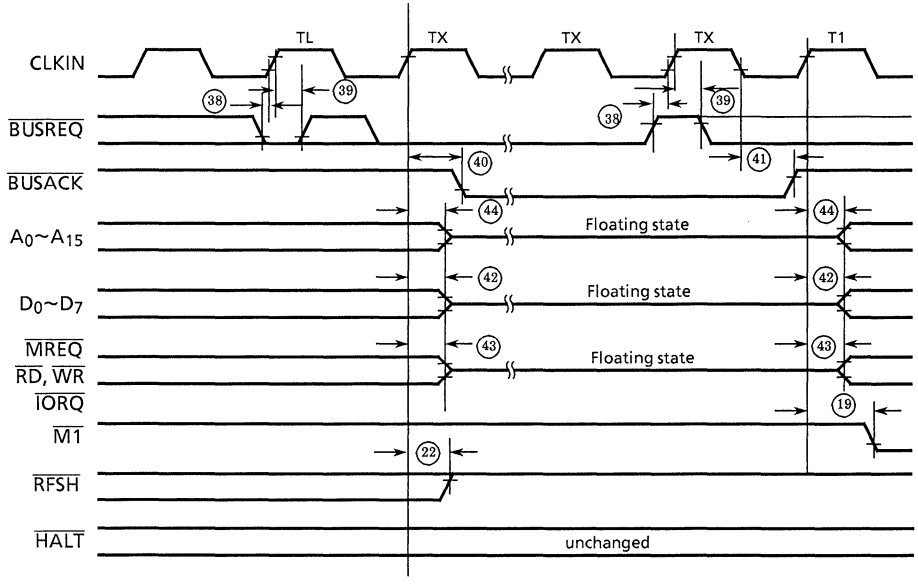
040489

Figure 4.4.4 Interrupt Request/Acknowledge Cycle



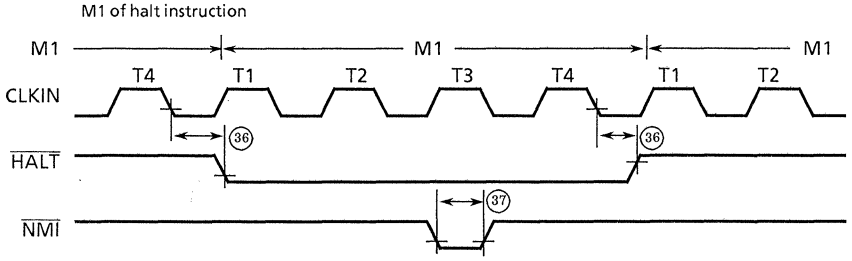
Note: $\overline{NMI}$ is asynchronous, but its falling edge signal must occur synchronously with the rising edge of previous TL state for correct response to the subsequent machine cycles.

Figure 4.4.5 Non-maskable Interrupt Request Cycle



Notes
1. TL : Last state of a given machine cycle
2. TX : Clock used by peripheral LSI that made request.

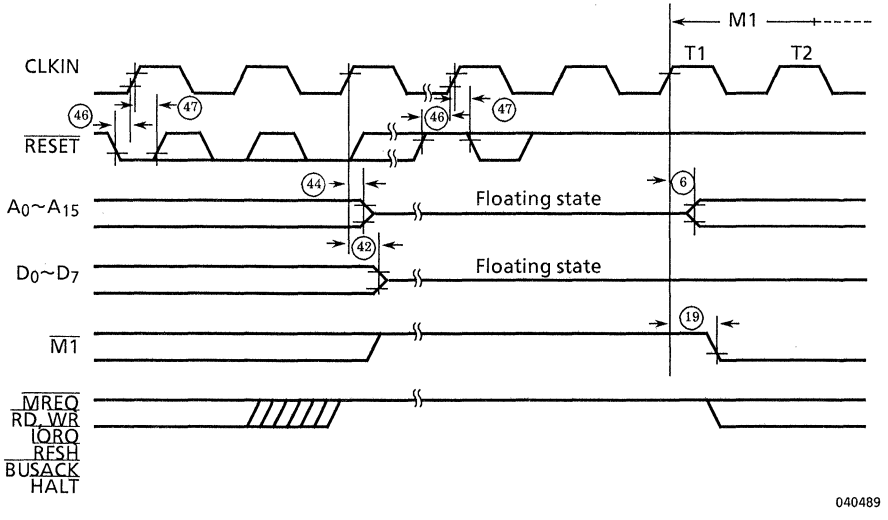
Figure 4.4.6 Bus Request/Acknowledge Cycle



Note : $\overline{INT}$ signal is also used for releasing HALT state.

040489

Figure 4.4.7 Halt Acknowledge Cycle

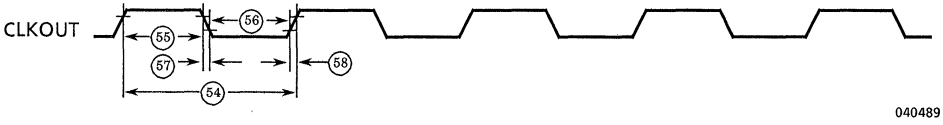


040489

Figure 4.4.8 Reset Cycle

4.4.2 AC Timing Charts of CGC (in Active State)

The following Figures show the timings in each operation mode with the CLKOUT pin connected to the CLKIN pin.



040489

Figure 4.4.9 CLKOUT Waveform

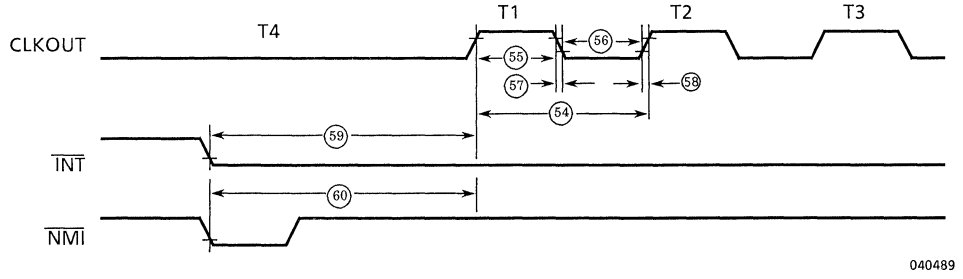


Figure 4.4.10 Clock Restart Timing (STOP mode)

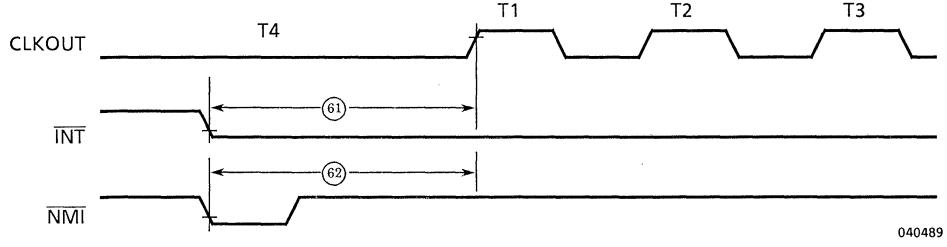


Figure 4.4.11 Clock Restart Timing (IDLE1 mode)

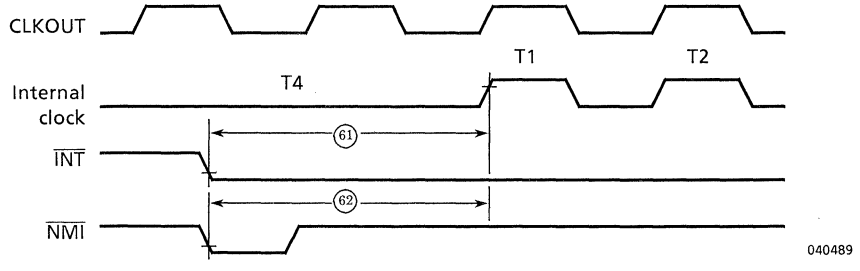


Figure 4.4.12 Clock Restart Timing (IDLE2 mode)

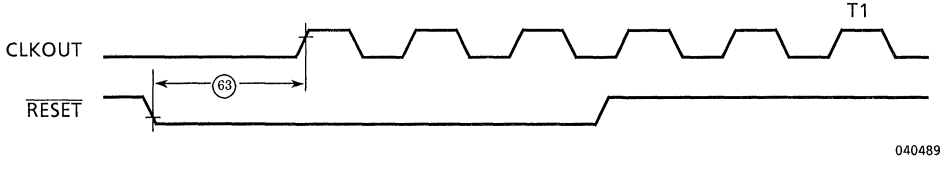


Figure 4.4.13 Timing of Clock Start by $\overline{\text{RESET}}$ (IDLE1 and IDLE2 modes)

4.4.3 AC Timing Charts of CTC (in Active State)

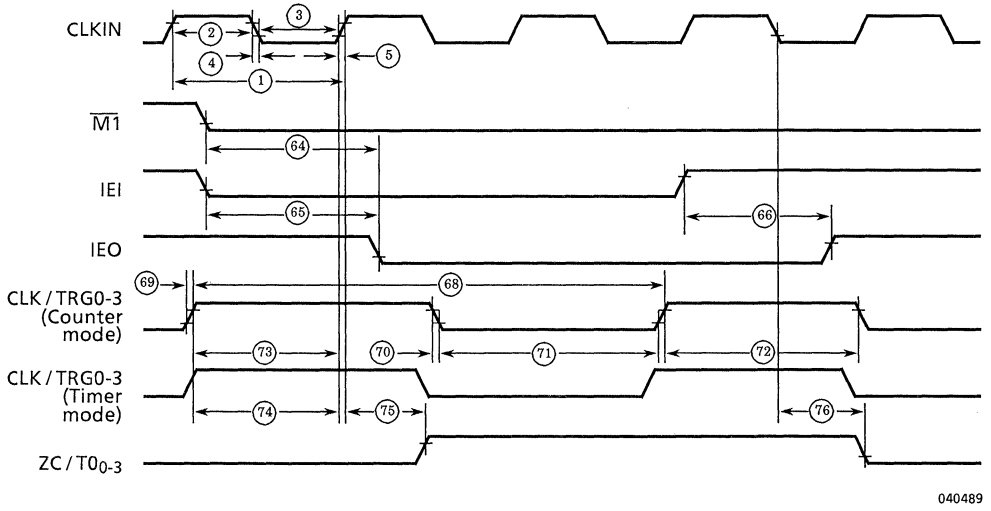


Figure 4.4.14 CTC Timing Diagram

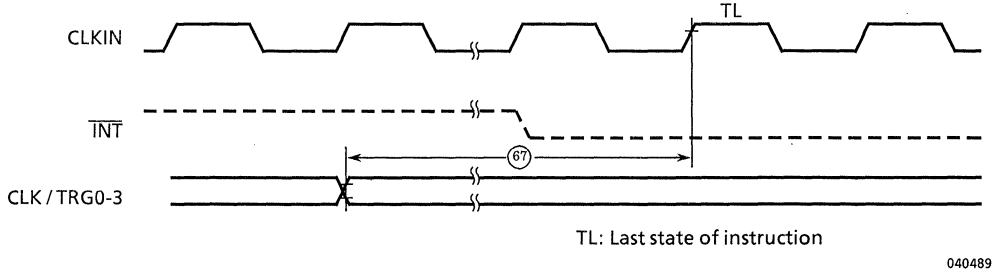


Figure 4.4.15 CTC Interrupt Occurrence Timing

4.4.4 AC Timing Charts of SIO (in Active State)

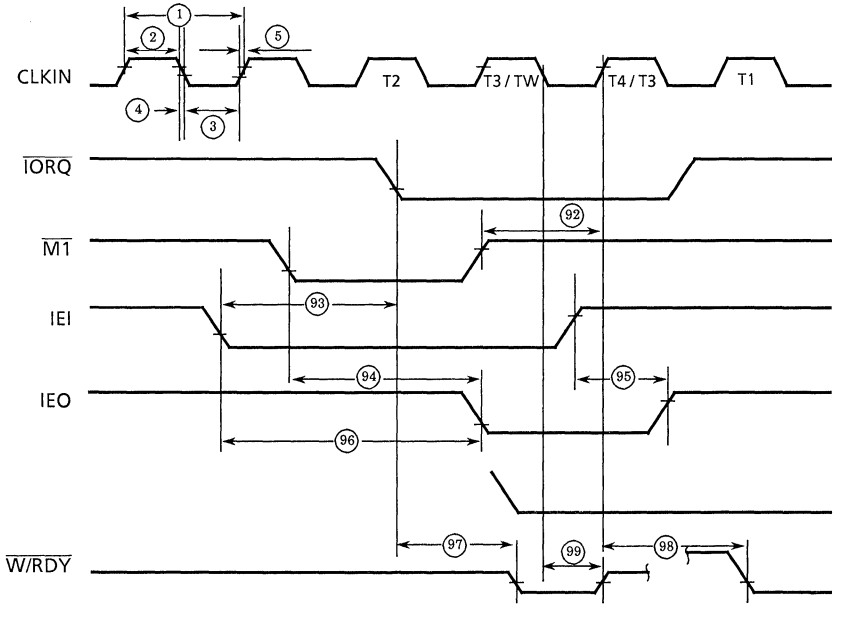


Figure 4.4.16 (a) SIO Timing Diagram

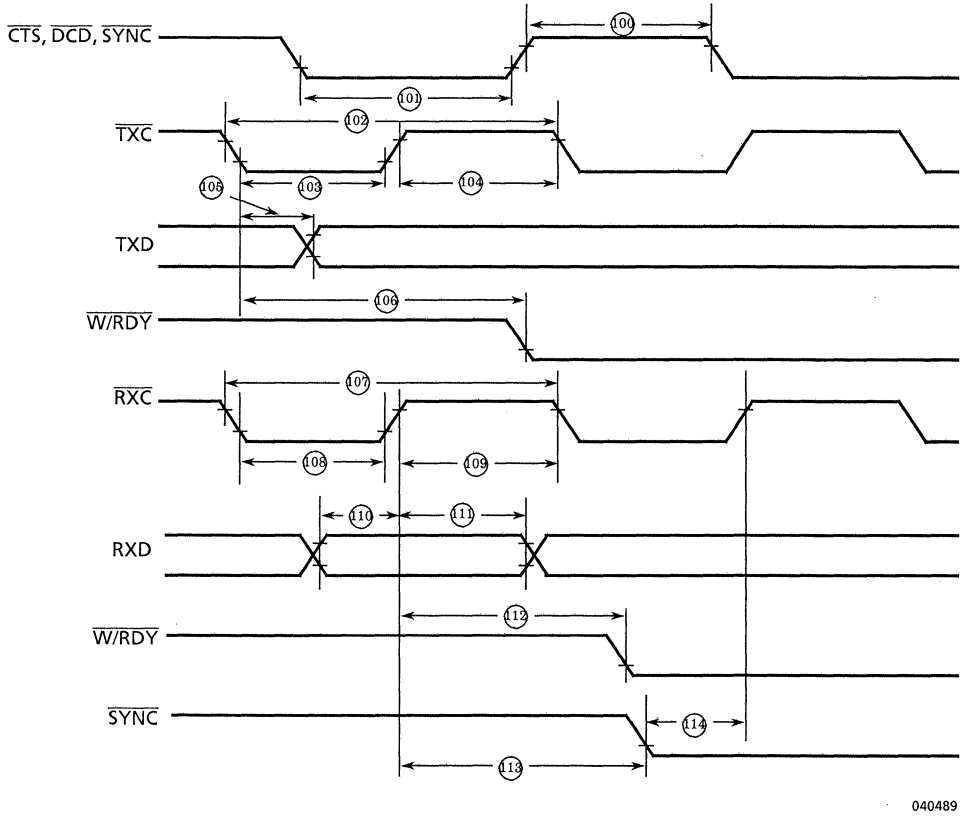


Figure 4.4.16 (b) SIO Timing Diagram

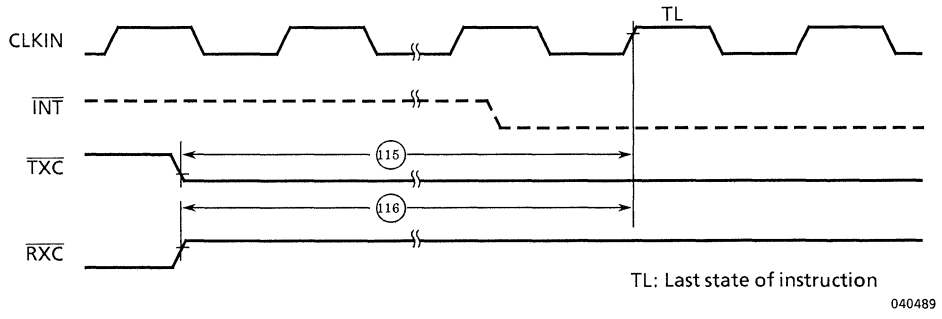


Figure 4.4.17 SIO Interrupt Occurrence Timing

4.4.5 AC Timing Charts of WDT (in Active State)

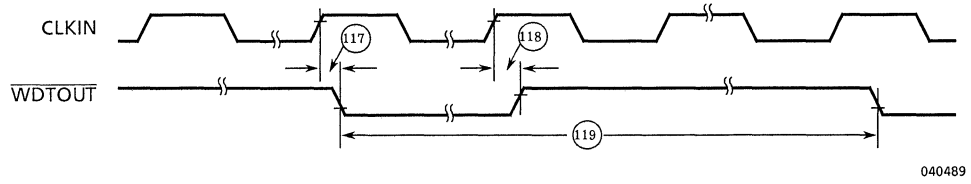


Figure 4.4.18 WDT Timing Diagram

4.5 AC ELECTRICAL CHARACTERISTICS (2) (IN INACTIVE STATE)

6MHz VERSION : TA = −40°C~+85°C, VCC = 5V+10%, VSS = 0V

4.5.1 AC Characteristics of CGC (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
1	TcCCLK	Output Clock Cycle	—	162	—	ns
2	TwChCLK	Output Clock Width (High)	—	70	—	ns
3	TwClCLK	Output Clock Width (Low)	—	70	—	ns
4	TfCCLK	Output Clock fall Time	—	12	—	ns
5	TrCCLK	Output Clock rise Time	—	12	—	ns
6	TRST (INT) S	Clock (CLKOUT) restart Time by $\overline{INT}$ (STOP mode)	—	2^{14} + 2.5TcC	—	ns
7	TRST (NMI) S	Clock (CLKOUT) restart Time by $\overline{NMI}$ (STOP mode)	—	2^{14} + 2.5TcC	—	ns
8	TRST (INT) S	Clock (CLKOUT) restart Time by $\overline{INT}$ (IDLE1/2 mode)	—	2.5 *TcC	—	ns
9	TRST (NMI) I	Clock (CLKOUT) restart Time by $\overline{NMI}$ (IDLE1/2 mode)	—	2.5 *TcC	—	ns
10	TRST (RESET) I	Clock (CLKOUT) restart Time by $\overline{RESET}$ (IDLE1/2 mode)	—	1TcC	—	ns
11	TsHALT (M1r)	$\overline{HALT}$ Setup Time	10	—	—	ns

040489

4.5.2 AC Characteristics of CTC (in Inactive State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
12	TcC	Clock Cycle Time	162	—	—	ns
13	Twch	Clock width (High)	65	—	—	ns
14	Twcl	Clock width (Low)	65	—	—	ns
15	TfC	Clock falling time	—	—	20	ns
16	TrC	Clock rising time	—	—	20	ns
17	Th	Hold Time	10	—	—	ns
18	TcCS (C)	CS (A1,A0) Setup time to clock ↑	100	—	—	ns
19	TSCE (C)	CE (A7~A2) Setup time to clock ↑	150	—	—	ns
20	TsIO (C)	$\overline{IORQ}$ ↓ Setup time to clock ↑	70	—	—	ns

040489

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
21	TsRD (C)	$\overline{RD}$ ↓ Setup time to clock ↑	70	—	—	ns
22	TdC (DO)	Clock ↑ to Data Valid Delay	—	—	130	ns
23	Tdc (DOz)	$\overline{IORQ}$, $\overline{RD}$ ↑ to Data Float Delay	—	—	90	ns
24	TsDI (C)	Data Input Setup time to clock ↑	40	—	—	ns
25	TsM1(C)	$\overline{M1}$ Setup time to clock ↑	70	—	—	ns
26	TdM1(IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (in case of generating only interrupt immediately before M1 cycle)	—	—	130	ns
27	TdIO (DOI)	$\overline{IORQ}$ ↓ to Data out Delay (INTA Cycle)	—	—	110	ns
28	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	ns
29	TdIEI (IEOr)	IEI ↑ to IEO ↑ Delay (after ED decode)	—	—	150	ns
30	TdC (INT)	Clock ↑ to $\overline{INT}$ ↓ Delay	—	—	TcC + 120	ns
31	TdA (IORQf)	CLK/TRG ↑ to $\overline{INT}$ ↓ Delay TsCTR (c) Satisfied TsCTR (c) not Satisfied	— —	TcC + 120 + 70 + T37 2TcC + 120 + 70 + T37	— —	ns
32	TcCTR	CLK/TRG Frequency	—	2TcC	—	ns
33	TrCTR	CLK/TRG rising time	—	—	40	ns
34	TfCTR	CLK/TRG falling time	—	—	40	ns
35	TwCTRI	CLK/TRG pulse width (Low)	120	—	—	ns
36	TwCTR	CLK/TRG pulse width (High)	120	—	—	ns
37	TsCTR (CS)	CLK/TRG ↑ to clock ↑ Setup Time for Immediate Count (counter mode)	150	—	—	ns
38	TsCTR (CT)	CLK/TRG ↑ to clock ↑ Setup Time for enabling of Prescaler on following clock ↑ (timer mode)	150	—	—	ns
39	TdC (ZC/TOr)	Clock ↑ to ZC/TO ↑ Delay	—	—	140	ns
40	TdC (ZC/TOf)	Clock ↓ to ZC/TO ↓ Delay	—	—	140	ns

040489

4.5.3 AC Characteristics of SIO (in Inactive State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
65	TsCS (C)	$\overline{CE}$ (A_7toA_2), C/$\overline{D}$ (A_0), B/$\overline{A}$ (A_1) Setup Time to Clock ↑	60	—	—	ns
66	TsRD (C)	$\overline{IORQ}$, $\overline{RD}$ Setup Time to clock ↑	60	—	—	ns
67	TdC (DO)	Clock ↑ to Data output Delay	—	—	150	ns
68	TsDI (C)	Data input setup time to clock ↑ (write cycle or $\overline{M1}$ cycle)	30	—	—	ns
69	TdRD (DOz)	$\overline{RD}$ ↑ to Data Out Float Delay	—	—	90	ns
70	TdIO (DOI)	$\overline{IORQ}$ ↓ to Data output Delay (INTACK cycle)	—	—	120	ns
71	TsM1 (C)	$\overline{M1}$ Setup time to clock ↑	75	—	—	ns
72	TsIEI (IO)	IEI Setup time to $\overline{IORQ}$ ↓ (INTACK cycle)	120	—	—	ns
73	TdM1 (IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (interrupt before $\overline{M1}$)	—	—	160	ns
74	TdIEI (IEOr)	IEI ↑ to IEO ↑ Delay (After ED Decode)	—	—	150	ns
75	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	ns
76	TdC (INT)	Clock ↑ to $\overline{INT}$ ↓ Delay	—	—	150	ns
77	TdIO (W/RWf)	$\overline{IORQ}$, $\overline{CE}$ (A_7 to A_2) ↓ to $\overline{W/RDY}$ ↓ Delay (Wait Mode)	—	—	175	ns
78	TdC (W/RRf)	Clock ↑ to $\overline{W/RDY}$ ↓ Delay (Ready Mode)	—	—	100	ns
79	TdC (W/RWz)	Clock ↓ to $\overline{W/RDY}$ Float Delay (Wait Mode)	—	—	110	ns
80	Th,Th (CS)	Any unspecified hold when setup is specified	0	—	—	ns
81	TwPh	Pulse width (High)	200	—	—	ns
82	TwPl	Pulse width (Low)	200	—	—	ns
83	TcTxC	$\overline{TxC}$ Cycle time	330	—	∞	ns
84	TwTxCl	$\overline{TxC}$ width (Low)	100	—	∞	ns
85	TwTxCh	$\overline{TxC}$ width (High)	100	—	∞	ns
86	TdTxC (TxD)	$\overline{TxC}$ ↓ to TxD Delay (x1 Mode)	—	—	220	ns
87	TdTxC (W/RRf)	$\overline{TxC}$ ↓ to $\overline{W/RDY}$ ↓ Delay (Ready mode)	5	—	9	CLK Periods
88	TdTxC (INT)	$\overline{TxC}$ ↓ to $\overline{INT}$ ↓ Delay	5	—	9	CLK Periods
89	TcRxC	$\overline{RxC}$ Cycle time	330	—	∞	ns

040489

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
90	TwRxCl	$\overline{RxC}$ width (Low)	100	—	∞	ns
91	TwRxCh	$\overline{RxC}$ width (High)	100	—	∞	ns
92	TsRxD (RxC)	RxD to $\overline{RxC}$ ↑ Setup time (x1 mode)	0	—	—	ns
93	ThRxD (RxC)	$\overline{RxC}$ ↑ to RxD hold time (x1 mode)	100	—	—	ns
94	TdRxC(W/RRf)	$\overline{RxC}$ ↑ to $\overline{W/RDY}$ ↓ Delay (Ready Mode)	10	—	13	CLK Periods
95	TdRxC (INT)	$\overline{RxC}$ ↑ to $\overline{INT}$ ↓ Delay	10	—	13	CLK Periods
96	TdRxC (SYNC)	$\overline{RxC}$ ↓ to $\overline{SYNC}$ ↑ Delay (Output modes)	4	—	7	CLK Periods
97	TsSYNC (RxC)	$\overline{SYNC}$ ↓ to $\overline{RxC}$ ↑ Setup Time (External syncmodes)	−100	—	—	ns
98	TsAdd (Cr)	Address Setup time to clock ↑	150	—	—	ns
99	TsIO (Cr)	$\overline{IORQ}$ ↓ Setup time to clock ↑	70	—	—	ns
100	TdRD (Cr)	$\overline{RD}$ ↓ Setup time to clock ↑	70	—	—	ns
101	TdCr (Do)	Data out Delay to clock ↑	—	—	130	ns
102	TdIORDr (DoZ)	Data Float Delay to $\overline{IORQ}$ ↑ $\overline{RD}$ ↑	—	—	90	ns
103	TsWR (Cr)	$\overline{WR}$ ↓ Setup time to clock ↑	70	—	—	ns
104	TsDI (Cr)	Data Input Setup time to clock ↑	0	—	—	ns
105	TdIOWRf (D)	Data Hold time to $\overline{IORQ}$, $\overline{WR}$ ↑	20	—	—	ns

040489

4.5.4 AC Characteristics of WDT (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C013AT-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
106	TdC (WDTf)	Clock ↑ to $\overline{WDTOUT}$ ↓ Delay	—	—	160	ns
107	TdC (WDTr)	Clock ↑ to $\overline{WDTOUT}$ ↑ Delay	—	—	165	ns
108	TCWDTZ	$\overline{WDTOUT}$ out put period				
		WDT Mode 0	—	$T12*2^{16}$	—	ns
		WDT Mode 1	—	$T12*2^{18}$	—	ns
		WDT Mode 2	—	$T12*2^{20}$	—	ns
		WDT Mode 3	—	$T12*2^{22}$	—	ns

040489

Note1 : Timing Measurements are made at the following voltage.

Input VIH = 2.4V, VIL = 0.4V
VIHC = VCC-0.6V, VILC = 0.6V
Output VOH = 2.2V, VOL = 0.8V (Except CLKOUT)
CL = 100PF

4.6 AC TIMING CHARTS (2) (IN INACTIVE STATE)

4.6.1 AC Timing Charts of CGC (in Inactive State)

The following Figures show the timing charts in each operation mode with the CLKOUT pin connected to the CLKIN pin.

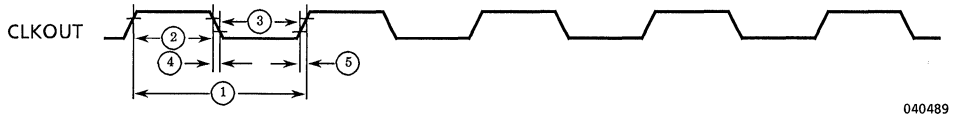


Figure 4.6.1 CLKOUT Waveform

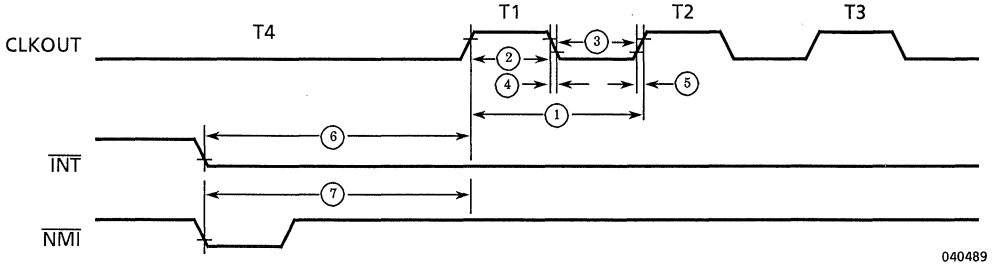


Figure 4.6.2 Clock Restart Timing (STOP mode)

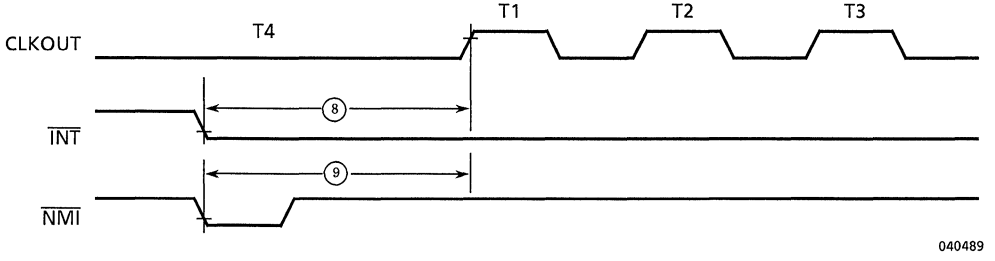


Figure 4.6.3 Clock Restart Timing (IDLE1 mode)

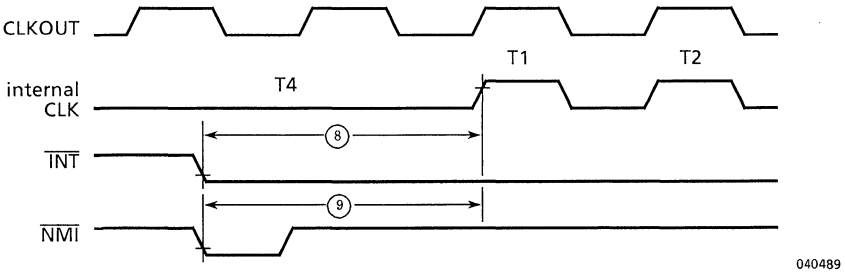


Figure 4.6.4 Clock Restart Timing (IDLE2 mode)

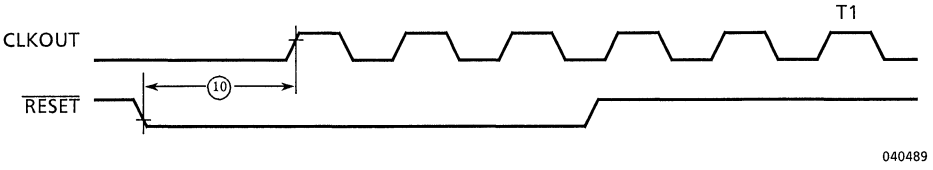


Figure 4.6.5 Timing of Clock Restart by $\overline{\text{RESET}}$ (IDLE1, IDLE2 mode)

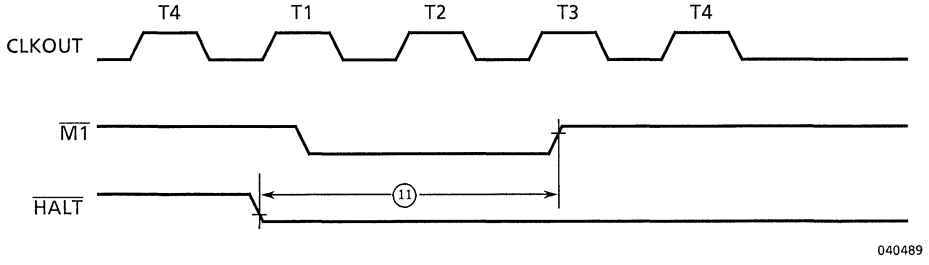


Figure 4.6.6 Clock Suspension Timing (IDLE1, IDLE2 and STOP modes)

4.6.2 AC Timing Charts of CTC (in Inactive State)

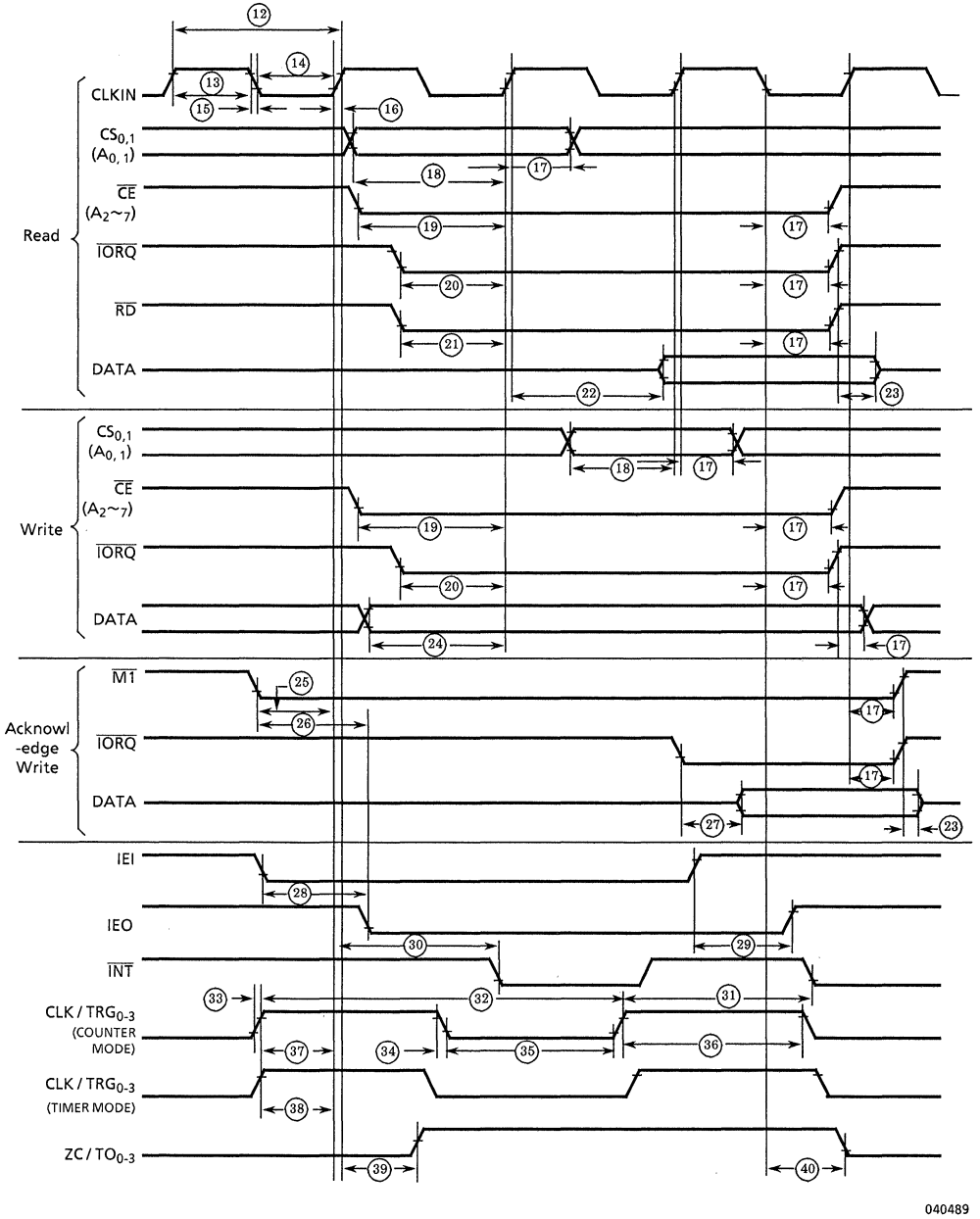


Figure 4.6.7 CTC Timing Diagram (Inactive)

4.6.3 AC Timing Charts of SIO (in Inactive State)

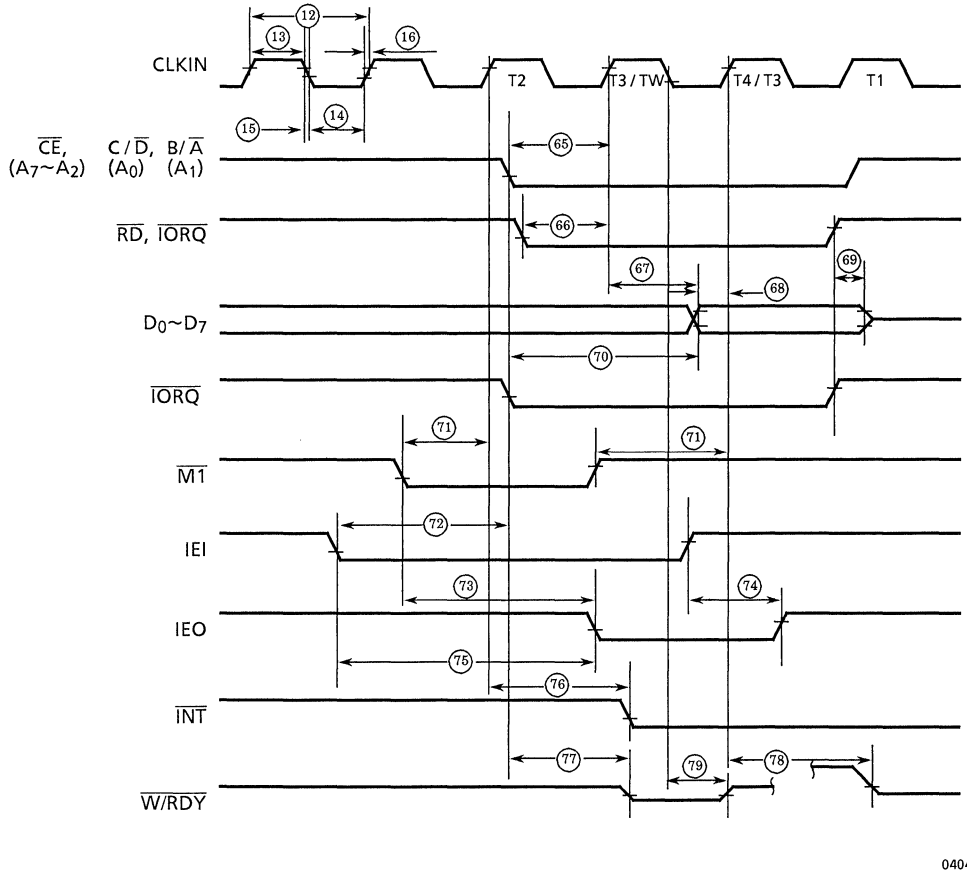


Figure 4.6.8 SIO Timing Diagram (a) (Inactive)

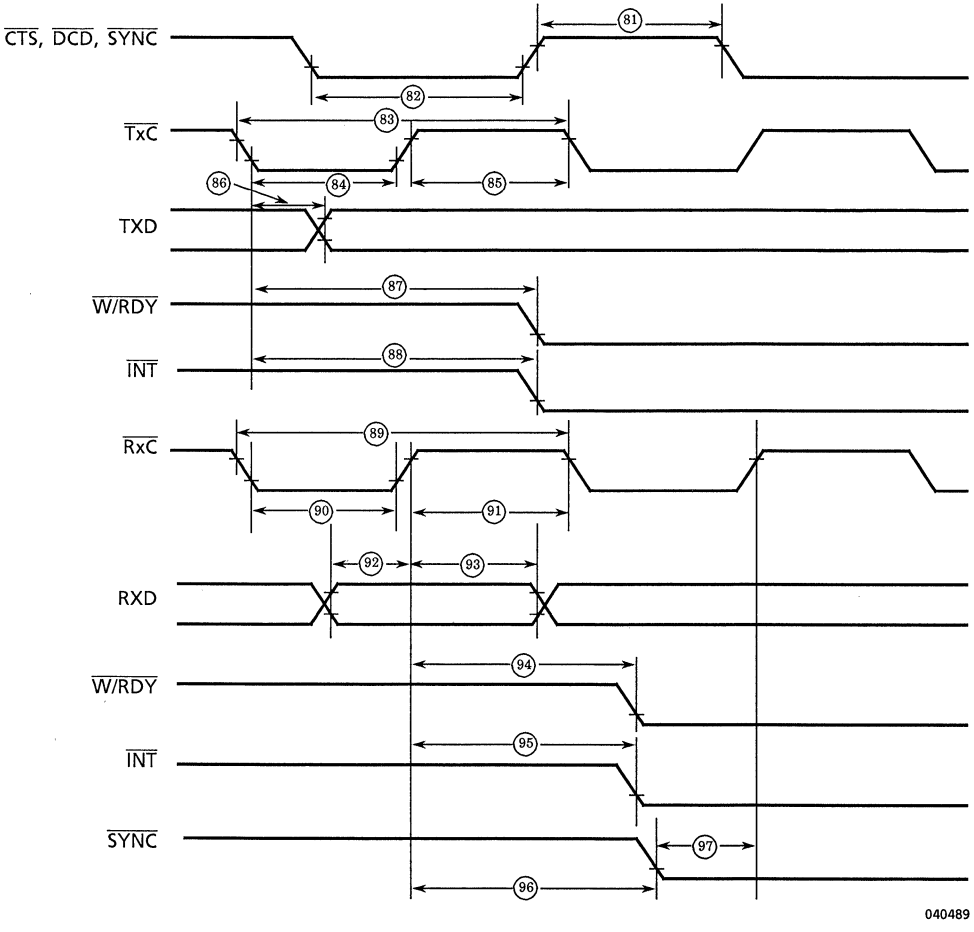


Figure 4.6.8 SIO Timing Diagram (b) (Inactive)

4.6.4 AC Timing Charts of WDT (in Inactive State)

(The mode setting and daisy chain interrupt setting registers on WDT)

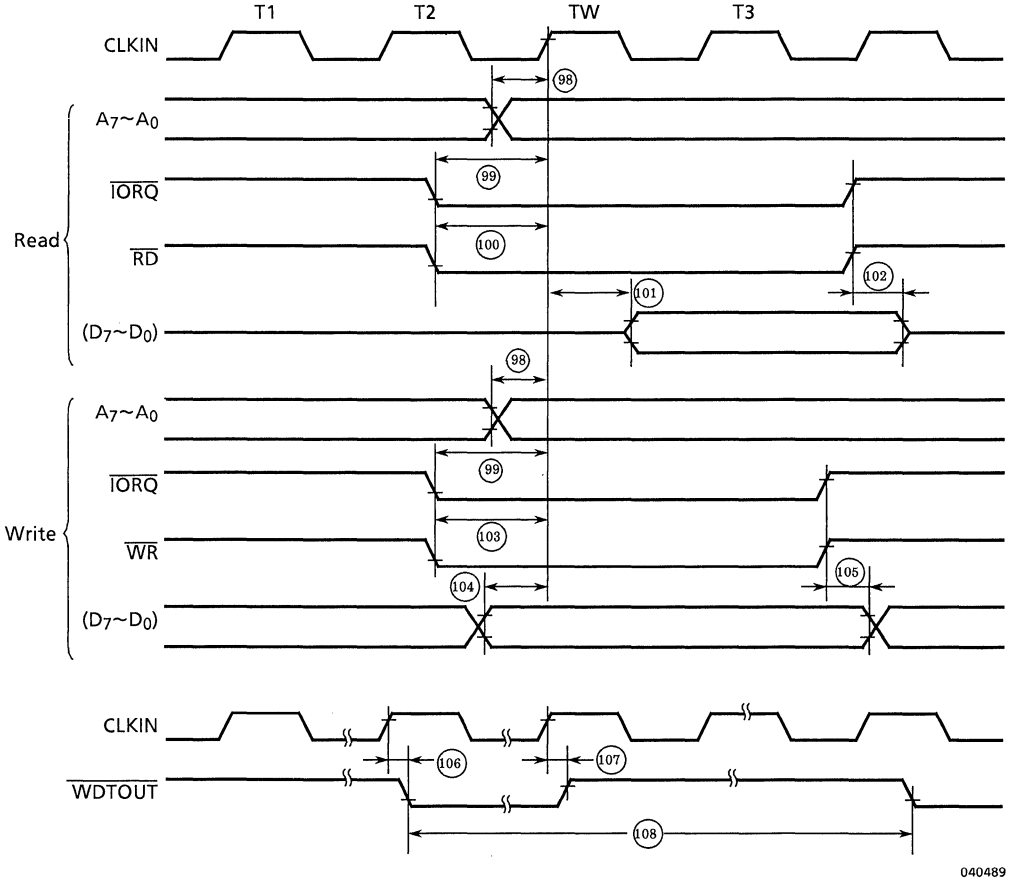


Figure 4.6.9 RD/WRITE, $\overline{\text{WDTOUT}}$ Timing Diagram

4.7 PIN CAPACITANCE

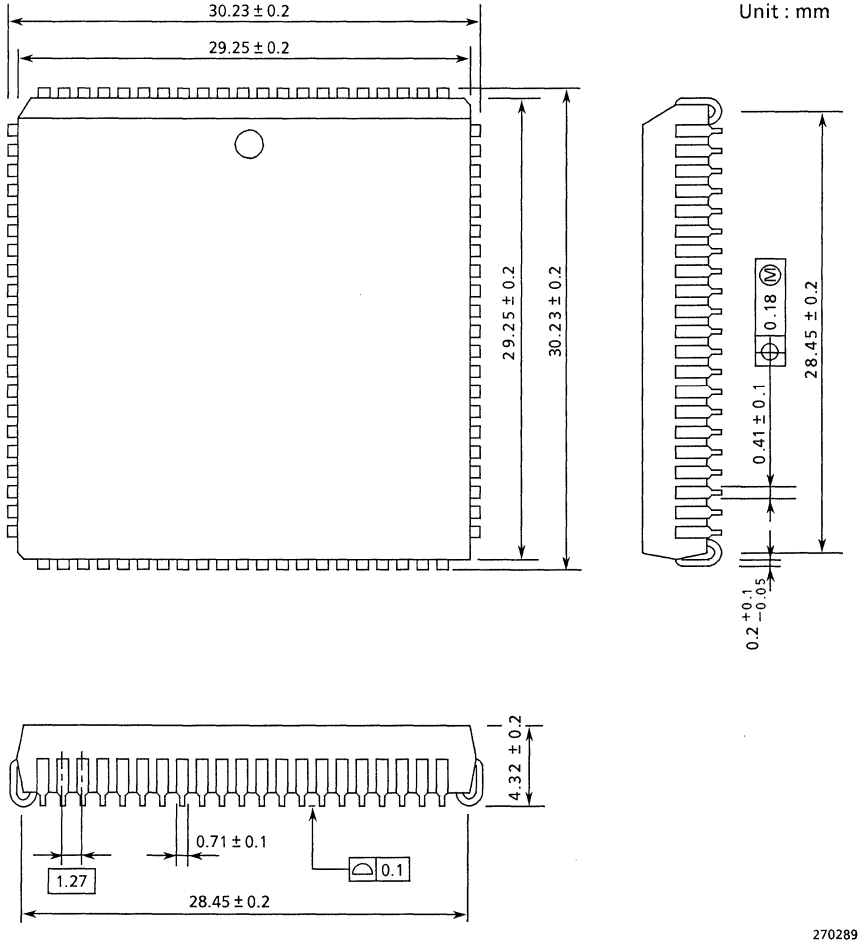
SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CCLOCK	Clock Input Capacitance	F = 1MHz	–	–	8	PF
CIN	Input Capacitance	All terminals except that to	–	–	6	
COUT	Output Capacitance	be measured be earthed	–	–	10	

040489

5. EXTERNAL DIMENSIONS

QFJ84-P-S115

Unit : mm



270289

TMPZ84C015AF-6 / TMPZ84C015AF-8
TLCS-Z80 MICROPROCESSOR

1. OUTLINE AND FEATURES

The TMPZ84C015A is a high-performance CMOS 8-bit microprocessor incorporating the counter timer circuit (CTC), the serial I/O port (SIO), the parallel I/O port (PIO), the clock generator/controller (CGC), and the watchdog timer (WDT) around the TLCS-Z80 MPU. This microprocessor inherits the basic architecture of the TLCS-Z80 series, allowing the user to utilize the software resources and development tools accumulated so far.

The TMPZ84C015A is based on the new CMOS process and housed in a standard 100-pin mini-flat package, greatly contributing to system miniaturization and power saving.

The TMPZ84C015A incorporates the high-performance serial I/O port, the counter timer circuit which can be used as the baud rate generator, and the watchdog timer indispensable for control applications, offering the user a wide range of system applications such as the communication controllers including a communication adaptor and the various other controllers which need miniaturization.

Features

- Built-in TLCS-Z80 series MPU, CTC, SIO, PIO,CGC and watchdog timer features.
- High speed operation (6MHz and 8MHz operations)
- Built-in clock generator (CGC: Clock Generator Controller)
- Built-in standby capability (with the controller built in) provides 4 operation modes:

  Run mode (Normal operation)
  Idle-1 mode (Only clock oscillation goes on.)
  Idle-2 mode (Wake-up by CTC enabled.)
  Stop mode (Clock oscillation stopped; standby state)

- Wide operational voltage range (5V±10% : 6MHz VERSION, 5V±5% : 8MHz VERSION) supported.
- Wide operating temperature range (−40°C to +85°C : 6MHz VERSION, −10°C to +70°C : 8MHz VERSION)
- Low power dissipation

  In operation : (RUN mode) 25mA TYP. at 6 MHz, 35mA TYP. at 8 MHz
  In idle : (IDLE-1 mode) 1.5mA TYP. at 6 MHz, 2mA TYP. at 8 MHz
  (IDLE-2 mode) 11mA TYP. at 6 MHz, 15mA TYP. at 8 MHz
  In standby : (STOP mode) 500nA TYP.

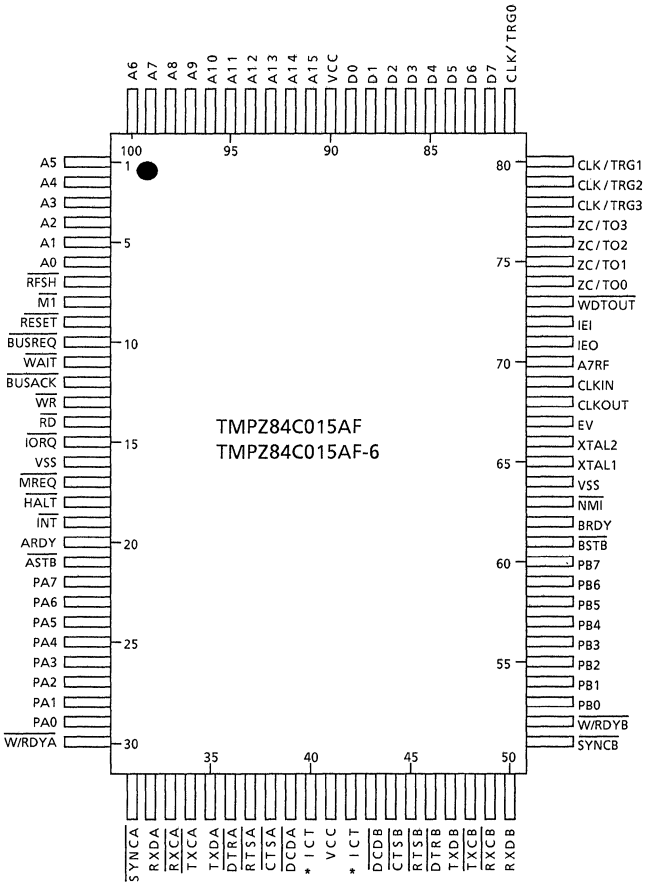
- Built-in TLCS-Z80 series SIO capability
- A pair of independent full duplex channels supports the asynchronous as well as synchronous byte-oriented (monosync and bisync) and bit-oriented HDLC and CCITT-X. 25 protocols.
- Built-in CRC generation and check capability.
- Data transfer rates of up to 1200 K bits/sec (6 MHz) and 1800 K bits/sec (8 MHz)
- Built-in TLCS-Z80 series CTC capability
  Four independent built-in channels.
  The timer or counter modes can be set .
  Also available as the SIO baud rate generator.
- Built-in TLCS-Z80 series PIO capability
  Two programmable independent 8-bit I/O ports having handshaking capability
  One of 4 operation modes can be selected for each port by using the program:

  Mode 0 (byte output mode)
  Mode 1 (byte input mode)
  Mode 2 (byte input/output mode)
  Mode 3 (bit mode)
- Built-in watchdog timer
- Programmed daisy-chain interrupt control
- Built-in dynamic RAM refresh controller
- TTL/CMOS compatible
- Housed compact standard 100-pin mini-flat package
- The Toshiba real time emulator (RTE80) and the Z80 ICE commercially available can be used (the TMPZ84C015A used as the evaluator).
- The Toshiba evaluator board installed.

note : Z80 is a trade mark of Zilog Inc.

2. PIN ASSIGNMENT AND FUNCTIONS

2.1 PIN ASSIGNMENTS (TOP VIEW)



170489

Note : The ICT pin is the test pin. Do not make anyexternal connection to this pin.

Figure 2.1.1 Pin Assignments

2.2 (A) PIN FUNCTIOS

(1/6)

Pin	Q'ty (Number)	Type	Function
D0-D7	8 (82-89)	Input/output 3-state	The 8-bit bi-directional data bus.
A0-A15	16 (91-100) (1-6)	Output 3-state	The 16-bit address bus. These pins specify memory and I/O port addresses. During a refresh cycle, the refresh address is output to the low-order 7 bits and A7RF.
$\overline{M1}$	1 (8)	Output 3-state	The Machine Cycle 1 signal. In an operation code fetch cycle, this pin goes "0" with the $\overline{MREQ}$ signal. At the execution of a 2-byte operation code, this pin goes "0" for each operation code fetch. In a maskable interrupt acknowledge cycle, this pin goes "0" with the $\overline{IORQ}$ signal. When the EV signal is applied, this pin is put in the high-impedance state.
$\overline{RD}$	1 (14)	Output 3-state	The Read signal. It indicates that the MPU is ready for accepting data from memory or I/O device. The data from the addressed memory or I/O devices is gated by this signal onto the MPU data bus. When the $\overline{BUSREQ}$ signal is applied, this pin is put in the high-impedance state.
$\overline{WR}$	1 (13)	Output 3-state	The Write signal. This signal is output when the data to be stored in the addressed memory or I/O device is on the data bus. When the $\overline{BUSREQ}$ signal is applied, this pin is put in the high-impedance state.
$\overline{MREQ}$	1 (17)	Output 3-state	The Memory Request signal. When the execution address for memory access is on the address bus, this pin goes "0". During a memory refresh cycle, this pin also goes "0" with $\overline{RFSH}$ signal.
$\overline{IORQ}$	1 (15)	Output 3-state	The Input/Output Request signal. This pin goes "0" when the address for an I/O read or write operation is on the low-order 8 bits (A0 through A7) of the address bus. The $\overline{IORQ}$ signal is also output with the $\overline{M1}$ signal at interrupt acknowledge to tell an I/O device that the interrupt response vector can be placed on the data bus. Note that the interrupt priority among the TMPZ84C015A CTC, and SIO is selected by a program.

170489

(2/6)

Pin	Q'ty (Number)	Type	Function
IEO	1 (71)	Output	The Interrupt Enable Output signal. In the daisy chain interrupt control, this signal controls the interrupt from the peripheral LSIs connected next to the TMPZ84C015A. The IEO pin goes "1" only when the IEI pin is "1" and the MPU is not servicing an interrupt from the built-in peripheral LSI.
XTAL1 XTAL2	2 (65) (66)	Input Output	The crystal oscillator connection. Connects an oscillator having the oscillation frequency 2 times as high as the system clock (CLKOUT) frequency.
CLKIN	1 (69)	Input	The Single-phase Clock Input. When the clock input is placed in the DC state (continued "1" or "0" level), this pin stops operating and holds the state of that time. Normally, this pin is connected with the CLKOUT pin. However, to operate the system with the external clock, connect the external clock to the CLKIN pin.
CLKOUT	1 (68)	Output	The Single-phase Clock Output. When a Halt instruction is executed in the Stop or Idle-1 mode, the CLKOUT output is retained at "0". In the Run and Idle-2 mode the clock is kept output. This pin provides the clock to other peripheral ICs.
$\overline{\text{RESET}}$	1 (9)	Input	The Reset signal input. This signal resets the internal states of the TMPZ84C015A. This signal is also used to return from the standby state in the Stop or Idle mode.
$\overline{\text{INT}}$	1 (19)	Input	The Maskable Interrupt signal. An interrupt is caused by the internal CTC, SIO PIO or the peripheral LSI. An interrupt is acknowledged when the interrupt enable flip-flop (IFF) is set to "1" by software. The $\overline{\text{INT}}$ pin is normally wire-ORed and requires an external pullup resistor for these applications. This signal is also used to return from the stand-by state in the Stop or Idle mode.
$\overline{\text{WAIT}}$	1 (11)	Input	The Wait Request signal. This signal indicates the MPU that the addressed memory or I/O device is not ready for data transfer. As long as this signal is "0", the MPU is in the Wait state.

170489

(3/6)

Pin	Q'ty (Number)	Type	Function
$\overline{BUSREQ}$	1 (10)	Input	The Bus Request signal. The $\overline{BUSREQ}$ signal forces the MPU address bus, data bus, and control signals $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ to be placed in the high-impedance state. This signal is normally Wire-ORed and requires an external pullup resistor for these applications.
$\overline{BUSACK}$	1 (12)	Output	The Bus Acknowledge signal. In response to the $\overline{BUSREQ}$ signal, the $\overline{BUSACK}$ signal indicates to the requesting peripheral LSI that the MPU address bus, data bus, and control signals $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$ have been put in the high-impedance state.
$\overline{HALT}$	1 (81)	Output 3-state	The Halt signal. This pin goes "0" when the MPU has executed a Halt instruction and is in the Halt state. It is put in the high-impedance state when the EV signal is applied.
$\overline{RFSH}$	1 (7)	Output	The refresh signal. When the dynamic memory refresh address is on the low-order 8 bits of the address bus, this signal goes "0" . At the same time, the $\overline{MREQ}$ signal also goes active ("0"). This pin is put in the high-impedance state when the EV signal is applied.
CLK/TRG0 ~CLK/TRG3	4 (78-81)	Input	The external clock/timer trigger. These 4 CLK/TRG pins correspond to 4 channels. In the counter mode, the down counter is decremented by 1 and in the timer mode, the timer is activated at each active edge (a rising or falling edge) of the signal which are input by these pins. It can be selected by program whether the active edge is a rising or falling edge.
ZC/TO0 ~ZC/T03	4 (74-77)	Output	The Zero Count/Timer Out signal. In either the Timer mode, or counter mode, pulses are output from these pins when the down counter has reached zero.
IEI	1 (72)	Input	The Interrupt Enable Input signal. This signal is used with the IEO to form a priority daisy chain when there is more than one interrupt-driven peripheral LSI.
$\overline{NMI}$	1 (63)	Input	The Non-maskable Interrupt Request signal. This interrupt request has a higher priority than the maskable interrupt and is not dependent on the interrupt enable flip-flop (IFF) state. This signal is also used to return from the stand-by state in the Stop or Idle mode.

170489

(4/6)

Pin	Q'ty (Number)	Type	Function
EV	1 (67)	Input	The Evaluator signal. When this signal is active, the $\overline{M1}$, $\overline{HALT}$, and $\overline{RFSH}$ pins are put in the high-impedance state. In using the TMPZ84C015A as an evaluator chip, the MPU is electrically disconnected (put in the high-impedance state) after one machine cycle is executed with the EV signal being "1" and the $\overline{BUSREQ}$ signal being "0", and follows the instructions from other MPU (such as the MPU of ICE). The signals of the disconnected MPU are A00 through A15, D0 through D7, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$, $\overline{M1}$, $\overline{HALT}$, and $\overline{RFSH}$. $\overline{BUSACK}$ needs to be disconnected by an externally connected circuit. The evaluator board is available to use the TMPZ84C015A as an evaluator chip.
A7RF	1 (70)	Output	The 1-bit auxiliary address bus. This pin outputs the same signal as the bit 7 (A7) of the address bus. However, during a refresh cycle, this pin outputs the address which is the most significant bit of the 8-bit refresh address signal linked to the low-order 7 bits of the address bus.
$\overline{ASTB}$	1 (21)	Input	The Port A Strobe Pulse From Peripheral Device. This signal is used at the handshaking between port A and external circuits. The meaning of this signal depends on the mode of operation selected for port A. (See "PIO Basic Timing".)
$\overline{BSTB}$	1 (61)	Input	The Port B Strobe Pulse From Peripheral Device. This signal is used at the handshaking between port B and external circuits. The meaning of this signal is the same as the $\overline{ASTB}$ signal except when port A is in the mode 2. (See "PIO Basic Timing".)
ARDY	1 (20)	Output	The Register A Ready signal. This signal is used at the handshaking between port A and external circuits. The meaning of this signal depends on the mode of operation selected for port A. (See "PIO Basic Timing".)
BRDY	1 (62)	Output	The Register B Ready signal. This signal is used at the handshaking between port B and external circuits. The meaning of this signal is the same as the ARDY signal except when port A is in the mode 2. (See "PIO Basic Timing".)
PA0-PA7	8 (22-29)	Input/Output 3-state	The Port Data A signal. These signals are used for data transfer between port A and external circuits.

170489

(5/6)

Pin	Q'ty (Number)	Type	Function
PB0-PB7	8 (53-60)	Input/Output 3-state	The Port Data B signal. These signals are used for data transfer between port B and external circuits.
$\overline{W/RDYA}$ $\overline{W/RDYB}$	2 (30, 52)	Output	The Wait/Ready signal A and the Wait/Ready signal B. These signals can be used as the Wait or Ready depending on SIO programming. When these signals are programmed as "Wait", they go active at "0" to indicate to the MPU that the addressed memory or I/O devices are not ready for data transfer, requesting the MPU to wait. When these signals are programmed as "Ready", they go active at "0" to determine when a peripheral device associated with a DMA port is ready for a read or write data. The DMA is requested to transfer data.
$\overline{SYNCA}$ $\overline{SYNCB}$	2 (31, 51)	Input/Output	The Synchronization signal. In the asynchronous receive mode, these signals act as the $\overline{CTS}$ and $\overline{DCD}$ signals. In the external sync mode, these signals act as inputs and in the internal sync mode, they act as outputs.
RXDA RXDB	2 (32, 50)	Input	The Serial Receive Data signal.
$\overline{RXCA}$ $\overline{RXCB}$	2 (33, 49)	Input	The Receive Clock signal. In the asynchronous mode, the Receive Clocks may be 1, 16, 32 or 64 times the data transfer rate.
$\overline{TXCA}$ $\overline{TXCB}$	2 (34, 48)	Input	The Transmitter Clock signal. In the asynchronous mode, the Transmitter Clocks may be 1, 16, 32, or 64 times the data transfer rate.
TXDA TXDB	2 (35, 47)	Output	The serial transmit data signal.
$\overline{DTRA}$ $\overline{DTRB}$	2 (36, 46)	Output	The Data Terminal Ready signal. These signals indicate whether the data terminal is ready to receive serial data or not. When it is ready, these signals go active to enable the transmitter of the terminal. When it is not ready, these signals go inactive to disable the transfer from the terminal.
$\overline{RTSA}$ $\overline{RTSB}$	2 (37, 45)	Output	The Request to Send signal. These pins are "0" when transmitting serial data. That is, to transmit data, these signals are made active to enable their receivers.

170489

(6/6)

Pin	Q'ty (Number)	Type	Function
$\overline{CTSA}$ $\overline{CTSB}$	2 (38, 44)	Input	The Clear To Send signal. When these pins are "0", the modem having transmitted these signals is ready to receive serial data. When it is ready, these signals go active to enable the transmitter of the terminal. When it is not ready, these signals go inactive to disable the transfer from the terminal.
$\overline{DCDA}$ $\overline{DCDB}$	2 (39, 43)	Input	The Data Carrier Detect signal. When these pins are "0" the receive of serial data can be enabled. That is, to transmit data, these signals are made active to enable their receivers.
ICT	2 (40, 42)	Output	The test pins. To be used in the open state.
$\overline{WDTOUT}$	1 (73)	Output	The Watchdog Timer signal. The output pulse width depends on the externally connected pin.
VCC	2 (41, 90)		The power supply (+ 5 V) pins.
VSS	1 (16, 64)		The ground (0 V) pins.

170489

2.2 (B) TMPZ84C015A Internal I/O Address Map

Internal I/O	Channel	I/O Address
CTC (Counter Timer)	ch 0	# 10
	ch 1	# 11
	ch 2	# 12
	ch 3	# 13
SIO (Serial I/O)	ch A Send/Receive buffer	# 18
	ch A Command/Status Register	# 19
	ch B Send/Receive buffer	# 1A
	ch B Command/Status Register	# 1B
PIO (Parallel I/O)	A Port Data	# 1C
	A Port Command	# 1D
	B Port Data	# 1E
	B Port Command	# 1F
Watch Dog Timer Stand-by mode Register	WDTER, WDTPR, HALTMR	# F0
Watch Dog Timer command Register	Clear command ($4E_H$) disable command ($B1_H$)	# F1
Daisy-chaine interrupt precedence Register	Only use bit2~bit0	# F4

180489

3. OPERATIONAL DESCRIPTION

3.1 BLOCK DIAGRAM AND OPERATIONAL OUTLINE

3.1.1 Block Diagram

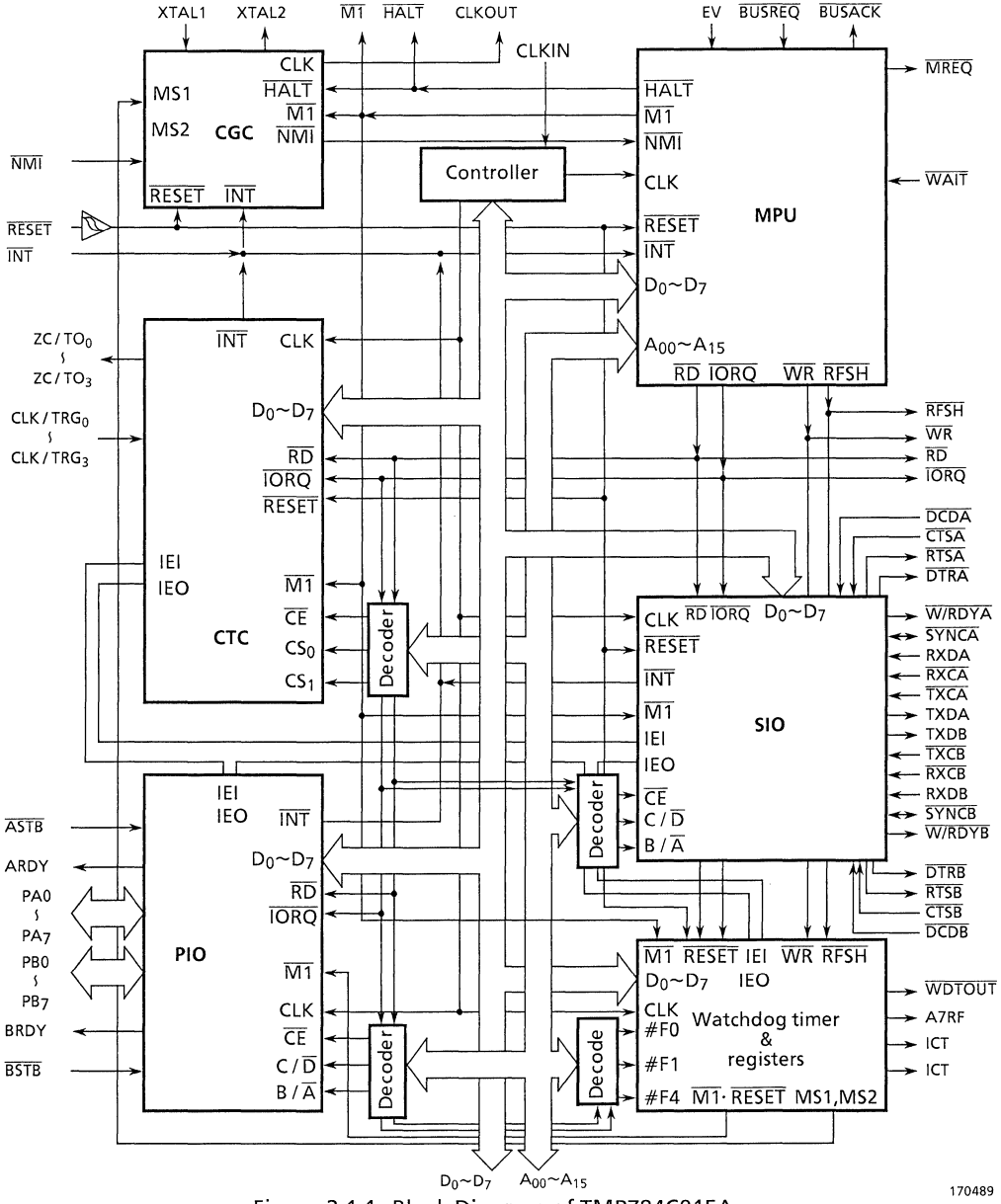


Figure 3.1.1 Block Diagram of TMPZ84C015A

3.1.2 Operational Outline

The TMPZ84C015A largely consists of a processor (MPU), a counter/timer circuit (CTC), a serial input/output controller (SIO), a parallel input/output controller (PIO), a watchdog timer (WDT), and a clock generator/controller (CGC).

- The MPU provides all the capabilities and pins of the Toshiba TLCS-Z80 MPU (TMPZ84C00A) to play the role of the TLCS-Z80 microprocessor perfectly.
- The CTC provides the capabilities of the Toshiba TLCS-Z80 CTC (TMPZ84C30A) and has the pins required to perform the necessary operations as a TLCS-Z80 peripheral LSI. The four independent timer channels are I/O-addressed internally.
- The SIO provides the capabilities of the Toshiba TLCS-Z80 SIO (TMPZ84C43A) and has the pins required to perform the necessary operations as a TLCS-Z80 peripheral LSI. The two independent serial channels are I/O-addressed internally.
- The PIO provides the capabilities of the Toshiba TLCS-Z80 PIO (TMPZ84C20A) and has the pins required to perform the necessary operations as a TLCS-Z80 peripheral LSI. The two independent parallel ports are I/O-addressed internally.
- The WDT incorporates one-channel watchdog timer and the read/write-enabled watchdog timer control registers indispensable for control applications. The WDT also has the register to determine interrupt priorities, allowing the user to set the daisy-chain interrupt priorities by program. Additionally, the WDT has the IEI and IEO pins required to process the daisy- chain interrupts caused by the peripheral LSIs to be added both inside and outside the TMPZ84C015A.
- The CGC provides the four operation modes to control the entire TMPZ84C015A chip; the Run, Idle-1, Idle-2, and Stop modes. They are program-selectable. This chip has two clock pins: CLKOUT to supply clock from the CGC and CLKIN to get clock from the outside. Therefore, the TMPZ84C015A can be operated on the clock supplied from the outside at the CLKIN pin without using the CGC. The following briefly describes the four operation modes of the CGC with the CLKOUT and CLKIN pins connected:
- In the Run mode, the clock generated by the CGC is supplied to the TMPZ84C015A and peripheral LSIs to perform the normal programmed microcomputer operations.
- In the Idle-1 mode, clock oscillation is going on but the clock is not supplied to the TMPZ84C015A and peripheral LSI, thereby saving the system power consumption and shortening the time required for system restart.

- In the Idle-2 mode, clock oscillation is performed and the clock is output from the CLKOUT pin. The clock is supplied only to the CTC in the TMPZ84C015A, permitting a wake-up operation by the CTC. Like the Idle-1 mode, the Idle-2 mode saves the system power consumption and shortens the time required for system restart.
- In the Stop mode, clock oscillation is not performed and the system operation can be stopped completely. In this mode, the system can be restarted with the internal data retained with an extremely low power consumption level unique to the CMOS technology.

Note that these modes can be set only when the MPU has executed a HALT instruction.

Additionally, the TMPZ84C015A has also the EV pin which is used with the $\overline{\text{BUSREQ}}$ pin to put the MPU in the high-impedance state for electrical disconnection, thus functioning as an evaluator chip. That is, the MPU in the TMPZ84C015A is electrically disconnected by these two pins to implement the emulation by the signal from the in-circuit emulator (ICE). For emulation, one machine cycle is performed on the MPU in the TMPZ84C015A with EV being "1" and the $\overline{\text{BUSREQ}}$ being "0" then the emulation is performed as instructed by the MPU. The MPU signals to be disconnected are A00 through A15, D0 through D7, $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$, $\overline{\text{RD}}$, $\overline{\text{M1}}$, $\overline{\text{HALT}}$, and $\overline{\text{RFSH}}$, $\overline{\text{BUSACK}}$ needs to be disconnected by an externally connected circuit.

3.2 MPU OPERATIONS

This subsection describes the system configuration, functions and the basic operation of the MPU of the TMPZ84C015A.

3.2.1 Block Diagram

Figure 3.2.1 shows the block diagram of the MPU.

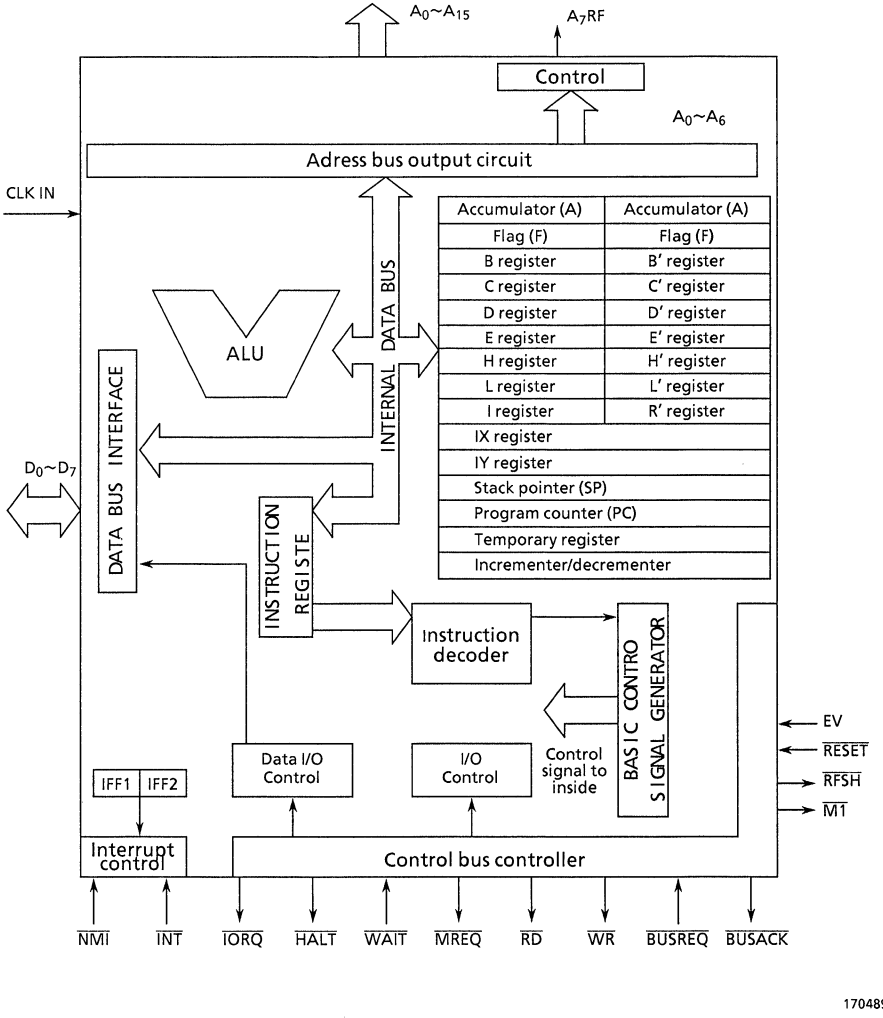


Figure 3.2.1 MPU Block Diagram

3.2.2 MPU System Configuration

The MPU has the configuration shown in Figure 3.2.1. The address signal is put on the address bus via the address buffer. The data bus is controlled for input or output by the data bus interface. Both the address and data buses are put in the high-impedance state by the $\overline{\text{BUSREQ}}$ signal input to make them available for other peripheral LSIs. The Opcode read from memory via the data bus is written to the instruction register. This Opcode is decoded by the instruction decoder. According to the result of the decoding, control signals are sent to the relevant devices. Receiving these control signals, the ALU performs various arithmetic operations. The register array temporarily hold the information required to perform operation.

The following describes the MPU's main components and functions which the user must understand to operate the TMPZ84C015A:

[1] Internal Register Groups

The configuration of the internal register groups is as follows:

(1) Main registers

A, F, B, C, D, E, H, L

(2) Alternate registers

A', F', B', C', D', E', H', L'

(3) Special purpose registers

I, R, IX, IY, SP, PC

Figure 3.2.3 shows the configuration of the internal register groups. The register groups, each being of a static RAM, consists of eighteen 8-bit registers and four 16-bit registers. The following describes the function of each register:

(1) Main registers (A, F, B, C, D, E, H, L)

(a) Accumulator (A)

The accumulator is an 8-bit register used for arithmetic and data transfer operations.

(b) Flag register (F) (see Figure 3.2.2)

The flag register is an 8-bit register to hold the result of each arithmetic operation. Actually, the 6 of the 8 bits are set("1")/reset("0") according to the condition specified by an instruction.

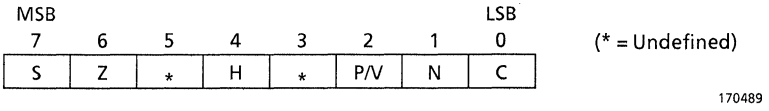


Figure 3.2.2 Flag Register Configuration

The following 4 bits are directly available to the programmer for setting the jump, call and return instruction conditions:

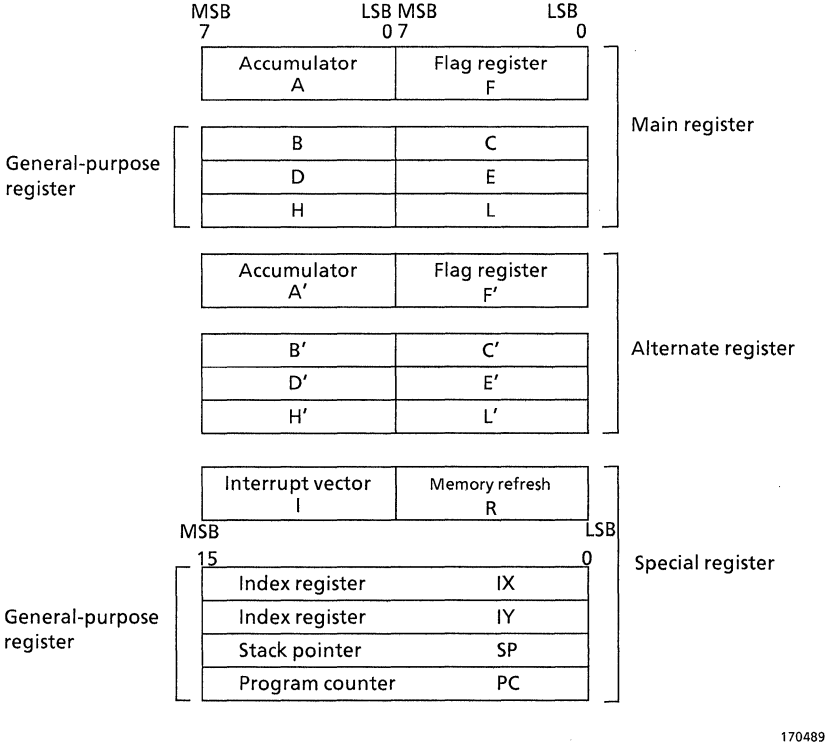


Figure 3.2.3 Flag Register Configuration

- Sign flag (S)

  When the result of an operation is negative, the S flag is set to "1" . Actually, the content of bit 7 of accumulator is stored in this flag.

- Zero flag (Z)

  When all bits turn out to be "0" s after operation, the Z flag is set to "1" . Otherwise, it is set to "0" . With a block search instruction (CPI, CPIR, CPD or CPDR), the Z flag is set to "1" if the source data and the accumulator data match. With a block I/O instruction (INI, IND, OUTI or OUTD), the Z flag is set to "1" if the content of the B register used as the byte counter is "0" at the end of comparison.

- Parity/overflow flag (P/V)

  This flag has two functions. One is the parity flag (P) that indicates the result of a logical operation (AND A, B etc.). The P flag is set to "1" if the parity is even as a result of the operation on signed values by two's complement. It is reset to "0" if the parity is odd. With a block search instruction (CPI, CPIR, CPD or CPDR) and a block transfer instruction (LDI or LDD), the P flag indicates the state of the byte counter (register pair B and C). It is set to "1" if the byte counter is not "0" and reset to "0" when the byte counter becomes"0" (at the end of comparison or data transfer). The content of the interrupt enable flip-flop (IFF) is saved to the P flag when the contents of the R register or I register are transferred to the accumulator.

  The other use of the P/V flag is the overflow flag (V) that indicates whether an overflow has occurred or not as a result of an arithmetic operation. The V flag is set to "1" when the value in the accumulator gets out of a range of the maximum value $+127$ and the minimum value -128 and therefore cannot be correctly represented as a two's complement notation.

  Whether the P/V flag operates as the P flag or V flag is determined by the type of the instruction executed.

- Carry flag (c)

  The C flag is set to "1" if a carry occurs from bit 7 of the accumulator or a borrow occurs as a result of an operation.

  The following two flags are not available to the programmer for the test and set ("1)/reset ("0") purposes. They are internally used by the MPU for BCD arithmetic operations.

- Half carry flag (H)

  The H flag is used for holding the carry or borrow from the low-order 4 bits of a BCD operation result. When a DAA instruction (decimal adjust) is executed, the MPU automatically uses the H flag to adjust the result of a decimal addition or

subtraction.

- Add/subtract flag (N)

In BCD operation, algorithm is different between addition and subtraction. The N flag indicates whether the executed operation is addition or subtraction.

For change of the flag state depending on the instruction, see 3.2.4 "TMPZ84C015A Instruction Set".

(c) General-purpose registers (B, C, D, E, H, L)

General-purpose registers consist of 8 bits each. They are used as 16-bit register pairs (BC, DE, HL) as well as independent 8-bit registers to supplement the accumulator. The B register and the register pair BC are used as counters when a block I/O, block transfer, or search instruction is executed. The register pair HL has various memory addressing features as compared with the register pairs BC and DE.

(2) Alternate registers (A', F', B', C', D', E', H', L')

The configuration of the alternate register is exactly the same as that of the main registers. There is no instruction that handles the alternate registers directly. The data in the alternate registers are processed by moving them into the main registers by means of exchange instructions as shown below:

EX AF, AF' (A↔A', F↔F')

EXX (B↔B', C↔C', D↔D', E↔E', H↔H', L↔L')

When a hign-speed interrupt response has been requested within the system, these instruction can be used to quickly move the contents of the accumulator, flag registers, and general-purpose registers into the corresponding registers. This eliminates the need for transferring the register contents to/from the external stack during execution of the interrupt handling routine, thereby shortening the interrupt servicing time greatly.

(3) Special purpose registers (I, R, IX, IY, SP, PC)

(a) Interrupt page address register (I)

The TMPZ84C015A provides two kinds of interrupts: maskable interrupt (INT) and non-maskable interrupt (NMI). The maskable interrupt provides three modes (0, 1, and 2) in which the interrupt is handled. These modes can be selected by instructions IMO, IMI, and IM2 respectively. In Mode 2, any memory location can be called indirectly depending on the interrupt. For this purpose, the I register stores the high-order 8 bits of the indirect address. The low-order 8 bits are supplied from the interrupting peripheral LSI. This scheme permits calling the interrupt handling routine from any memory location in an extremely short access

time. For the details of interrupts, see [4] "Interrupt Capability".

(b) Memory refresh register (R)

The R register is used as the memory refresh counter when the dynamic RAM is used for memory. This permits using of the dynamic memory in the same manner as the static memory. This 8-bit register is automatically incremented for each instruction fetch. While the MPU decodes and executes the fetched instruction, the contents of the R register are synchronized with the refresh signal to place the low-order 7 bits and A7RF on the address bus. This operation is all performed by the MPU and, therefore, dose not need a special processing by program. The MPU operation is not delayed by this operation. During refresh, the contents of the I register are placed on the high-order 8 bits of the address bus.

(c) Index registers (IX, IY)

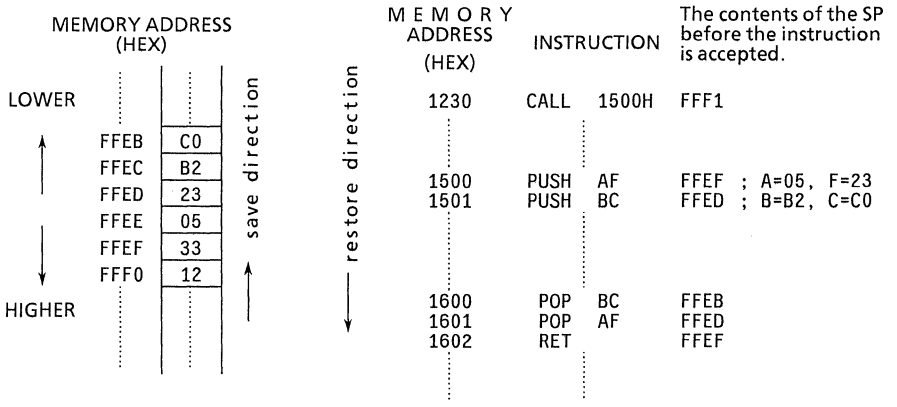
The two independent index registers IX and IY hold the 16-bit base address when used in the index addressing mode. In this addressing mode, the memory address obtained by adding the contents of an index register to the displacement value (for example, LD IX+40H) is specified. This mode is convenient for using data tables. Also these registers can be used separately for memory addressing and data retaining registers.

(d) Stack pointer (SP)

The stack pointer is a 16-bit register to provide the start address information in the stack area in the external RAM. The content of the stack pointer is decremented at the execution of a CALL instruction or PUSH instruction or interrupt handling and is incremented at the execution of a RET instruction or POP instruction. At the execution of a CALL instruction or interrupt handling, the current content of the program counter is saved into the stack. At the execution of a RET instruction, the content is restored from the stack to the program counter. These operations are all performed by the MPU automatically. However, the other registers are not saved or restored automatically. For the storing of the contents of these registers, an exchange instruction (EX or EXX) for alternate register, a PUSH or a POP instruction must be used. When a PUSH instruction is executed, the contents of the specified register are saved into the stack. When a POP instruction is executed, the contents of the stack are moved to the specified register.

These data are restored on a last-in, first-out basis. Use of the stack permits processing of multiple-level interrupts, deep subroutine nestings, and various data manipulation very easily. The stack pointer is not initialized in the hardware approach. Therefore, it is required to allocate the stack area in RAM to specify initialization (at the highest address of the stack area) in the initial program.

(ex)



The foregoing example shows the stack pointer and stack operations in which the instructions starting with the CALL at address 1230H and ending with the RET at address 1602H have been executed. However, it is assumed that there is no instruction or interrupt other than shown above that uses the stack during the execution. When the value of the stack pointer before executing the CALL instruction at address 1230H indicates address FFF1H, address 1233H is stored at addresses FFF0H and FFEFH because the CALL instruction consists of 3 bytes, then the stack pointer is decremented. Similarly, the data are saved or restored sequentially according to the instructions. These stack and stack pointer operations are all performed automatically.

(e) Program counter (PC)

The program counter holds, in 16 bits, the memory address of the instruction to be executed next. The MPU fetches the instruction from the memory location indicated by the program counter. When the content of the program counter is put on the address bus, the program counter is incremented automatically. However, it is not incremented with a jump instruction, a call instruction, or interrupt processing. Instead, the specified new address is set on it. With a return instruction, the content restored from the stack is set on the program counter. These operations are all performed automatically and therefore, no care is required for programming.

[2] Halt Capability

When a HALT instruction has been executed, the MPU is put in the halt state. The halt capability can be used to halt the MPU against the external interrupts, thereby reducing the power dissipation. In the halt state the states of MPU's internal registers are retained. The halt state is cleared by reset or when an interrupt is accepted.

For the details of halt operation, see [3] "Basic Timing."

(1) Halt operation

When a HALT instruction has been executed, the MPU sets the $\overline{\text{HALT}}$ signal to "0" to indicate that the MPU is going to be put in the halt state. Actually, the MPU in the halt state automatically continues executing NOP instructions if there is the system clock input. However, the program counter is not incremented. This keeps the refresh signal generated when the dynamic memory is used. During halt, the MPU's internal states are retained. The TMPZ84C015A contains the clock generator/controller, easily implementing the clock input control for these halt operations.

(2) Releasing the halt state

The halt state is cleared by accepting an interrupt (the $\overline{\text{INT}}$ or $\overline{\text{NMI}}$ signal input) or by reset (the $\overline{\text{RESET}}$ signal input) . When an interrupt is accepted, the halt state is cleared and the interrupt handling routine is executed. However, a maskable interrupt (INT) cannot be accepted unless the interrupt enable flip-flop (IFF) is set.

Note that when the halt state is cleared by the $\overline{\text{RESET}}$ signal, the MPU is reset and the program counter is set to "0".

[3] RESET Signal

Holding the $\overline{\text{RESET}}$ pin at the low level ("0") under the following conditions, the MPU's internal states are reset:

(1) The supply voltage level is within the operational voltage range.

(2) System clock stabilization.

(3) Holding the $\overline{\text{RESET}}$ signal at the low level ("0") for at least 3 full clock cycles. When the $\overline{\text{RESET}}$ signal goes high ("1") , the MPU starts executing instructions from address 0000H after at least 2T state dummy cycles.

When reset, the MPU performs the following processing:

(1) Program counter

0000H is set.

(2) Interrupt

The interrupt enable flip-flop (IFF) is reset to "0" to disable the maskable interrupt. For the maskable interrupt processing, mode 0 is specified.

(3) Control output

All control outputs are made inactive ("1") . Therefore, the halt state is also cleared.

(4) Interrupt page address register (I register)

The content of the R register becomes 00H.

(5) Refresh register (R register)

The content of the R register becomes 00H.

The contents of the registers other than above and the external memory do not change.
Therefore, they must be initialized as required.

[4] Interrupt Capability

The interrupt capability is used to suspend the execution of the current program and execute the processing of the requested peripheral LSI. Normally, this interrupt processing routine contains the data exchange and transfer of status and control information between the MPU and the peripheral LSI. When this routine has been completed, the MPU returns to the state active before the interrupt has been accepted.

The TMPZ84C015A provides the non-maskable interrupt (NMI) and maskable interrupt (INT) capabilities which are detected by the $\overline{NMI}$ and $\overline{INT}$ interrupt request signals, respectively. A non-maskable interrupt, when caused by a peripheral LSI, is accepted unconditionally. This interrupt is used to support critical functions such as the protection of the system from unpredictable happening including power failure. A maskable interrupt can be enabled or disabled by program. For example, if the timer is used and, therefore, an interrupt is not desired, the system can be programmed to disable the interrupt. Table 3.2.1 lists the processing by interrupt source.

(1) Interrupt enable/disable

A non-maskable interrupt cannot be disabled by program, while a maskable interrupt can be enabled or disabled by program. The MPU has the interrupt enable flip-flop (IFF) . A maskable interrupt can be enabled or disabled by setting this flip-flop to "1" (set) or "0" (rest) through an EI instruction (enable) or a DI instruction (disable) in program. Actually, the IFF consists of two flip-flops IFF1 and IFF2. IFF1 is used to select between the enable and disable of a maskable interrupt. IFF2 holds the state of IFF1 before a maskable interrupt has been accepted. Both IFF1 and IFF2 are reset to "0" when any of the following conditions occurs, disabling an interrupt:

- MPU reset
- Execution of DI instruction
- Acceptance of maskable interrupt

Both IFF1 and IFF2 are set to "1" when the the following condition occurs, enabling an interrupt:

- Execution of EI instruction

Actually, the waiting maskable interrupt request is accepted after the execution of the instruction that following the EI instruction.
This delay by one instruction is caused by accepting an interrupt after completion of the execution of a return instruction if the instruction following the EI instruction is a return instruction.
In the above operation, the contents of IFF1 and IFF2 are the same.

Table 3.2.1 Processing by Interrupt Source

Interrupt Source	Priority	Programmed condition		Vector addres	Interrupt return instruction
Non-maskable interrupt (the falling edge of $\overline{NMI}$)	1	None		Address 66H	RETN
Maskable interrupt ($\overline{INT}$ becomes "0" at instruction's last clock)	2	IFF = 1	Mode 0	Instruction from peripheral LSI. Normally, CALL or RST instruction.	(Note) RETI
			Mode 1	Address 38H.	
			Mode 2	The address indicated by the data table (memory) at the address specified by I register (high-order 8 bits) and data from peripheral LSI (low-order 8 bits, LSB = "0").	

Note : Mode 0 applies when the instruction from peripheral LSI is CALL or RST instruction.

170489

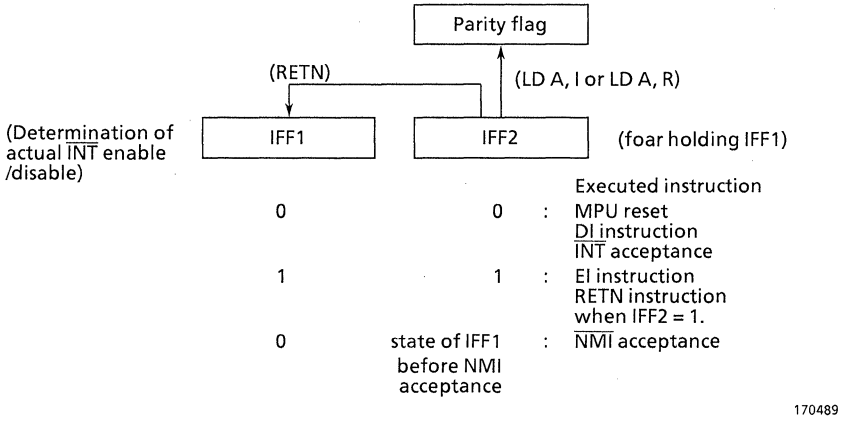


Figure 3.2.4 Interrupt Enable Flip-Flop (IFF)

When a non-maskable interrupt has been accepted, IFF1 is reset to "0" (interrupt disable) until an EI or RETN instruction is executed, so as to prevent from accepting the next interrupt. For this purpose, the state (interrupt enable/disable) of IFF1 immediately before non-maskable interrupt acceptance must be stored. This state is copied into IFF2 upon acceptance of a non-maskable interrupt. The content of IFF2 is copied into the parity flag at the execution of the following instructions, so that the copied data can be tested or stored:

- The load instruction (LD A, I) to load the contents of the I register into the accumulator.
- The load instruction (LD A, I) to load the contents of the R register into the accumulator.

When the return instruction (RETN) from the non-maskable interrupt is executed, the contents of the current IFF2 are copied back to IFF1. If an operation which changes the contents of IFF2 (due to the execution of EI or DI instruction, for example) has not been performed during interrupt handling, IFF1 automatically returns to the state immediately before the interrupt acceptance. Table 3.2.1 lists the states of IFF1 and IFF2 after execution of interrupt-related instructions.

Table 3.2.2 State of IFF1 and IFF2

Operation sequence	IFF1	IFF2	Remarks
MPU reset	0	0	
EI	1	1	
NMI acceptance	0	1	
LD A, I	*	*	Parity flag←IFF2
RETN	1	1	IFF1←IFF2
LD A, R	*	*	Parity flag←IFF2
INT acceptance	0	0	
RETI	*	*	
EI	1	1	
NMI acceptance	0	1	
DI	0	0	
RETN	*	*	

170489

Note : * = no change.

(2) Interrupt processing

With a non-maskable interrupt, the internal NMI flip-flop is set to "1" on the falling edge of the interrupt signal, $\overline{NMI}$. The state of this flip-flop is sampled on the rising edge of the last clock of each instruction to accept an interrupt. A maskable interrupt is accepted if the interrupt signal $\overline{INT}$ is low ("0") on the rising edge of the last clock of each instruction and the interrupt enable state (IFF = 1 and $\overline{BUSREQ}$ signal = inactive ("1")) is on. The following is the processing to be performed after a non-maskable interrupt and a maskable interrupt are accepted:

(a) Non-maskable interrupt (NMI)

When a non-maskable interrupt has been accepted, the MPU performs the following processing:

1 The internal NMI flip-flop is reset to "0".

2 IFF1 is reset to "0", disabling the maskable interrupt.

The contents of the IFF1 immediately before the interrupt acceptance are copied into the IFF2

3 The contents of the current program counter are saved into the stack.

4 The instructions starting from non-maskable interrupt vector address 66H are executed.

The non-maskable interrupt processing program terminates after executing the RETN instruction. This return instruction performs the following:

1 The contents of the current IFF2 are copied into IFF1.

2 The contents of the program counter are restored from the stack.

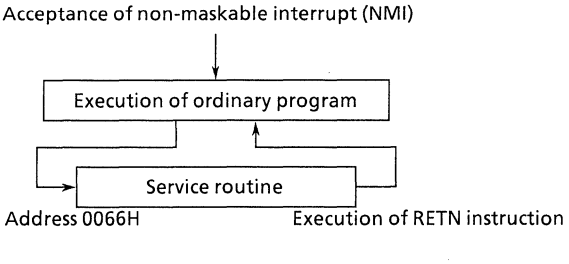


Figure 3.2.5 Non-Maskable Interrupt Processing

(b) Maskable interrupt (INT)

When a maskable interrupt has been accepted, the MPU performs the following processings:

1 Both IFF1 and IFF2 are reset to "0", disabling the maskable interrupts.

2 The contents of the current program counter are saved into the stack.

3 A maskable interrupt is serviced in one of the three modes 0, 1 and 2. A mode is selected by executing the instruction IM0, IM1 or IM2 before the interrupt is serviced. The instructions are executed starting from the vector address corresponding to the selected mode.

- Mode 0

In mode 0, the interrupting peripheral LSI puts a restart instruction (RST) or a call instruction (CALL) on the data bus and the MPU executes the interrupt service routine according to that instruction. At reset, this mode is automatically set.

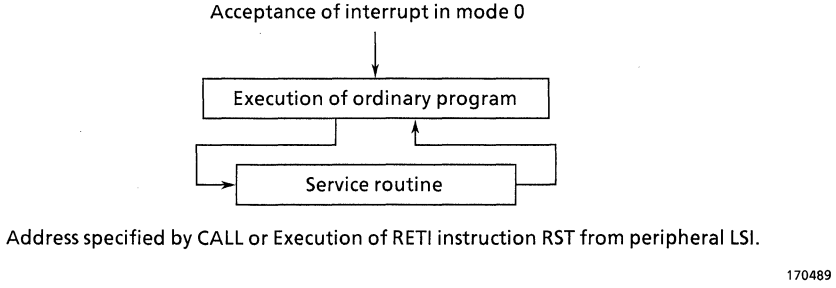


Figure 3.2.6 Interrupt Processing in Mode 0

- Mode 1

When an interrupt is accepted in mode 1, restart is performed from address 0038H. Therefore, the service routine for this interrupt is programmed from the address 0038H .

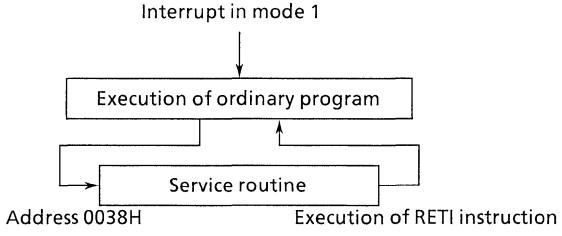


Figure 3.2.7 Interrupt Processing in Mode 1

- Mode 2

The interrupt processing in mode 2 requires a 16-bit pointer consisting of the high-order 8 bits of the I register and the low-order 8 bits (with the LSB="0") of the data fetched from the interrupting CTC or TLCS-Z80 family peripheral LSI. Therefore, the necessary value must be loaded in the I register beforehand. This pointer is used to specify the memory address in the table. The contents of the specified address and the next address provide the start address of the service routine. Therefore, use of this mode requires the table of the service routine's start address (16 bits) to be set at appropriate location under software control . This location can be anywhere in memory. The LSB of the table pointer is set to "0" because a 2-byte data is needed to specify the service routine start address in 16 bits and start that address from an even-number address. In the table, the start that address begins with the low-order byte followed by the high-order byte as shown in Figure 3.2.8.

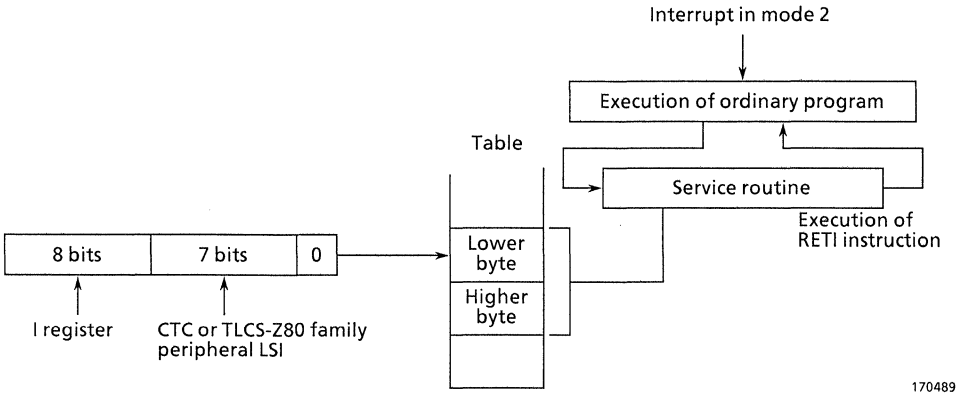


Figure 3.2.8 Interrupt Processing in Mode 2

Mode 2 is used in the daisy chain interrupt processing using the CTC and TLCS-Z80 family LSI. The CTC and TLCS-Z80 family peripheral LSIs all contain the interrupt priority controller in daisy chain structure. In this interrupt structure, the interrupt request signals are connected one after another and given priorities for processing when two or more maskable interrupt requests occur at a time. Only the interrupt vector from the peripheral LSI having the highest priority is put on the data bus. By receiving the interrupt vector in mode 2, the processing for that peripheral LSI can be performed. When an interrupt requested by a peripheral LSI having a priority higher than that of the current peripheral LSI during the execution of the interrupt processing routine, the higher priority interrupt can be enable by the EI instruction to form a interrupt nesting.

The maskable interrupt processing program terminates by executing an RETI instruction. This return instruction performs the following processings:

- Restores the content of the program counter from the stack.
- Notifies the requesting peripheral LSI of the termination of interrupt processing.

3.2.3 MPU Status Transition Diagram and Basic Timing

The following describes the MPU status transition and the basic timing of each MPU operation.

[1] Instruction Cycle

Each TMPZ84C015A instruction is executed by combining the basic operations of memory read/write, input/output, bus request/acknowledge, and interrupt. These basic operations are performed synchronizing with the system clock (the CLK signal).

One clock period is called a state (T). The smallest unit of each basic operation is called a machine cycle (M). Each instruction consists of 1 to 6 machine cycles and each machine cycle consists of 3 to 6 clock states basically. However, the number of clock states in a machine cycle can be increased by the $\overline{WAIT}$ signal described later on. Figure 3.2.9 shows an example of the basic timing of a 3-machine-cycle instruction.

The first machine cycle (M1) of each instruction is the cycle in which the Opcode of the instruction to be executed next is read (this is called the Opcode fetch cycle). The Opcode fetch cycle basically consists of 4 to 6 clock states. In the machine cycle that follows the Opcode fetch cycle, data is transferred between the MPU and the memory or peripheral LSIs. This operation basically consists of 3 to 5 clock states.

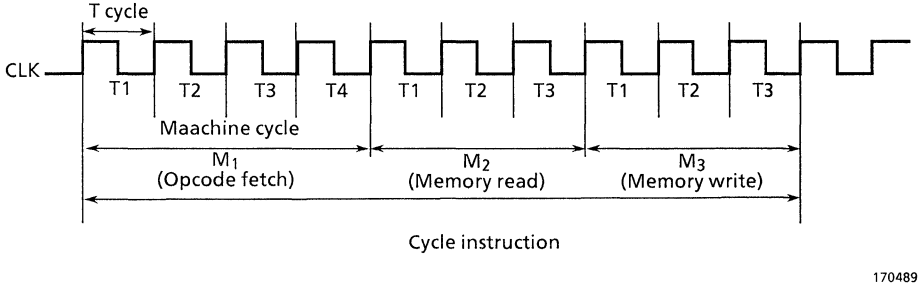


Figure 3.2.9 Example of MPU Basic Timing (3-Machine-Cycle Instruction)

[2] Status Transition Diagram

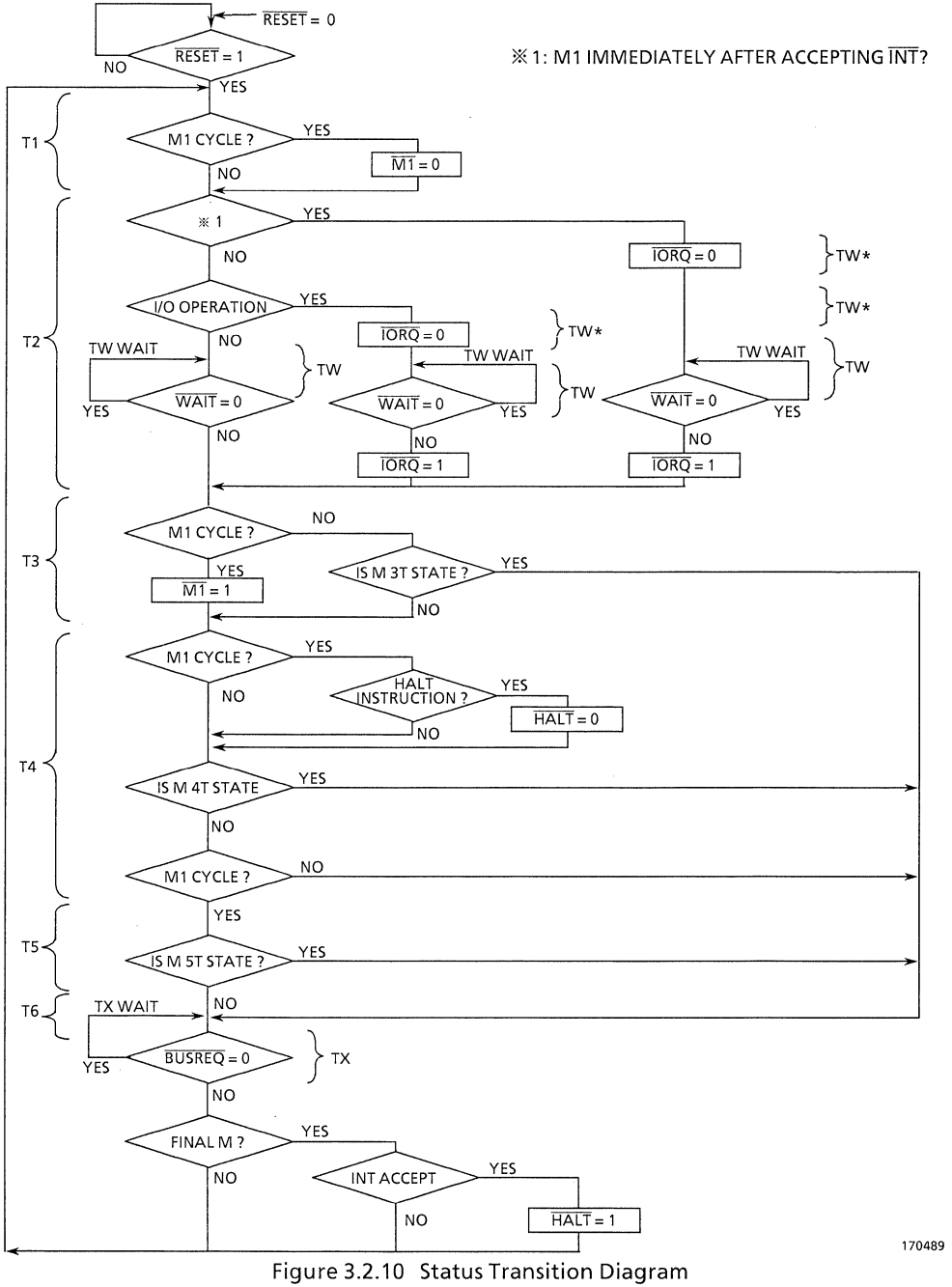


Figure 3.2.10 Status Transition Diagram

[3] Basic Timing

(1) Opcode fetch cycle (M1)

In the Opcode fetch cycle, MPU fetches an Opcode in the machine-language codes in memory. This is also called the M1 cycle because it is the first machine cycle to execute each instruction.

Figure 3.2.12 shows the basic timing of a basic Opcode fetch cycle.

In clock state T1, the content of the program counter is put on the address bus. The $\overline{M1}$ signal goes "0", indicating to the MPU that this is the Opcode fetch cycle. At the same time, $\overline{MREQ}$ and $\overline{RD}$ signals go "0". When the $\overline{MREQ}$ signal goes "0", the address signal has already been stabilized. Therefore, this signal can be used for the memory chip enable signal. The $\overline{RD}$ signal indicates that the MPU is ready to accept the data from memory. By these signals, the MPU accesses memory to fetch the Opcode in the instruction register. The MPU samples the $\overline{WAIT}$ signal on the falling edge of clock state T2. If the $\overline{WAIT}$ signal is "0" on the falling edge of clock state T2 and the following wait state (TW), the next state becomes clock state TW. Figure 3.2.13 shows the delay state of the Opcode fetch cycle caused by the $\overline{WAIT}$ signal.

The data (Opcode) on the data bus is fetched on the rising edge of clock state T3 then, the $\overline{MREQ}$, $\overline{RD}$, and $\overline{M1}$ signals go "1". In clock state T3, a memory refresh address is put on the 8 bits consisting of the low-order 7 bits of the address bus and the A7RF corresponding to bit 8 and the $\overline{RFSH}$ signal goes "0" and the $\overline{MREQ}$ signal goes "0" again. This signal indicates that the memory refresh cycle is on. At this time, the contents of the I register are put on the high-order 8 bits of the address bus and the 7 bits of the R register contents and the A7RF signal corresponding to bit 8 are put on the low-order 8 bits of the address bus. By using the $\overline{RFSH}$ and $\overline{MREQ}$ signals, memory refresh is performed in clock state T3 and T4. However, the $\overline{RD}$ signal remains "1" because the contents of the memory refresh address are not put on the data bus.

The address bus of 8 bits consisting of the address low-order 7 bits of address (A6 through A0) and the A7RF are used as the 8-bit refresh address. That is, when A7RF is used for the refresh address, signals "00H" through "FFH" are output. In cycles other than the refresh cycle, the signal equivalent to A7RF are output. However, at reset, the signals to be output are uncertain. Figure 3.2.11 shows the refresh timing.

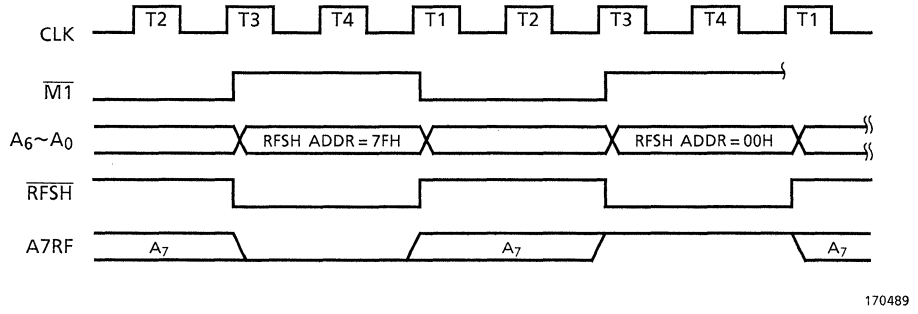


Figure 3.2.11 Refrish Timing

In clock state T4, the $\overline{MREQ}$ signal returns to "1". The refresh address is kept output until the rising edge of the clock state T1 in the next machine cycle, keeping the $\overline{RFSH}$ signal set to "0". The cycle delay state caused by setting the $\overline{WAIT}$ signal to "0" is the same in the memory read/write, input/output, and maskable interrupt acknowledge cycles. The diagram of the cycle delay state caused by the $\overline{WAIT}$ signal set to "0" is omitted in the following description.

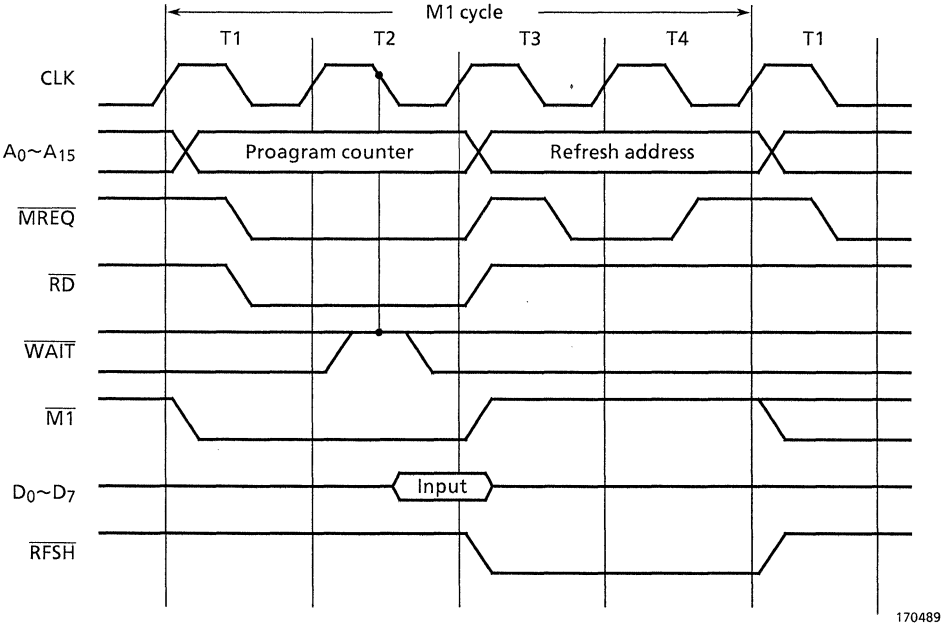


Figure 3.2.12 Opcode Fetch Timing

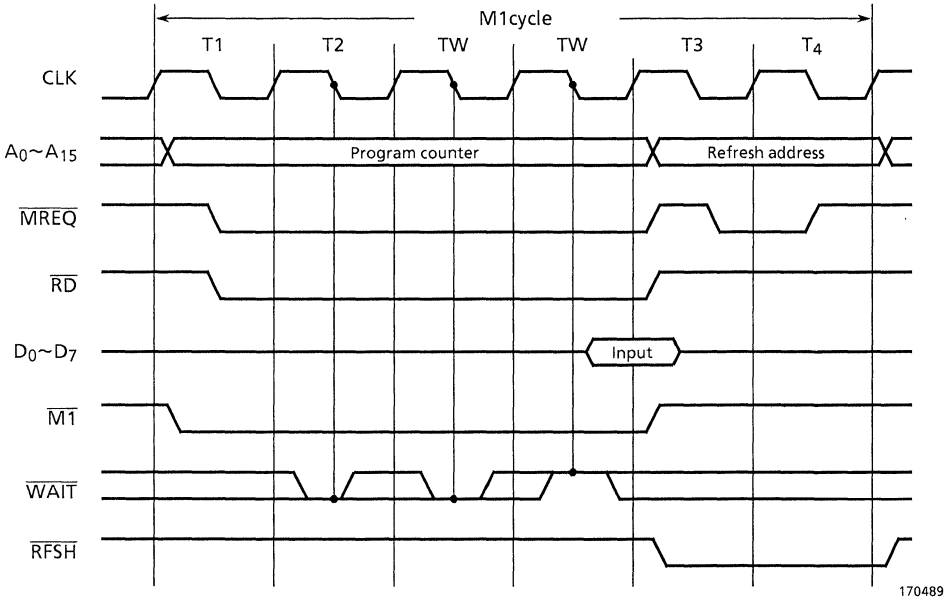


Figure 3.2.13 Opcode Fetch Timing Including Wait State

(2) Memory read/write operations

Figure 3.2.14 shows the basic timing of memory read/write operations (except for the Opcode fetch cycle) in the same diagram for convenience.

In each operation, the memory address signal to read/write data on the address bus is output in clock state T1. The operation in which the $\overline{WAIT}$ signal is sampled in clock state T2 and the following TW state is the same as the Opcode fetch cycle.

In memory read, memory data is put on the data bus by the address $\overline{MREQ}$, and $\overline{RD}$ signals. The MPU reads this data.

In memory write, the memory address signal is put on the address bus then the $\overline{MREQ}$ signal is set to "0" to put the write data onto the data bus. When the data bus has been stabilized the $\overline{WR}$ signal is output in clock state T2. The $\overline{WR}$ signal can be used as the memory write signal.

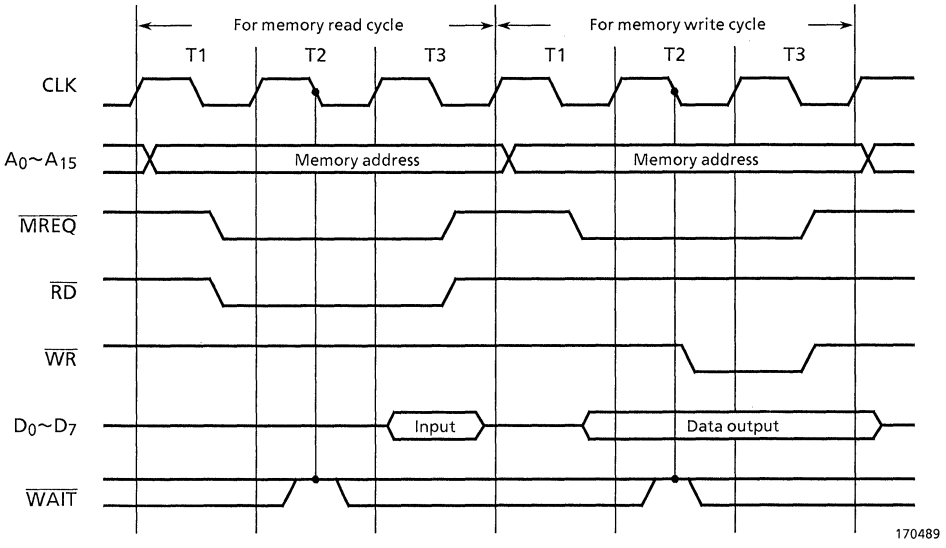


Figure 3.2.14 Memory Read/Write Cycle Timing

(3) Input/output operations

Figure 3.2.15 shows the basic timing of input/output operations. The feature of the I/O operation timing is that, regardless of the state of the $\overline{WAIT}$ signal in clock state T2, the I/O cycle automatically goes in the wait state (TW*) after clock T2. The $\overline{WAIT}$ signal is sampled on the falling edge of TW*. If the $\overline{WAIT}$ signal is "0" on the falling edges of TW* and the following clock state, the I/O operation enters into clock state TW*. Clock state TW* is inserted because the $\overline{IORQ}$ signal goes "0" in clock state T2, so that it is too late to sample the $\overline{WAIT}$ signal after decoding the I/O port address. In each of input and output operations, the I/O port address is put on the low-order 8 bits of the address bus in clock state T1. On the high-order 8 bits, the contents of the accumulator or B register are output. In clock state T2, the $\overline{IORQ}$ signal goes "0" instead of the $\overline{MREQ}$ signal. The $\overline{IORQ}$ signal can be used as the chip enable signal for a peripheral LSI.

In an input operation, the contents of the input port are read onto the data bus by the address, $\overline{IORQ}$, or $\overline{RD}$ signals. The MPU reads this data.

In an output operation, the output port address and the output data are respectively put on the address bus and data bus in clock state Tl, then the $\overline{IORQ}$ and $\overline{WR}$ signals go "0" in clock state T2. The $\overline{WR}$ signal can be used as the output port write signal.

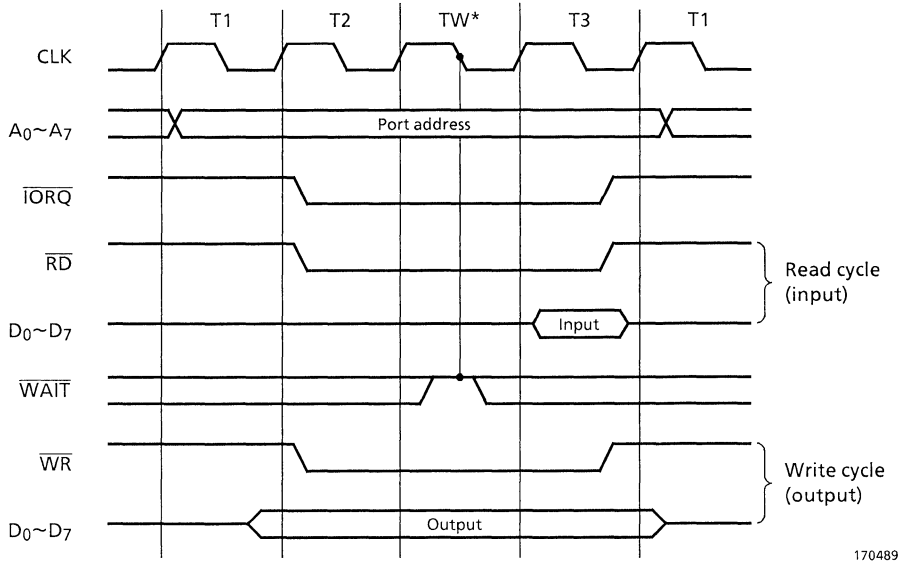


Figure 3.2.15 I/O Operating Timing

(4) Bus request and bus acknowledge operations

Figure 3.2.16 shows the basic timing of bus request and bus acknowledge operations.

The address bus (A0 through A15), data bus (D0 through D7), $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ signals controlled by the MPU can be put in the high-impedance state (floating) to electrically disconnect them from the MPU. This operation, after sampling the $\overline{BUSREQ}$ signal on the rising edge of the last clock of each machine cycle, starts on the rising edge of the next clock if this signal is found "0".

Subsequently, these buses are controlled by external peripheral LSIs. For example, data can be directly transferred between memory and these peripheral LSIs. This state is cleared if the $\overline{BUSREQ}$ signal is found"1"after sampling it on the rising edge of each subsequent clock state (TX), and enters into the next machine cycle. During the floating state, the $\overline{BUSACK}$ signal goes "0" to indicate it to the peripheral LSIs.

In this state, however, no memory refresh is performed and, therefore, the $\overline{RFSH}$ signal is set to "1". Hence, to maintain this state for a long time with a system using dynamic memory, memory refresh must be performed by the external controller.

Note that, in the floating state, neither maskable interrupt (INT) nor non-maskable interrupts (NMI) can be accepted.

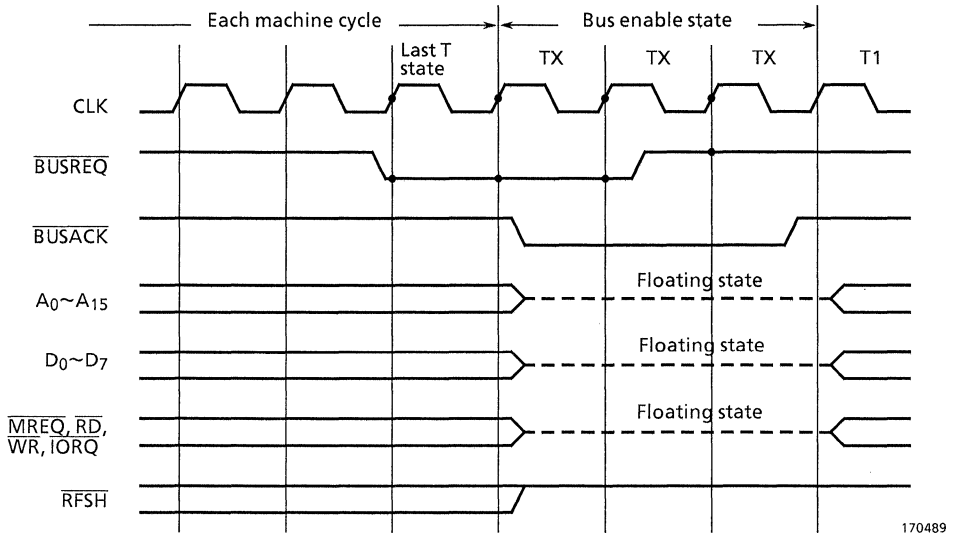


Figure 3.2.16 Bus Request and Bus Acknowledge Timing

(5) Maskable interrupt acknowledge operation

Figure 3.2.17 shows the basic timing of the maskable interrupt acknowledge.

The MPU samples the maskable interrupt request signal (INT) on the rising edge of the last clock of each instruction execution. If the $\overline{INT}$ signal is found "0", a maskable interrupt is accepted except in the following cases:

- The interrupt enable flip-flop is reset to "0".
- The $\overline{BUSREQ}$ signal is "0".

When a maskable interrupt has been accepted, a special Opcode fetch cycle is generated. In this cycle, 2 clock states of wait state (TW*) is automatically inserted after the clock state T2. The $\overline{WAIT}$ signal is sampled on the falling edges of the second clock state TW* and the following clock state TW and, if the $\overline{WAIT}$ signal is found "0", the instruction cycle enters in the next clock state TW. In this Opcode fetch cycle, the $\overline{IORQ}$ signal goes "0" in the first TW* state instead of the $\overline{MREQ}$ signal while, in a normal Opcode fetch cycle, the $\overline{MREQ}$ signal goes "0" in clock state T1. This indicates to the maskable interrupt requesting LSI that the 8-bit interrupt vector can be put on the data bus. The MPU reads this data to perform interrupt processing. Therefore, the contents of the program counter put on the address bus are not used. Unlike an ordinary I/O operation, the $\overline{RD}$ signal does not go "0".

In clock state T3 the memory refresh address signal is put on the address bus for memory refresh like normal Opcode fetch cycle and the $\overline{RFSH}$ signal goes "0". In the subsequent machine cycles (M2 and M3) , the contents of the current program counter are saved into the stack. In machine cycles M4 and M5, the contents of the I register (the high-order 8 bits) and the contents of the address indicated by the address of the vector (the low-order 8 bits) from the CTC and the peripheral LSI are fetched in the program counter.

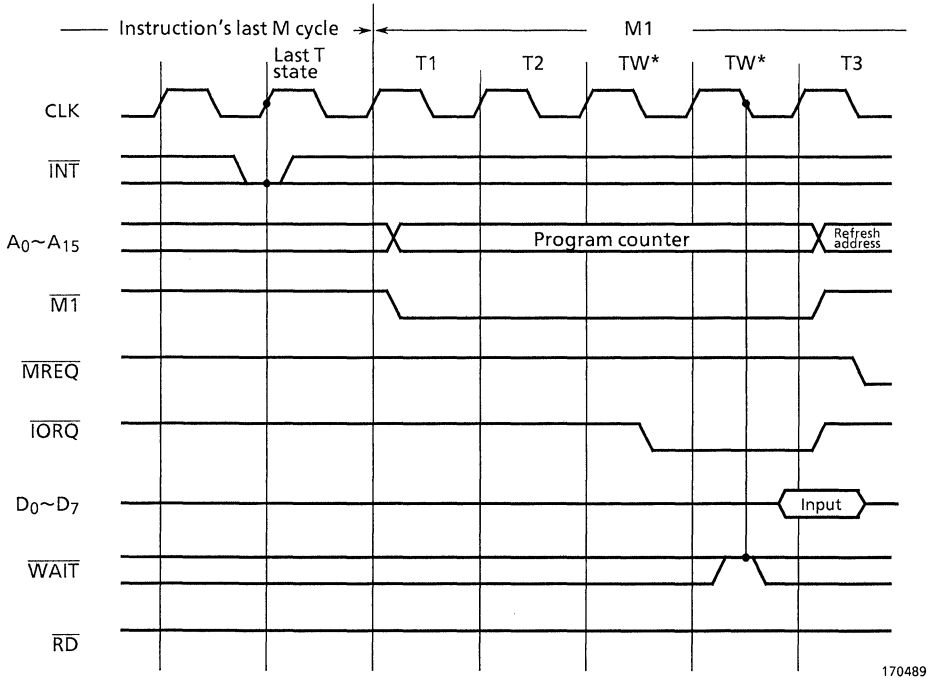


Figure 3.2.17 Maskable Interrupt Acknowledge Timing

(6) Non-maskable interrupt acknowledge operation

Figure 3.2.18 shows the basic timing of non-maskable interrupt acknowledge.

When the non-maskable interrupt request signal ($\overline{NMI}$) goes low, the internal non-maskable flip-flop is set to "1". The $\overline{NMI}$ signal is detected in any timing of each instruction. However, the internal NMI flip-flop is sampled on the rising edge of the last clock of each instruction. Therefore, the $\overline{NMI}$ signal should go low by the last clock state of an instruction.

The Opcode fetch cycle for non-maskable interrupt request acknowledge is generally the same as the ordinary Opcode fetch cycle. However, the Opcode on the data bus at the time is ignored. The contents of the current program counter are saved into the stack in the subsequent machine cycles (M2 and M3). In the following machine cycle, the operation jumps to address 0066H, the non-maskable interrupt vector address. The machine cycles after these depend on the contents of the fetched Opcode.

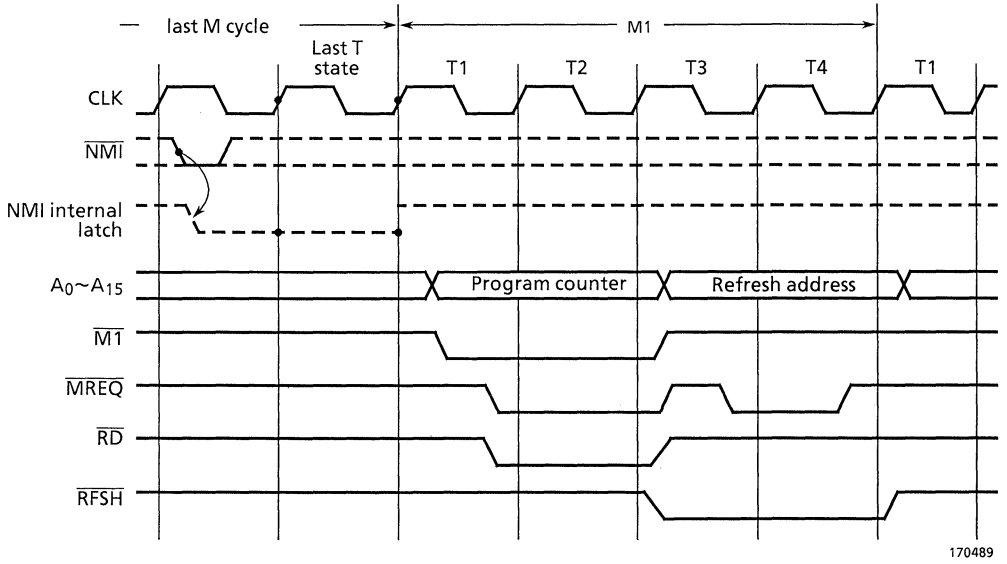


Figure 3.2.18 Non-Maskable Inpterrupt Acknowledge Timing

(7) Halt operation

When a HALT instruction is fetched in the Opcode fetch cycle, the MPU sets the $\overline{\text{HALT}}$ signal to "0" synchronized with the falling edge of clock state T4 to indicate it to the peripheral LSI and stops operating. If the system clock is kept supplied in the halt state, the MPU continues executing NOP instructions. This is done to output refresh signals when the dynamic memory is used. The NOP instruction execution cycle is the same as the ordinary Opcode fetch cycle except the data on the data bus are ignored.

The halt state is cleared when an interrupt is accepted or the $\overline{\text{RESET}}$ signal is set to "0" to reset the MPU. Figure 3.2.19 shows the halt state clear operation by interrupt acknowledge. An interrupt is sampled on the rising edge of the last clock (clock state T4) of the NOP instruction. A maskable interrupt can be accepted when the $\overline{\text{INT}}$ signal is "0". A non-maskable interrupt is accepted when the internal NMI flip-flop which is set on the falling edge of the $\overline{\text{NMI}}$ signal is set at

"1". However, it is required that the interrupt enable flip-flop is set to "1" for a maskable interrupt to be accepted. The interrupt processing for the accepted interrupt starts from the next cycle.

However, when the supply of the system clock from the CGC has been stopped by the power down operation, it is required to restart the supply of the system clock and input the $\overline{INT}$ signal until the execution of one instruction is completed or the $\overline{RESET}$ signal until 3 clocks are output. Figure 3.2.20 shows the timing of clearing the halt state caused by power down.

For the reset operation, see (8) "Reset operation" . Note that the $\overline{INT}$ and $\overline{NMI}$ signals are shown on the same diagram in Figures 3.2.19 and 3.2.20 for convenience.

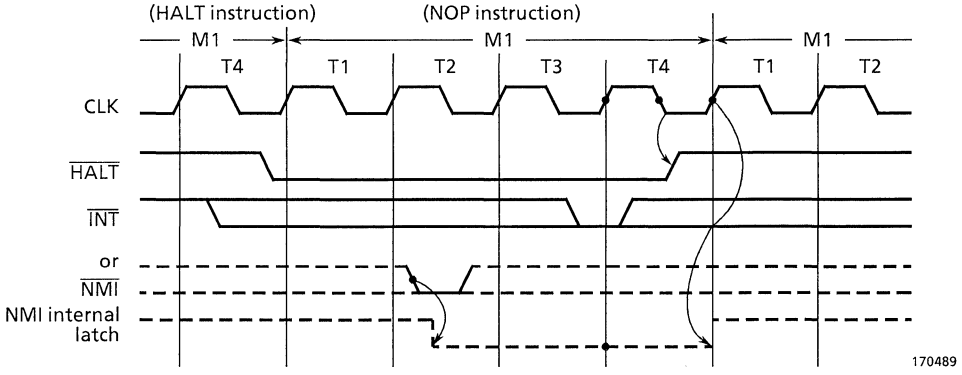


Figure 3.2.19 Timing of Clearing Halt State Caused by Interrupt Acknowledge

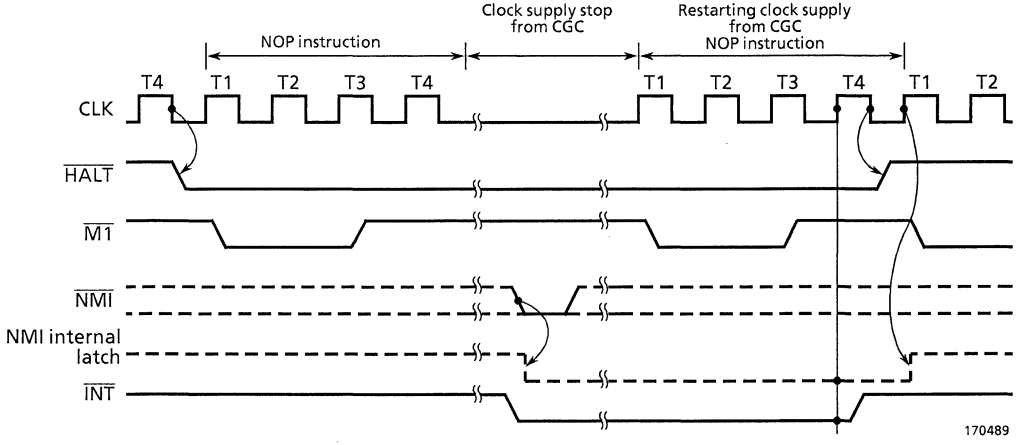


Figure 3.2.20 Timing of Clearing Halt State Caused By Power Down

(8) Reset operation

Figure 3.2.21 shows the basic timing of reset operation.

To reset the MPU, the $\overline{\text{RESET}}$ signal must be kept at "0" for at least 3 clocks. When the $\overline{\text{RESET}}$ signal goes "1", instruction execution starts from address 0000H after a dummy cycle of at least 2T clock states.

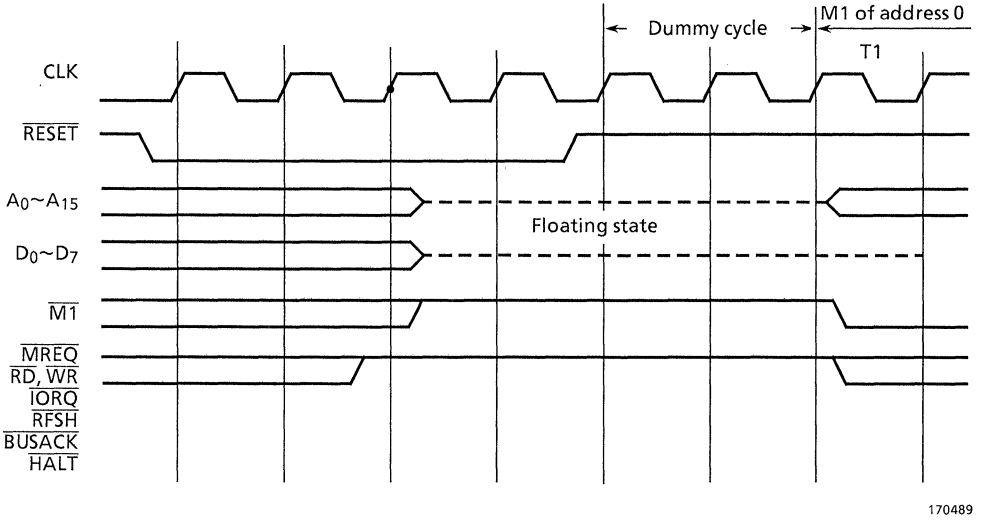


Figure 3.2.21 Reset Timing

To clear the power down state by using the $\overline{\text{RESET}}$ signal, the $\overline{\text{RESET}}$ signal must be input until 3 clocks or more are supplied by restarting the supply of the system clock from the CGC.

(9) Evaluation operation

Each of the MPU signals (A0 through A15, D0 through D7, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, $\overline{WR}$, $\overline{HALT}$, $\overline{M1}$, and $\overline{RFSH}$) can be put in the high-impedance state by EV and $\overline{BUSREQ}$ signals to electrically disconnect them from the MPU.

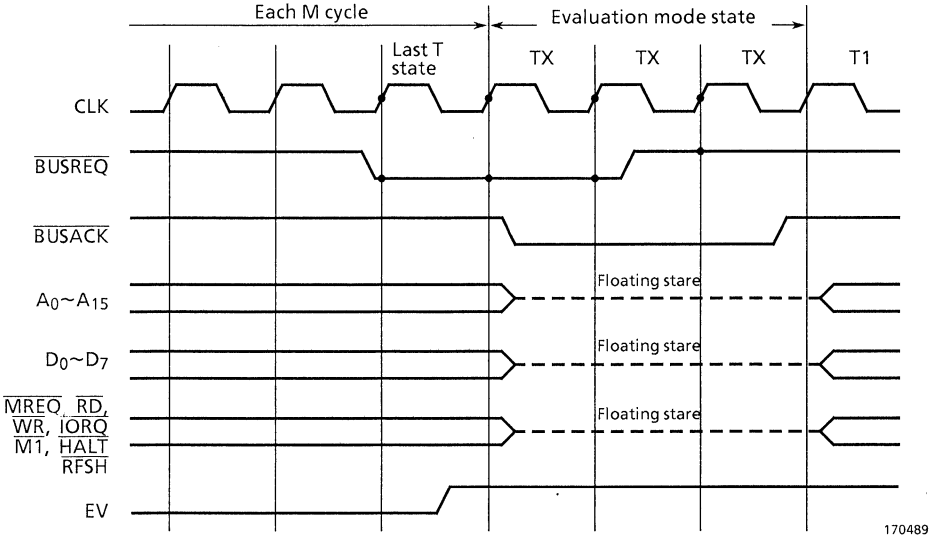


Figure 3.2.22 Evaluation Timing

Figure 3.2.23 shows the block diagram of the TMPZ84C015A operating as an evaluator in the evaluation mode.

The operations controlled by signals from the external MPU in the evaluation mode are the same as those of each device constituting the TMPZ84C015A. (However, for the watchdog timer operations, see "WDT Operational Description" because the watchdog timer is of random logic configuration.)

For the electrical characteristics and timing of each device, see "Inactive State".

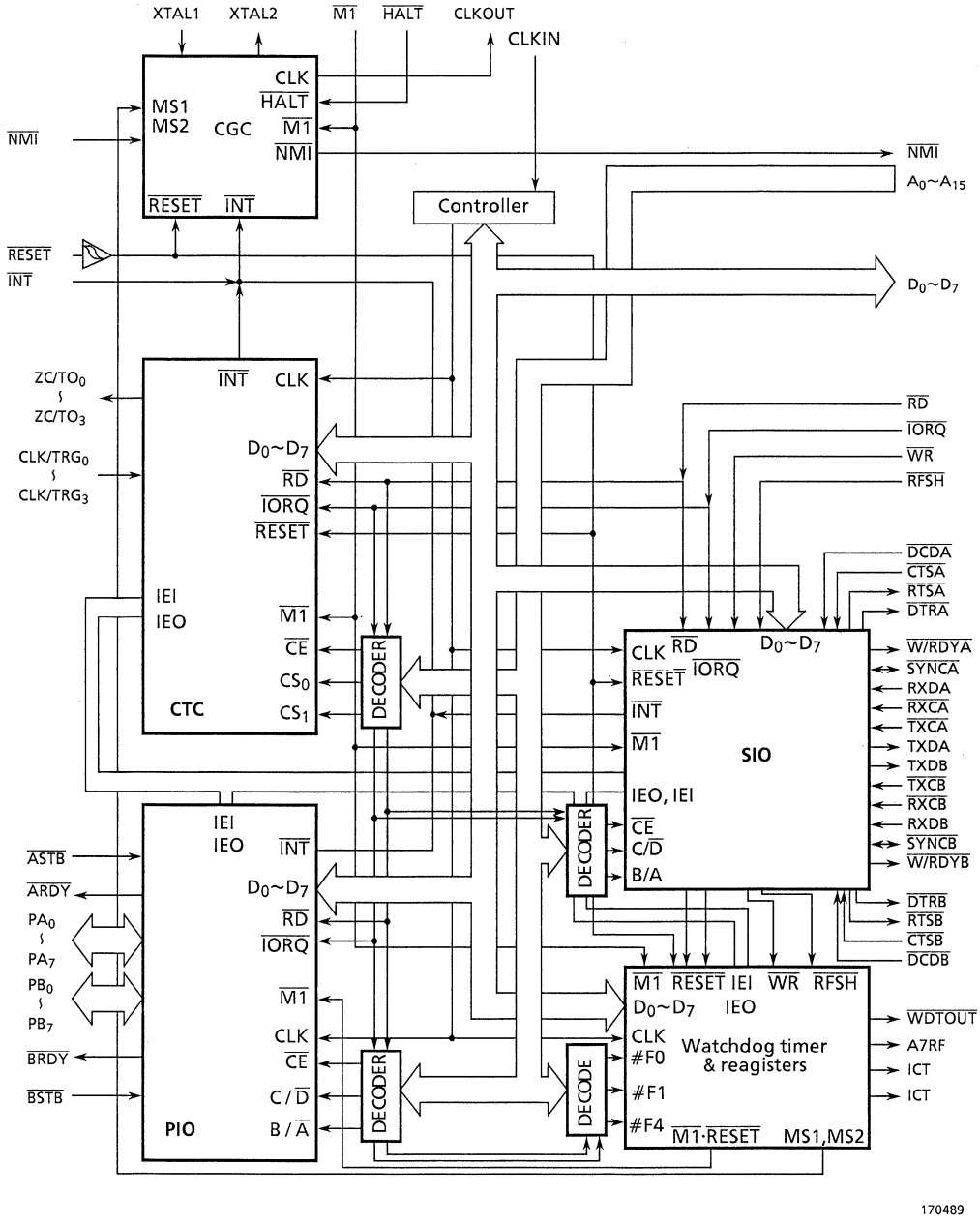


Figure 3.2.23 Block Diagram of the TMPZ84C015A Functioning As Evaluator

3.2.4 TMPZ84C015A Instruction Set

This subsection lists the TMPZ84C015A instruction codes and their functions. The table below lists the symbols and abbreviations used to describe the instruction set. The symbols which require special attention are described in the locations in which they appear.

- Symbols (1/2)

Classification	Symbol	Meaning	
Register	r, g	Register	B, C, D, E, H, L, A,
	t	Register pair	BC, DE, HL
		Stack pointer	SP
	q	Register pair	BC, DE, HL, AF
	p	Register pair	BC, DE
		Index register	IX
		Stack pointer	SP
	s	Register pair	BC, DE
		Index register	IY
		Stack pointer	SP
	t_H	Higher register of register pair (B, D, H)	
		Higher 8 bits of stack pointer (SP)	
	q_H	Higher register of register pair (B, D, H, A)	
	IX_H	Higher 8 bits of index register IX	
	IY_H	Higher 8 bits of index register IY	
	PC_H	Higher 8 bits of program counter (PC)	
	t_L	Lower register of register pair (C, E, L)	
		Lower 8 bits of stack pointer (SP)	
	q_L	Lower register of register pair (C, E, L, F)	
	IX_L	Lower 8 bits of index register IX	
	IY_L	Lower 8 bits of index register IY	
	PC_L	Lower 8 bits of program counter (PC)	
	rb	Bit b (0-7) of register (B, C, D, E, H, L, A)	

120489

- Symbols (2/2)

Classification	Symbol	Meaning
Memory	mn	Memory address represented in 16 bits. m indicates higher 8 bits and n, lower 8 bits.
	$(HL)_b$	Bit b (0-7) of the contents of the memory address indicated by register pair HL.
	$(IX+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IX.
	$(IY+d)_b$	Bit b (0-7) of the contents of the memory address indicated by the value obtained by adding 8-bit data d to the content of index register IY.
Flag change symbol	0	Reset to "0" by operation.
	1	Set to "1" by operation.
	-	No change
	*	Affected by operation
	X	Undefined
	P	Handled as parity flag. P = 0: odd parity P = 1: even parity
	V	Handled as overflow flag. V = 0: No overflow V = 1: Overflow
Operator	←	Transfer
	↔	Exchange
	+	Add
	−	Subtract
	∧	Logical and between bits.
	∨	Logical or between bits.
	⊕	Exclusive or between bits
Others	IFF	Interrupt enable flip-flop
	CY	Carry flag
	Z	Zero flag

120489

TMPZ84C015A Instruction Set (1/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT DATA LOAD	LD r,g	01 rrr ggg	40+r×8+g	r←g	–	–	X	–	X	–	–	–	1	4
	LD r,n	00 rrr 110 nn nnn nnn	06+r×8 n	r←n	–	–	X	–	X	–	–	–	2	7
	LD r,(HL)	01 rrr 110	46+r×8	r←(HL)	–	–	X	–	X	–	–	–	2	7
	LD r,(IX+d)	11 011 101 01 rrr 110 dd ddd ddd	DD 46+r×8 d	r←(IX+d)	–	–	X	–	X	–	–	–	5	19
	LD r,(IY+d)	11 111 101 01 rrr 110 dd ddd ddd	FD 46+r×8 d	r←(IY+d)	–	–	X	–	X	–	–	–	5	19
	LD (HL),r	01 110 rrr	70+r	(HL)←r	–	–	X	–	X	–	–	–	2	7
	LD (IX+d),r	11 011 101 01 110 rrr dd ddd ddd	DD 70+r d	(IX+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),r	11 111 101 01 110 rrr dd ddd ddd	FD 70+r d	(IY+d)←r	–	–	X	–	X	–	–	–	5	19
	LD (HL),n	00 110 110 nn nnn nnn	36 n	(HL)←n	–	–	X	–	X	–	–	–	3	10
	LD (IX+d), n	11 011 101 00 110 110 dd ddd ddd nn nnn nnn	DD 36 d n	(IX+d)←n	–	–	X	–	X	–	–	–	5	19
	LD (IY+d),n	11 111 101 00 110 110 dd ddd ddd nn nnn nnn	FD 36 d n	(IY+d)←n	–	–	X	–	X	–	–	–	5	19
	LD A,(BC)	00 001 010	0A	A←(BC)	–	–	X	–	X	–	–	–	2	7
	LD A,(DE)	00 011 010	1A	A←(DE)	–	–	X	–	X	–	–	–	2	7
	LD A,(mn)	00 111 010 nn nnn nnn mm mmm mmm	3A n m	A←(mn)	–	–	X	–	X	–	–	–	4	13
	LD (BC),A	00 000 010	02	(BC)←A	–	–	X	–	X	–	–	–	2	7
	LD (DE),A	00 010 010	12	(DE)←A	–	–	X	–	X	–	–	–	2	7
	LD (mn),A	00 110 010 nn nnn nnn mm mmm mmm	32 n m	(mn)←A	–	–	X	–	X	–	–	–	4	13
	LD A,I	11 101 101 01 010 111	ED 57	A←I	*	*	X	0	X	IFF	0	–	2	9
	LD A,R	11 101 101 01 011 111	ED 5F	A←R	*	*	X	0	X	IFF	0	–	2	9
	LD I,A	11 101 101 01 000 111	ED 47	I←A	–	–	X	–	X	–	–	–	2	9
	LD R,A	11 101 101 01 001 111	ED 4F	R←A	–	–	X	–	X	–	–	–	2	9
16-BIT DATA LOAD	LD t,mn	00 tt0 001 nn nnn nnn mm mmm mmm	01+t×10 n m	t←mn	–	–	X	–	X	–	–	–	3	10
	LD IX,mn	11 011 101 00 100 001 nn nnn nnn mm mmm mmm	DD 21 n m	IX←mn	–	–	X	–	X	–	–	–	4	14

r	rrr
g	ggg
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

Note : r,g means any of the registers A, B, C, D, E, H, L.
IFF in "Flag" column indicates that the content of the interrupt enable flip-flop is copied into the P/V flag.

120489

TMPZ84C015A Instruction Set (2/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
16-BIT DATA LOAD	LD IY,mn	11 111 101 00 100 001 nn nnn nnn mm mmm mmm	FD 21 n m	IY←mn	–	–	X	–	X	–	–	–	4	14
	LD HL,(mn)	00 101 010 nn nnn nnn mm mmm mmm	2A n m	H←(mn+1) L←(mn)	–	–	X	–	X	–	–	–	5	16
	LD t,(mn)	11 101 101 01 tt1 011 nn nnn nnn mm mmm mmm	ED 4B+t×10 n m	t_H←(mn+1) t_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IX,(mn)	11 011 101 00 101 010 nn nnn nnn mm mmm mmm	DD 2A n m	IX_H←(mn+1) IX_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD IY,(mn)	11 111 101 00 101 010 nn nnn nnn mm mmm mmm	FD 2A n m	IY_H←(mn+1) IY_L←(mn)	–	–	X	–	X	–	–	–	6	20
	LD (mn),HL	00 100 010 nn nnn nnn mm mmm mmm	22 n m	(mn+1)←H (mn)←L	–	–	X	–	X	–	–	–	5	16
	LD (mn),t	11 101 101 01 tt0 011 nn nnn nnn mm mmm mmm	ED 43+t×10 n m	(mn+1)←t_H (mn)←t_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IX	11 011 101 00 100 010 nn nnn nnn mm mmm mmm	DD 22 n m	(mn+1)←IX_H (mn)←IX_L	–	–	X	–	X	–	–	–	6	20
	LD (mn),IY	11 111 101 00 100 010 nn nnn nnn mm mmm mmm	FD 22 n m	(mn+1)←IY_H (mn)←IY_L	–	–	X	–	X	–	–	–	6	20
	LD SP,HL	11 111 001	F9	SP←HL	–	–	X	–	X	–	–	–	1	6
	LD SP,IX	11 011 101 11 111 001	DD F9	SP←IX	–	–	X	–	X	–	–	–	2	10
	LD SP,IY	11 111 101 11 111 001	FD F9	SP←IY	–	–	X	–	X	–	–	–	2	10
	PUSH q	11 qq0 101	C5+q×10	(SP-2)←q_L,(SP-1)←q_H, SP←SP-2	–	–	X	–	X	–	–	–	3	11
	PUSH IX	11 011 101 11 100 101	DD E5	(SP-2)←IX_L,(SP-1)←IX_H SP←SP-2	–	–	X	–	X	–	–	–	4	15
	PUSH IY	11 111 101 11 100 101	FD E5	(SP-2)←IY_L,(SP-1)←IY_H SP←SP-2	–	–	X	–	X	–	–	–	4	15
	POP q	11 qq0 001	C1+q×10	q_H←(SP+1),q_L←(SP), SP←SP+2	–	–	X	–	X	–	–	–	3	10
	POP IX	11 011 101 11 100 001	DD E1	IX_H←(SP+1),IX_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
	POP IY	11 111 101 11 100 001	FD E1	IY_H←(SP+1),IY_L←(SP) SP←SP+2	–	–	X	–	X	–	–	–	4	14
*1	EX DE,HL	11 101 011	EB	DE↔HL	–	–	X	–	X	–	–	–	1	4
	EX AF,AF'	00 001 000	08	AF↔AF'	–	–	X	–	X	–	–	–	1	4
	EXX	11 011 001	D9	BC↔BC',DE↔DE',HL↔HL'	–	–	X	–	X	–	–	–	1	4

t	tt
BC	00
DE	01
HL	10
SP	11

q	qq
BC	00
DE	01
HL	10
AF	11

Note : t is any of the register pairs BC, DE, HL, SP.
q is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively. (Ex) $BC_L = C$, $AF_H = A$.

*1 : EXCHANGE

120489

TMPZ84C015A Instruction Set (3/9)

ITEM/CLASSIFICATION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z	H	P/V	N	C	No. OF CYCLES	No. OF STATES
EXCHANGE	EX (SP),HL	11 100 011	E3	H↔(SP+1),L↔(SP)	–	–	X	–	X	–	–	–	5	19
	EX (SP),IX	11 011 101 11 100 011	DD E3	IX_H↔(SP+1) IX_L↔(SP)	–	–	X	–	X	–	–	–	6	23
	EX (SP),IY	11 111 101 11 100 011	FD E3	IY_H↔(SP+1) IY_L↔(SP)	–	–	X	–	X	–	–	–	6	23
BLOCK TRANSFER BLOCK SEARCH	LDI	11 101 101 10 100 000	ED A0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDIR	11 101 101 10 110 000	ED B0	(DE)←(HL),DE←DE+1 HL←HL+1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	LDD	11 101 101 10 101 000	ED A8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1	–	–	X	0	X	*M	0	–	4	16	
	LDDR	11 101 101 10 111 000	ED B8	(DE)←(HL),DE←DE−1 HL←HL−1,BC←BC−1 Repeat until BC=0	–	–	X	0	X	0	0	–	5 4	21 16	←[BC< >0] ←[BC=0]
	CPI	11 101 101 10 100 001	ED A1	A−(HL) HL←HL+1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPIR	11 101 101 10 110 001	ED B1	A−(HL),HL←HL+1,BC←BC−1 Repeat until A=(HL) or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
	CPD	11 101 101 10 101 001	ED A9	A−(HL) HL←HL−1,BC←BC−1	*	*N	X	*	X	*M	1	–	4	16	
	CPDR	11 101 101 10 111 001	ED B9	A−(HL),HL←HL−1,BC←BC−1 Repeat until A=(HL)or BC=0	*	*N	X	*	X	*M	1	–	5 4	21 16	←[BC< >0 & A< > (HL)] ↖[BC=0 or A= (HL)]
8-BIT ARITHMETIC AND LOGICAL	ADD A,r	10 000 rrr	80+r	A←A+r	*	*	X	*	X	V	0	*	1	4	
	ADD A,n	11 000 110 nn nnn nnn	C6 n	A←A+n	*	*	X	*	X	V	0	*	2	7	
	ADD A,(HL)	10 000 110	86	A←A+(HL)	*	*	X	*	X	V	0	*	2	7	
	ADD A,(IX+d)	11 011 101 10 000 110 dd ddd ddd	DD 86 d	A←A+(IX+d)	*	*	X	*	X	V	0	*	5	19	
	ADD A,(IY+d)	11 111 101 10 000 110 dd ddd ddd	FD 86 d	A←A+(IY+d)	*	*	X	*	X	V	0	*	5	19	
	ADC A,r	10 001 rrr	88+r	A←A+r+CY	*	*	X	*	X	V	0	*	1	4	
	ADC A,n	11 001 110 nn nnn nnn	CE n	A←A+n+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(HL)	10 001 110	8E	A←A+(HL)+CY	*	*	X	*	X	V	0	*	2	7	
	ADC A,(IX+d)	11 011 101 10 001 110 dd ddd ddd	DD 8E d	A←A+(IX+d)+CY	*	*	X	*	X	V	0	*	5	19	
	ADC A,(IY+d)	11 111 101 10 001 110 dd ddd ddd	FD 8E d	A←A+(IY+d)+CY	*	*	X	*	X	V	0	*	5	19	
	SUB r	10 010 rrr	90+r	A←A−r	*	*	X	*	X	V	1	*	1	4	
	SUB n	11 010 110 nn nnn nnn	D6 n	A←A−n	*	*	X	*	X	V	1	*	2	7	
	SUB (HL)	10 010 110	96	A←A−(HL)	*	*	X	*	X	V	1	*	2	7	
	SUB (IX+d)	11 011 101 10 010 110 dd ddd ddd	DD 96 d	A←A−(IX+d)	*	*	X	*	X	V	1	*	5	19	
	SUB (IY+d)	11 111 101 10 010 110 dd ddd ddd	FD 96 d	A←A−(IY+d)	*	*	X	*	X	V	1	*	5	19	

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : *M P/V flag is 0 if the result of BC − 1 = 0, otherwise P/V = 1.
*N Z flag is 1 if A = (HL), otherwise Z = 0.
[ ]indicates the total condition of the number of cycles and states indicated by arrow.
r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C015A Instruction Set (4/9)

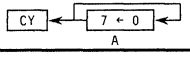
ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
8-BIT ARITHMETIC AND LOGICAL	SBC A,r	10 011 rrr	98+r	A←A−r−CY	*	*	X	*	X	V	1	*	1	4
	SBC A,n	11 011 110 nn nnn nnn	DE n	A←A−n−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(HL)	10 011 110	9E	A←A−(HL)−CY	*	*	X	*	X	V	1	*	2	7
	SBC A,(IX+d)	11 011 101 10 011 110 dd ddd ddd	DD 9E d	A←A−(IX+d)−CY	*	*	X	*	X	V	1	*	5	19
	SBC A,(IY+d)	11 111 101 10 011 110 dd ddd ddd	FD 9E d	A←A−(IY+d)−CY	*	*	X	*	X	V	1	*	5	19
	AND r	10 100 rrr	A0+r	A←A∧r	*	*	X	1	X	P	0	0	1	4
	AND n	11 100 110 nn nnn nnn	E6 n	A←A∧n	*	*	X	1	X	P	0	0	2	7
	AND (HL)	10 100 110	A6	A←A∧(HL)	*	*	X	1	X	P	0	0	2	7
	AND (IX+d)	11 011 101 10 100 110 dd ddd ddd	DD A6 d	A←A∧(IX+d)	*	*	X	1	X	P	0	0	5	19
	AND (IY+d)	11 111 101 10 100 110 dd ddd ddd	FD A6 d	A←A∧(IY+d)	*	*	X	1	X	P	0	0	5	19
	OR r	10 110 rrr	B0+r	A←A∨r	*	*	X	0	X	P	0	0	1	4
	OR n	11 110 110 nn nnn nnn	F6 n	A←A∨n	*	*	X	0	X	P	0	0	2	7
	OR (HL)	10 110 110	B6	A←A∨(HL)	*	*	X	0	X	P	0	0	2	7
	OR (IX+d)	11 011 101 10 110 110 dd ddd ddd	DD B6 d	A←A∨(IX+d)	*	*	X	0	X	P	0	0	5	19
	OR (IY+d)	11 111 101 10 110 110 dd ddd ddd	FD B6 d	A←A∨(IY+d)	*	*	X	0	X	P	0	0	5	19
	XOR r	10 101 rrr	A8+r	A←A∀r	*	*	X	0	X	P	0	0	1	4
	XOR n	11 101 110 nn nnn nnn	EE n	A←A∀n	*	*	X	0	X	P	0	0	2	7
	XOR (HL)	10 101 110	AE	A←A∀(HL)	*	*	X	0	X	P	0	0	2	7
	XOR (IX+d)	11 011 101 10 101 110 dd ddd ddd	DD AE d	A←A∀(IX+d)	*	*	X	0	X	P	0	0	5	19
	XOR (IY+d)	11 111 101 10 101 110 dd ddd ddd	FD AE d	A←A∀(IY+d)	*	*	X	0	X	P	0	0	5	19
	CP r	10 111 rrr	B8+r	A−r	*	*	X	*	X	V	1	*	1	4
	CP n	11 111 110 nn nnn nnn	FE n	A−n	*	*	X	*	X	V	1	*	2	7
	CP (HL)	10 111 110	BE	A−(HL)	*	*	X	*	X	V	1	*	2	7
	CP (IX+d)	11 011 101 10 111 110 dd ddd ddd	DD BE d	A−(IX+d)	*	*	X	*	X	V	1	*	5	19
	CP (IY+d)	11 111 101 10 111 110 dd ddd ddd	FD BE d	A−(IY+d)	*	*	X	*	X	V	1	*	5	19
	INC r	00 rrr 100	04+r×8	r←r+1	*	*	X	*	X	V	0	−	1	4
	INC (HL)	00 110 100	34	(HL)←(HL)+1	*	*	X	*	X	V	0	−	3	11
	INC (IX+d)	11 011 101 00 110 100 dd ddd ddd	DD 34 d	(IX+d)←(IX+d)+1	*	*	X	*	X	V	0	−	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C015A Instruction Set (5/9)

ITEM/CLASSIFICATION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CYCLES	No. OF STATES
8-BIT ARITHMETIC AND LOGICAL	INC (IY+d)	11 111 101 00 110 100 dd ddd ddd	FD 34 d	(IY+d)←(IY+d)+1	*	*	X	*	X	V	0	–	6	23
	DEC r	00 rrr 101	05+r×8	r←r−1	*	*	X	*	X	V	1	–	1	4
	DEC (HL)	00 110 101	35	(HL)←(HL)−1	*	*	X	*	X	V	1	–	3	11
	DEC (IX+d)	11 011 101 00 110 101 dd ddd ddd	DD 35 d	(IX+d)←(IX+d)−1	*	*	X	*	X	V	1	–	6	23
	DEC (IY+d)	11 111 101 00 110 101 dd ddd ddd	FD 35 d	(IY+d)←(IY+d)−1	*	*	X	*	X	V	1	–	6	23
GENELAL-PURPOSE ARITHMETIC AND MPU CONTROL	DAA	00 100 111	27	Decimal adjust accumulator	*	*	X	*	X	P	–	*	1	4
	CPL	00 101 111	2F	A←$\overline{A}$	–	–	X	1	X	–	1	–	1	4
	NEG	11 101 101 01 000 100	ED 44	A←0−A	*	*	X	*	X	V	1	*	2	8
	CCF	00 111 111	3F	CY←$\overline{CY}$	–	–	X	X	X	–	0	*	1	4
	SCF	00 110 111	37	CY←1	–	–	X	0	X	–	0	1	1	4
	NOP	00 000 000	00	no operation	–	–	X	–	X	–	–	–	1	4
	HALT	01 110 110	76	MPU Halted	–	–	X	–	X	–	–	–	1	4
	DI	11 110 011	F3	IFF←0	–	–	X	–	X	–	–	–	1	4
	EI	11 111 011	FB	IFF←1	–	–	X	–	X	–	–	–	1	4
	IM 0	11 101 101 01 000 110	ED 46	Set interrupt mode 0	–	–	X	–	X	–	–	–	2	8
	IM 1	11 101 101 01 010 110	ED 56	Set interrupt mode 1	–	–	X	–	X	–	–	–	2	8
	IM 2	11 101 101 01 011 110	ED 5E	Set interrupt mode 2	–	–	X	–	X	–	–	–	2	8
16-BIT ARITHMETIC	ADD HL,t	00 tt1 001	09+t×10	HL←HL+t	–	–	X	X	X	–	0	*	3	11
	ADC HL,t	11 101 101 01 tt1 010	ED 4A+t×10	HL←HL+t+CY	*	*	X	X	X	V	0	*	4	15
	SBC HL,t	11 101 101 01 tt0 010	ED 42+t×10	HL←HL−t−CY	*	*	X	X	X	V	1	*	4	15
	ADD IX,p	11 011 101 00 pp1 001	DD 09+p×10	IX←IX+p	–	–	X	X	X	–	0	*	4	15
	ADD IY,s	11 111 101 00 ss1 001	FD 09+s×10	IY←IY+s	–	–	X	X	X	–	0	*	4	15
	INC t	00 tt0 011	03+t×10	t←t+1	–	–	X	–	X	–	–	–	1	6
	INC IX	11 011 101 00 100 011	DD 23	IX←IX+1	–	–	X	–	X	–	–	–	2	10
	INC IY	11 111 101 00 100 011	FD 23	IY←IY+1	–	–	X	–	X	–	–	–	2	10
	DEC t	00 tt1 011	0B+t×10	t←t−1	–	–	X	–	X	–	–	–	1	6
	DEC IX	11 011 101 00 101 011	DD 2B	IX←IX−1	–	–	X	–	X	–	–	–	2	10
	DEC IY	11 111 101 00 101 011	FD 2B	IY←IY−1	–	–	X	–	X	–	–	–	2	10
ROTATE	RLCA	00 000 111	07		–	–	X	0	X	–	0	*	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

t	tt
BC	00
DE	01
HL	10
SP	11

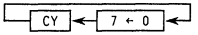
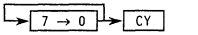
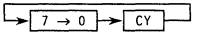
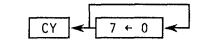
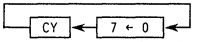
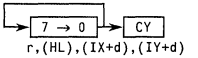
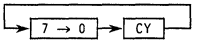
p	pp
BC	00
DE	01
IX	10
SP	11

s	ss
BC	00
DE	01
IY	10
SP	11

Note : ss is any of the register pairs BC, DE, HL, SP. PP is any of the register pairs BC, DE, IX, SP.
rr is any of the register pairs BC, DE, IY, SP.

120489

TMPZ84C015A Instruction Set (6/9)

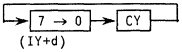
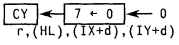
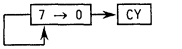
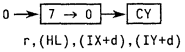
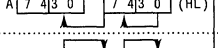
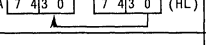
ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
ROTATE SHIFT	RLA	00 010 111	17		–	–	X	0	X	–	0	*	1	4
	RRCA	00 001 111	0F		–	–	X	0	X	–	0	*	1	4
	RRA	00 011 111	1F		–	–	X	0	X	–	0	*	1	4
	RLC r	11 001 011 00 000 rrr	CB 00+r		*	*	X	0	X	P	0	*	2	8
	RLC (HL)	11 001 011 00 000 110	CB 06		*	*	X	0	X	P	0	*	4	15
	RLC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 000 110	DD CB d 06		*	*	X	0	X	P	0	*	6	23
	RLC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 000 110	FD CB d 06		*	*	X	0	X	P	0	*	6	23
	RL r	11 001 011 00 010 rrr	CB 10+r		*	*	X	0	X	P	0	*	2	8
	RL (HL)	11 001 011 00 010 110	CB 16		*	*	X	0	X	P	0	*	4	15
	RL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 010 110	DD CB d 16		*	*	X	0	X	P	0	*	6	23
	RL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 010 110	FD CB d 16		*	*	X	0	X	P	0	*	6	23
	RRC r	11 001 011 00 001 rrr	CB 08+r		*	*	X	0	X	P	0	*	2	8
	RRC (HL)	11 001 011 00 001 110	CB 0E		*	*	X	0	X	P	0	*	4	15
	RRC (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 001 110	DD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RRC (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 001 110	FD CB d 0E		*	*	X	0	X	P	0	*	6	23
	RR r	11 001 011 00 011 rrr	CB 18+r		*	*	X	0	X	P	0	*	2	8
	RR (HL)	11 001 011 00 011 110	CB 1E		*	*	X	0	X	P	0	*	4	15
	RR (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 011 110	DD CB d 1E		*	*	X	0	X	P	0	*	6	23

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

Note : r means any of the registers A, B, C, D, E, H, L.

120489

TMPZ84C015A Instruction Set (7/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary 76 543 210	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES	
ROTATE SHIFT	RR (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 011 110	FD CB d 1E	 (IY+d)	*	*	X	0	X	P	0	*	6	23	
	SLA r	11 001 011 00 100 rrr	CB 20+r		*	*	X	0	X	P	0	*	2	8	
	SLA (HL)	11 001 011 00 100 110	CB 26		*	*	X	0	X	P	0	*	4	15	
	SLA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 100 110	DD CB d 26	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23	
	SLA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 100 110	FD CB d 26		*	*	X	0	X	P	0	*	6	23	
	SRA r	11 001 011 00 101 rrr	CB 28+r		*	*	X	0	X	P	0	*	2	8	
	SRA (HL)	11 001 011 00 101 110	CB 2E		*	*	X	0	X	P	0	*	4	15	
	SRA (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 101 110	DD CB d 2E	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23	
	SRA (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 101 110	FD CB d 2E		*	*	X	0	X	P	0	*	6	23	
	SRL r	11 001 011 00 111 rrr	CB 38+r		*	*	X	0	X	P	0	*	2	8	
	SRL (HL)	11 001 011 00 111 110	CB 3E		*	*	X	0	X	P	0	*	4	15	
	SRL (IX+d)	11 011 101 11 001 011 dd ddd ddd 00 111 110	DD CB d 3E	 r,(HL),(IX+d),(IY+d)	*	*	X	0	X	P	0	*	6	23	
	SRL (IY+d)	11 111 101 11 001 011 dd ddd ddd 00 111 110	FD CB d 3E		*	*	X	0	X	P	0	*	6	23	
	RLD	11 101 101 01 101 111	ED 6F		*	*	X	0	X	P	0	–	5	18	*1
	RRD	11 101 101 01 100 111	ED 67		*	*	X	0	X	P	0	–	5	18	*1
BIT SET RESET AND TEST	BIT b,r	11 001 011 01 bbb rrr	CB 40+b×8+r	$Z \leftarrow \overline{r_b}$	X	*	X	1	X	X	0	–	2	8	
	BIT b,(HL)	11 001 011 01 bbb 110	CB 46+b×8	$Z \leftarrow \overline{(HL)_b}$	X	*	X	1	X	X	0	–	3	12	

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Note : *1 : Rotate digit left and right between the accmulator and location (HL).
The content of the upper half of the accumulator is unaffected.
The notation $(HL)_b$ indicates bit $_b$ (0 to 7) within the contents of the HL register pair.
The notation r_b indicates bit $_b$ (0 to 7) within the r register.

120489

TMPZ84C015A Instruction Set (8/9)

ITEM/ CLASSI -FICA- TION	Assembler mnemonic	Object code Binary	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CY- CLES	No. OF STA- TES
		76 543 210												
BIT SET RESET AND TEST	BIT b,(IX+d)	11 011 101	DD		X	*	X	1	X	X	0	–	5	20
		11 001 011	CB	$Z \leftarrow \overline{(IX+d)_b}$										
		dd ddd ddd	d											
		01 bbb 110	46+b×8											
	BIT b,(IY+d)	11 111 101	FD		X	*	X	1	X	X	0	–	5	20
		11 001 011	CB	$Z \leftarrow \overline{(IY+d)_b}$										
		dd ddd ddd	d											
		01 bbb 110	46+b×8											
	SET b,r	11 001 011	CB	$r_b \leftarrow 1$	–	–	X	–	X	–	–	–	2	8
		11 bbb rrr	C0+b×8+r											
	SET b,(HL)	11 001 011	CB	$(HL)_b \leftarrow 1$	–	–	X	–	X	–	–	–	4	15
		11 bbb 110	C6+b×8											
	SET b,(IX+d)	11 011 101	DD	$(IX+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
		11 001 011	CB											
		dd ddd ddd	d											
		11 bbb 110	C6+b×8											
	SET b,(IY+d)	11 111 101	FD	$(IY+d)_b \leftarrow 1$	–	–	X	–	X	–	–	–	6	23
		11 001 011	CB											
		dd ddd ddd	d											
		11 bbb 110	C6+b×8											
	RES b,r	11 001 011	CB	$r_b \leftarrow 0$	–	–	X	–	X	–	–	–	2	8
		10 bbb rrr	80+b×8+r											
	RES b,(HL)	11 001 011	CB	$(HL)_b \leftarrow 0$	–	–	X	–	X	–	–	–	4	15
		10 bbb 110	86+b×8											
	RES b,(IX+d)	11 011 101	DD	$(IX+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
		11 001 011	CB											
		dd ddd ddd	d											
		10 bbb 110	86+b×8											
	RES b,(IY+d)	11 111 101	FD	$(IY+d)_b \leftarrow 0$	–	–	X	–	X	–	–	–	6	23
		11 001 011	CB											
		dd ddd ddd	d											
		10 bbb 110	86+b×8											
JUMP	JP mn	11 000 011	C3	PC←mn	–	–	X	–	X	–	–	–	3	10
		nn nnn nnn	n											
		mm mmm mmm	m											
	JP c,mn	11 ccc 010	C2+c×8	PC←mn	–	–	X	–	X	–	–	–	3	10
		nn nnn nnn	n	(Only when condition is met)										
		mm mmm mmm	m											
	JR $+e	00 011 000	18	PC←$+e	–	–	X	–	X	–	–	–	3	12
		aa aaa aaa	a											
	JR C,$+e	00 111 000	38	If C=0, continue	–	–	X	–	X	–	–	–	2	7
		aa aaa aaa	a	If C=1, PC←$+e	–	–	X	–	X	–	–	–	3	12
	JR NC,$+e	00 110 000	30	If C=0, PC←$+e	–	–	X	–	X	–	–	–	3	12
		aa aaa aaa	a	If C=1, continue	–	–	X	–	X	–	–	–	2	7
	JR Z,$+e	00 101 000	28	If Z=0, continue	–	–	X	–	X	–	–	–	2	7
		aa aaa aaa	a	If Z=1, PC←$+e	–	–	X	–	X	–	–	–	3	12
	JR NZ,$+e	00 100 000	20	If Z=0, PC←$+e	–	–	X	–	X	–	–	–	3	12
		aa aaa aaa	a	If Z=1, continue	–	–	X	–	X	–	–	–	2	7
	DJNZ $+e	00 010 000	10	B←B−1, If B=0, continue	–	–	X	–	X	–	–	–	2	8
		aa aaa aaa	a	B←B−1, If B<>0, continue	–	–	X	–	X	–	–	–	3	13
	JP (HL)	11 101 001	E9	PC←HL	–	–	X	–	X	–	–	–	1	4

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

b	bbb
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

e represents the extension in the relative addressing mode, a = e − 2. e is a signed two's complement number in the range of $-126 \leq e \leq 129$

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign Negative

Note :
- a = e − 2 in the Opcode provides an effective address of PC + e as PC is incremented by 2 before the addition of e.
- $ indicates the reference to the location counter value of the current segment.
- The notation $(HL)_b$, $(IX + d)_b$ indicates bit $_b$ (0 to 7) within the contents of the register pair.
- The notation r_b indicates bit $_b$ (0 to 7) within the r register.
- a = e-2 in the op-code provides effective address of PC + e as PC is incremented by 2 prior to the addition of e.

270489

TMPZ84C015A Instruction Set (9/9)

ITEM/CLASSIFICATION	Assembler mnemonic	Object code Binary (76 543 210)	Object code Hex	Function	Flag S	Z		H		P/V	N	C	No. OF CYCLES	No. OF STATES
JUMP	JP (IX)	11 011 101 11 101 001	DD E9	PC←(IX)	–	–	X	–	X	–	–	–	2	8
	JP (IY)	11 111 101 11 101 001	FD E9	PC←(IY)	–	–	X	–	X	–	–	–	2	8
CALL AND RETURN	CALL mn	11 001 101 nn nnn nnn mm mmm mmm	CD n m	(SP−1)←PC_H, (SP−2)←PC_L PC←mn SP←SP−2	–	–	X	–	X	–	–	–	5	17
	CALL c,mn	11 ccc 100 nn nnn nnn mm mmm mmm	C4+cX8 n m	If condition c is met, same as CALL mn. If condition c is not met, continue	– –	– –	X X	– –	X X	– –	– –	– –	5 3	17 10
	RET	11 001 001	C9	PC_L←(SP), PC_H←(SP+1) SP←SP+2	–	–	X	–	X	–	–	–	3	10
	RET c	11 ccc 000	C0+c×8	If condition c is met, same as RET. If condition c is not met, continue	– –	– –	X X	– –	X X	– –	– –	– –	3 1	11 5
	RETI	11 101 101 01 001 101	ED 4D	Return from interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RETN	11 101 101 01 000 101	ED 45	Return from non-maskable interrupt Processing routine	–	–	X	–	X	–	–	–	4	14
	RST j	11 kkk 111	C7+k×8	(SP−1)←PC_H, (SP−2)←PC_L PC_H←0, PC_L←j, SP←SP−2	–	–	X	–	X	–	–	–	3	11
INPUT AND OUTPUT	IN A,(n)	11 011 011 nn nnn nnn	DB n	A←(n) n→A0~A7, A→A8~A15	–	–	X	–	X	–	–	–	3	11
	IN r,(C)	11 101 101 01 rrr 000	ED 40+rx8	r←(C) If r=110, only the flags will be affected.	*	*	X	*	X	P	0	–	3	12
	INI	11 101 101 10 100 010	ED A2	(HL)←(C), B←B−1, HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	INIR	11 101 101 10 110 010	ED B2	(HL)←(C), B←B−1, HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	IND	11 101 101 10 101 010	ED AA	(HL)←(C), B←B−1, HL←HL−1	X	*M	X	X	X	X	1	X	4	16
	INDR	11 101 101 10 111 010	ED BA	(HL)←(C), B←B−1, HL←HL−1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUT (n),A	11 010 011 nn nnn nnn	D3 n	(n)←A n→A0~A7, A→A8~A15	–	–	X	–	X	–	–	–	3	11
	OUT (C),r	11 101 101 01 rrr 001	ED 41+rX8	(C)←r	–	–	X	–	X	–	–	–	3	12
	OUTI	11 101 101 10 100 011	ED A3	(C)←(HL), B←B−1, HL←HL+1	X	*M	X	X	X	X	1	X	4	16
	OTIR	11 101 101 10 110 011	ED B3	(C)←(HL), B←B−1, HL←HL+1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16
	OUTD	11 101 101 10 101 011	ED AB	(C)←(HL), B←B−1, HL←HL−1	X	*M	X	X	X	X	1	X	4	16
	OTDR	11 101 101 10 111 010	ED BB	(C)←(HL), B←B−1, HL←HL−1 Repeat until B=0	X	1	X	X	X	X	1	X	5 4	21 16

j	kkk
00H	000
08H	001
10H	010
18H	011
20H	100
28H	101
30H	110
38H	111

r	rrr
B	000
C	001
D	010
E	011
H	100
L	101
A	111

*1 ←[B<>0] ←[B=0]

*1 ←[B<>0] ←[B=0]

*1 ←B[<>0] ←[B=0]

*1 ←[B<>0] ←[B=0]

Note :
- *M If the result of B − 1 is zero, the Z flag is set, otherwise it is reset.
- A0 through A15 indicate the address bus.
- [ ] indicates the total condition of the number of cycles and states indicated by arrow.

*1 C→A0~A7
B→A8~A15

c	ccc	Condition
NZ	000	Non-Zero
Z	001	Zero
NC	010	No-Carry
C	011	Carry
PO	100	Odd Parity
PE	101	Even Parity
P	110	Sign Positive
M	111	Sign negative

120489

TMPZ84C015A Instruction Map (1/7)

MPU Instruction Table (I)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	LD BC, mn	LD (BC), A	INC BC	INC B	DEC B	LD B, n	RLCA	EX AF,AF	ADD HC,BC	LD A, (BC)	DEC BC	INC C	DEC C	LD C, n	RRCA
1	DJNZ e	LD DE, mn	LD (DE), A	INC DE	INC D	DEC D	LD D,N	RLA	JE e	ADD HL,DE	LD A, (DE)	DEC DE	INC E	DEC E	LD E, n	RRA
2	JR NZ, e	LD HL, mn	LD (mn),HL	INC HL	INC D	DEC H	LD H, n	DAA	JR Z e	ADD HL,HL	LD HL,(mn)	DEC HL	INC L	DEC L	LD L, n	CPL
3	JR NC, e	LD SP, mn	LD (mn),A	INC SP	INC (HL)	DEC (HL)	LD (HL), n	SCF	JR C e	ADD HL,SP	LD A,(mn)	DEC SP	INC A	DEC A	LD A, n	CCF
4	LD B, B	LD B, C	LD B, D	LD B, E	LD B, H	LD B, L	LD B, (HL)	LD B, A	LD C, B	LD C, C	LD C, D	LD C, E	LD C, H	LD C, L	LD C, (HL)	LD C, A
5	LD D, B	LD D, C	LD D, D	LD D, E	LD D, H	LD D, L	LD D, (HL)	LD D, A	LD E, B	LD E, C	LD E, D	LD E, E	LD E, H	LD E, L	LD E, (HL)	LD E, A
6	LD H, B	LD H, C	LD H, D	LD H, E	LD H, H	LD H, L	LD H, (HL)	LD H, A	LD L, B	LD L, C	LD L, D	LD L, E	LD L, H	LD L, L	LD L, (HL)	LD L, A
7	LD (HL), B	LD (HL), C	LD (HL), D	LD (HL), E	LD (HL), H	LD (HL), L	HALT	LD (HL), A	LD A, B	LD A, C	LD A, D	LD A, E	LD A, H	LD A, L	LD A, (HL)	LD A, A
8	ADD A, B	ADD A, C	ADD A, D	ADD A, E	ADD A, H	ADD A, L	ADD A, (HL)	ADD A, A	ADC A, B	ADC A, C	ADC A, D	ADC A, E	ADC A, H	ADC A, L	ADC A, (HL)	ADC A, A
9	SUB B	SUB C	SUB D	SUB E	SUB H	SUB L	SUB (HL)	SUB A	SBC A, B	SBC A, C	SBC A, D	SBC A, E	SBC A, H	SBC A, L	SBC A, (HL)	SBC A, A
A	AND B	AND C	AND D	AND E	AND H	AND L	AND (HL)	AND A	XOR B	XOR C	XOR D	XOR E	XOR H	XOR L	XOR (HL)	XOR A
B	OR B	OR C	OR D	OR E	OR H	OR L	OR (HL)	OR A	CP B	CP C	CP D	CP E	CP H	CP L	CP (HL)	XOR A
C	RET NZ	POP BC	JP NZ, mn	JP mn	CALL NZ, mn	PUSH BC	ADD A, n	RST 00H	RET C	RET	JP Z, mn	①	CALL Z, mn	CALL mn	ADC A, n	RST 08H
D	RET NC	POP DE	JP NC, mn	OUT (n), A	CALL NC, mn	PUSH DE	SUB n	RST 10H	RET C	EXX	JP C, mn	IN A, (n)	CALL C, mn	③	SBC A, n	RST 18H
E	RET PO	POP HL	JP PO, mn	EX (SP),HL	CALL PO, mn	PUSH HL	AND n	RST 20H	RET PE	JP (HL)	JP PE, mn	EX DE,HL	CALL PI, mn	②	XOR n	RST 28H
F	RET P	POP AF	JP P, mn	DI	CALL P, mn	PUSH AF	OR n	RST 30H	RET M	LD SP, HL	JP M, mn	EI	CALL M, mn	④	CP n	RST 38H

120489

Note ①~④: Multi-Opcode Instructions (ref. Table (II)~(VII))

TMPZ84C015A Instruction Map (2/7)

① Byte 1 "CB"

Instruction Table (II) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	RLC B	RLC C	RLC D	RLC E	RLC H	RLC L	RLC (HL)	RLC A	RRC B	RRC C	RRC D	RRC E	RRC H	RRC L	RRC (HL)	RRC A
1	RL B	RL C	RL D	RL E	RL H	RL L	RL (HL)	RL A	RR B	RR C	RR D	RR E	RR H	RR L	RR (HL)	RR A
2	SLA B	SLA C	SLA D	SLA E	SLA H	SLA L	SLA (HL)	SLA A	SRA B	SRA C	SRA D	SRA E	SRA H	SRA L	SRA (HL)	SRA A
3									SRL B	SRL C	SRL D	SRL E	SRL H	SRL L	SRL (HL)	SRL A
4	BIT 0, B	BIT 0, C	BIT 0, D	BIT 0, E	BIT 0, H	BIT 0, L	BIT 0,(HL)	BIT 0, A	BIT 1, B	BIT 1, C	BIT 1, D	BIT 1, E	BIT 1, H	BIT 1, L	BIT 1, (HL)	BIT 1, A
5	BIT 2, B	BIT 2, C	BIT 2, D	BIT 2, E	BITT 2, H	BIT 2, L	BIT 2, (HL)	BIT 2, A	BIT 3, B	BIT 3, C	BIT 3, D	BIT 3, E	BIT 3, H	BIT 3, L	BIT 3, (HL)	BIT 3, A
6	BIT 4, B	BIT 4, C	BIT 4, D	BIT 4, E	BIT 4, H	BIT 4, L	BIT 4, (HL)	BIT 4, A	BIT 5, B	BIT 5, C	BIT 5, D	BIT 5, E	BIT 5, H	BIT 5, L	BIT 5, (HL)	BIT 5, A
7	BIT 6, B	BIT 6, C	BIT 6, D	BIT 6, E	BIT 6, H	BIT 6, L	BIT 6, (HL)	BIT 6, A	BIT 7, B	BIT 7, C	BIT 7, D	BIT 7, E	BIT 7, H	BIT 7, L	BIT 7, (HL)	BIT 7, A
8	RES 0, B	RES 0, C	RES 0, D	RES 0, E	RES 0, H	RES 0, L	RES 0, (HL)	RES 0, A	RES 1, B	RES 1, C	RES 1, D	RES 1, E	RES 1, H	RES 1, L	RES 1, (HL)	RES 1, A
9	RES 2, B	RES 2, C	RES 2, D	RES 2, E	RES 2, H	RES 2, L	RES 2, (HL)	RES 2, A	RES 3, B	RES 3, C	RES 3, D	RES 3, E	RES 3, H	RES 3, L	RES 3, (HL)	RES 3, A
A	RES 4, B	RES 4, C	RES 4, D	RES 4, E	RES 4, H	RES 4, L	RES 4, (HL)	RES 4, A	RES 5, B	RES 5, C	RES 5, D	RES 5, E	RES 5, H	RES 5, L	RES 5, (HL)	RES 5, A
B	RES 6, B	RES 6, C	RES 6, D	RES 6, E	RES 6, H	RES 6, L	RES 6, (HL)	RES 6, A	RES 7, B	RES 7, C	RES 7, D	RES 7, E	RES 7, H	RES 7, L	RES 7, (HL)	RES 7, A
C	SET 0, B	SET 0, C	SET 0, D	SET 0, E	SET 0, H	SET 0, L	SET 0, (HL)	SET 0, A	SET 1, B	SET 1, C	SET 1, D	SET 1, E	SET 1, H	SET 1, L	SET 1, (HL)	SET 1, A
D	SET 2, B	SET 2, C	SET 2, D	SET 2, E	SET 2, H	SET 2, L	SET 2, (HL)	SET 2, A	SET 3, B	SET 3, C	SET 3, D	SET 3, E	SET 3, H	SET 3, L	SET 3, (HL)	SET 3, A
E	SET 4, B	SET 4, C	SET 4, D	SET 4, E	SET 4, H	SET 4, L	SET 4, (HL)	SET 4, A	SET 5, B	SET 5, C	SET 5, D	SET 5, E	SET 5, H	SET 5, L	SET 5, (HL)	SET 5, A
F	SET 6, B	SET 6, C	SET 6, D	SET 6, E	SET 6, H	SET 6, L	SET 6, (HL)	SET 6, A	SET 7, B	SET 7, C	SET 7, D	SET 7, E	SET 7, H	SET 7, L	SET 7, (HL)	SET 7, A

120489

TMPZ84C015A Instruction Map (3/7)

② Byte 1 "ED"

Instruction Table (III) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4	IN B, (C)	OUT (C), B	SBC HL,BC	LD (mn),BC	NEG	RETN	IM0	LD I, A	IN C, (C)	OUT (C), C	ADC HL,BC	LD BC,(mn)		RETI		LD R, A
5	IN D, (C)	OUT (C), D	SBC HL,DE	LD (mn),DE			IM1	LD A, I	IN E, (C)	OUT (C), E	ADC HL,DE	LD DE,(mn)			IM2	LD A, R
6	IN H, (C)	OUT (C), H	SBC HL,HL	LD (mn),HL				RRD	IN L, (C)	OUT (C), L	ADC HL,HL	LD HL,(mn)				RLD
7			SBC HL,SP	LD (mn),SP					IN A, (C)	OUT (C), A	ADC HL,SP	LD SP,(mn)				
8																
9																
A	LDI	CPI	INI	OUTI					LDD	CPD	IND	OUTD				
B	LDIR	CPIR	INIR	OTIR					LDDR	CPDR	INDR	OTDR				
C																
D																
E																
F																

120489

TMPZ84C015A Instruction Map (4/7)

③ Byte 1 "DD"

Instruction Table (Ⅳ) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0										ADD IX,BC						
1										ADD IX,DE						
2		LD IX, mn	LD (mn), IX	INC IX						ADD IX, IX	LD IX,(mn)	DEC IX				
3					INC (IX + d)	DEC (IX + d)	LD (IX + d) , n			ADD IX, SP						
4							LD B, (IX + d)								LD C, (IX + d)	
5							LD D, (IX + d)								LD E, (IX + d)	
6							LD H, (IX + d)								LD L, (IX + d)	
7	LD (IX + d) , B	LD (IX + d) , C	LD (IX + d) , D	LD (IX + d) , E	LD (IX + d) , H	LD (IX + d) , L		LD (IX + d) , A							LD A, (IX + d)	
8							ADD (IX + d)								ADC 1, (IX + d)	
9							SUB (IX + d)								SBC A, (IX + d)	
8							AND (IX + d)								XOR (IX + d)	
9							OR (IX + d)								CP (IX + d)	
C												⑤				
D																
E		POP IX		EX (SP), IX		PUSH IX				JP (IX)						
F										LD SP, IX						

120489

Note ⑤: Special 2 byte Opcode Instructions (ref. Table (Ⅵ))

TMPZ84C015A Instruction Map (5/7)

④ Byte 1 "FD"

Instruction Table (Ⅴ) (Byte 2 of 2-byte Opcode)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0										ADD IY,BC						
1										ADD IY,DE						
2		LD IY, nn	ID (mn), IY	INC IY						ADD IY, IY	LD IY, (mn)	DEC IY				
3					INC (IY + d)	DEC (IY + d)	LD (IY + d) , n			ADD IY, SP						
4							LD B, (IY + d)								LD C, (IY + d)	
5							LD D, (IY + d)								LD E, (IY + d)	
6							LD H, (IY + d)								LD L, (IY + d)	
7	LD (IY + d) , B	LD (IY + d) , C	LD (IY + d) , D	LD (IY + d) , E	LD (IY + d) , H	LD (IY + d) , L		LD (IY + d) , A							LD A, (IY + d)	
8							ADD (IY + d)								ADDC A, (IY + d)	
9							SUB (IY + d)								SUBC A, (IY + d)	
8							AND (IY + d)								XOR (IY + d)	
9							OR (IY + d)								CP (IY + d)	
C												⑥				
D																
E		POP IY		EX (SP), IY		PUSH IY				JP (IY)						
F										LD SP, IY						

120489

Note ⑥: Special 2 byte Opcode Instructions (ref. Table (Ⅶ))

TMPZ84C015A Instruction Map (6/7)

⑤ Byte 1 "DD"

Byte 2 "CB"

Instruction Table (VI) (Special case of 2-byte Opcode : Byte 3)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0							RLC (IX + d)								RRC (IX + d)	
1							RL (IX + d)								RR (IX + d)	
2							SLA (IX + d)								SRA (IX + d)	
3															SRL (IX + d)	
4							BIT 0, (IX + d)								BIT 1, (IX + d)	
5							BIT 2, (IX + d)								BIT 3, (IX + d)	
6							BIT 4, (IX + d)								BIT 5, (IX + d)	
7							BIT 6, (IX + d)								BIT 7, (IX + d)	
8							RES 0, (IX + d)								RES 1, (IX + d)	
9							RES 2, (IX + d)								RES 3, (IX + d)	
A							RES 4, (IX + d)								RES 5, (IX + d)	
B							RES 6, (IX + d)								RES 7, (IX + d)	
C							SET 0, (IX + d)								SET 1, (IX + d)	
D							SET 2, (IX + d)								SET 3, (IX + d)	
E							SET 4, (IX + d)								SET 5, (IX + d)	
F							SET 6, (IX + d)								SET 7, (IX + d)	

120489

TMPZ84C015A Instruction Map (7/7)

⑥ Byte 1 "FD"

Byte 2 "CB"

Instruction Table (VII) (Special case of 2-byte Opcode : Byte 3)

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0							RLC (IY + d)								RRC (IY + d)	
1							RL (IY + d)								RR (IY + d)	
2							SLA (IY + d)								SRA (IY + d)	
3															SRL (IY + d)	
4							BIT 0, (IY + d)								BIT 1, (IY + d)	
5							BIT 2, (IY + d)								BIT 3, (IY + d)	
6							BIT 4, (IY + d)								BIT 5, (IY + d)	
7							BIT 6, (IY + d)								BIT 7, (IY + d)	
8							RES 0, (IY + d)								RES 1, (IY + d)	
9							RES 2, (IY + d)								RES 3, (IY + d)	
A							RES 4, (IY + d)								RES 5, (IY + d)	
B							RES 6, (IY + d)								RES 7, (IY + d)	
C							SET 0, (IY + d)								SET 1, (IY + d)	
D							SET 2, (IY + d)								SET 3, (IY + d)	
E							SET 4, (IY + d)								SET 5, (IY + d)	
F							SET 6, (IY + d)								SET 7, (IY + d)	

120489

3.3 CGC OPERATIONS

This subsection describes the system configuration, functions, and basic operations of the clock generator/controller (CGC).

3.3.1 Block Diagram

Figure 3.3.1 shows the block diagram of CGC.

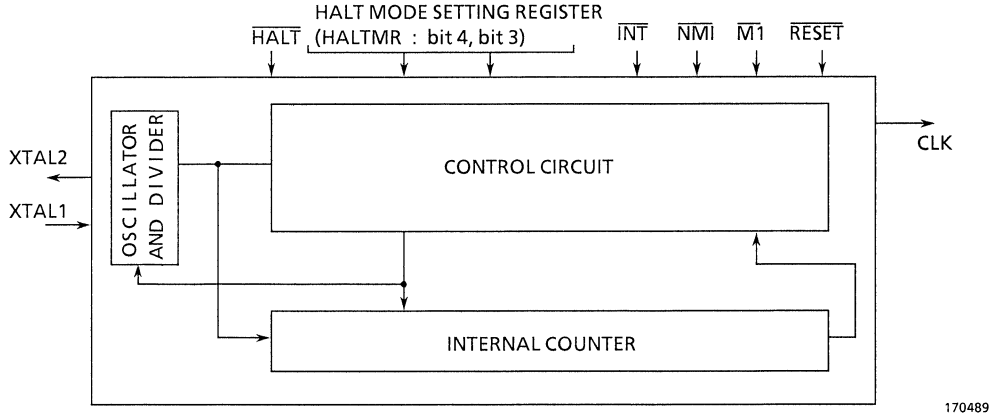


Figure 3.3.1 Block Diagram

3.3.2 CGC System Configuration

The internal configuration of the CGC is shown in Figure 3.3.1. The waveform from the external oscillator oscillated by the internal oscillator and divided by the divider is converted into the square wave for clock. The clock is controlled by the control circuit and the counter to be sent to the outside the CGC. The following describes the CGC's main components and their functions.

(1) Clock Generation

(2) Operation Modes

[1] Clock Generation

The CGC contains an oscillation circuit. By connecting oscillator to external pins (XTAL1 and XTAL2), the required clock can be generated easily. The CGC provides the clock whose frequency is 1/2 of the osicillation frequency. Figure 3.3.2 shows an example of oscillator connection.

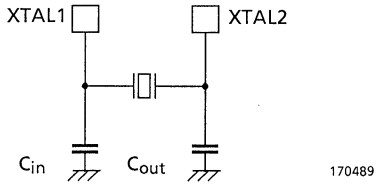


Figure 3.3.2 (a) Example of Crystal Connection

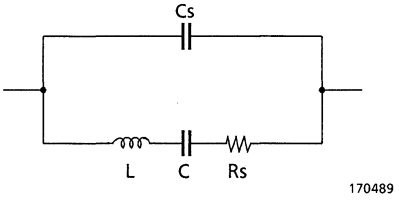


Figure 3.3.2 (b) Oscillator Equivalent Circuit

(1) For the quartz crystal oscillator, use the MR8000-C20 (oscillation frequency 8 MHz) or MR12000-C20 (oscillation frequency 12 MHz) manufactured by Tokyo Denpa Company Ltd., or the equivalent;

Product No.	Holder Type	Frequency (MHz)	Cin (pF)	Cout (pF)	Quartz Crystal Parameter (Typ.)			Drive Level (mW)	Condition Load Capacitance (pF)
					C_1 (pF)	C_0 (pF)	R_1 (Ω)		
MR8000-C20	HC-49 -U (TR-49)	8	22	33	—	4.00	30.0	—	—
MR8000-C14		8	20	20	0.0189	3.87	6.0	0.5	12.67
MR12000-C20		12	33	33	—	4.00	25.0	—	—
MR12000-C14		12	20	20	0.0190	3.81	6.9	0.5	12.55
MR16000-C14		16	20	20	0.0197	4.00	5.7	0.5	12.20

190589

Note : The load capacitance in the condition does not include any stray capacitance.

(2) For the ceramic resonater, use the CSA8.00MT, CST8.00MT (oscillation frequency 8 MHz) or CSA12.00MT, CST12.00MT (osillation frequency 12 MHz) manufactured by Murata MFG Co., Ltd.

Product No.	Frequency (MHz)	Cin (pF)	Cout (pF)
CSA8.00MT	8	30	30
CST8.00MT	8		
CSA12.0MT	12	30	30
CST12.0MT	12		

190589

Note : The CST8.00MT and CST12.0MT need no outer capacitance.

[2] Operation Modes

The CGC has the capability to control 4 operation modes; Run, Idle 1, Idle 2, and Stop. Any one of them can be selected through the mode setting register (#F0: bit4, bit3: HALTMR). These modes become valid when the MPU executes a HALT instruction. Fetching a HALT instruction, the MPU sets the HALT signal to "0", indicating that it has been put in the halt state. After the execution of the $\overline{\text{HALT}}$ instruction, the CGC performs the operation in the specified mode. Table 3.3.1 shows the operations in each mode.

Table 3.3.1 CGC Operation Modes

Halt mode setting register (#F0:HALTMR)		Operational Mode	Description
Bit 4	Bit 3		
0	0	IDLE1 Mode	Only the internal oscillator operates, stopping the supply of clock outside. The clock output (CLKOUT) is held at "0".
0	1	IDLE2 Mode	The internal oscillator continues operating with the supply of clock outside (CLKOUT) continued. When pins CLKOUT and CLKIN are connected, only the supply of clock (CLKOUT) to the CTC is continued.
1	0	STOP Mode	All internal operations are stopped. The clock output (CLK) is held at "0".
1	1	RUN Mode	The supply of clock outside is continued.

170489

The restart from the clock stop state in Idle 1, Idle 2 (these two modes are referred to as Idle mode hereafter), or Stop mode is performed by reset ($\overline{\text{RESET}}$ signal) or acknowledge of maskable interrupt ($\overline{\text{INT}}$ signal) or non-maskable interrupt ($\overline{\text{NMI}}$ signal).

[3] Warm-up Time for Restart (from Stop mode)

Releasing the halt state by interrupt acknowledge, the MPU begins executing interrupt processing. Therefore, when restarting the clock by the $\overline{NMI}$ or $\overline{INT}$ restart signal in the Stop mode, the oscillation must be fully stabilized before supplied outside. The CGC provides, by means of the internal counter, the warm-up time enough for the clock to stabilize frequency. The warm-up ends on the rising edge of the internal counter output dividing the oscillation frequency to start clock output. The warm-up time is equal to the time derived by dividing the frequency of the externally attached oscillator by 2^{14}.

Figure 3.3.3 shows the block diagram of the internal counter. Table 3.3.2 shows the relationship between the oscillation frequency and the warm-up time.

In the restart by the $\overline{RESET}$ signal, no warm-up is performed for the quick operation at power-on. Therefore, expand the width of the $\overline{RESET}$ signal adequately to provide the warm-up time.

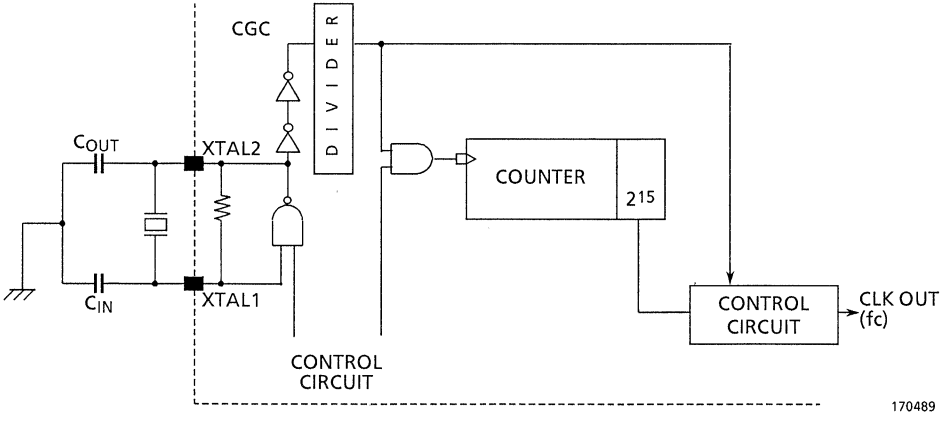


Figure 3.3.3 Block Diagram of Internal Counter

Table 3.3.2 Warm-up Time

Counter output	Warm-up Time		
		fXTAL = 12MHz	fXTAL = 8MHz
2^{15}	2^{14} / fc	2.7 ms	4 ms

170489

* $fc = f_{XTAL} / 2$

3.3.3 CGC Status Transition Diagram and Basic Timing

The following describes the status transition and basic timing to be provided when the CGC operates.

[1] Status Transition Diagram

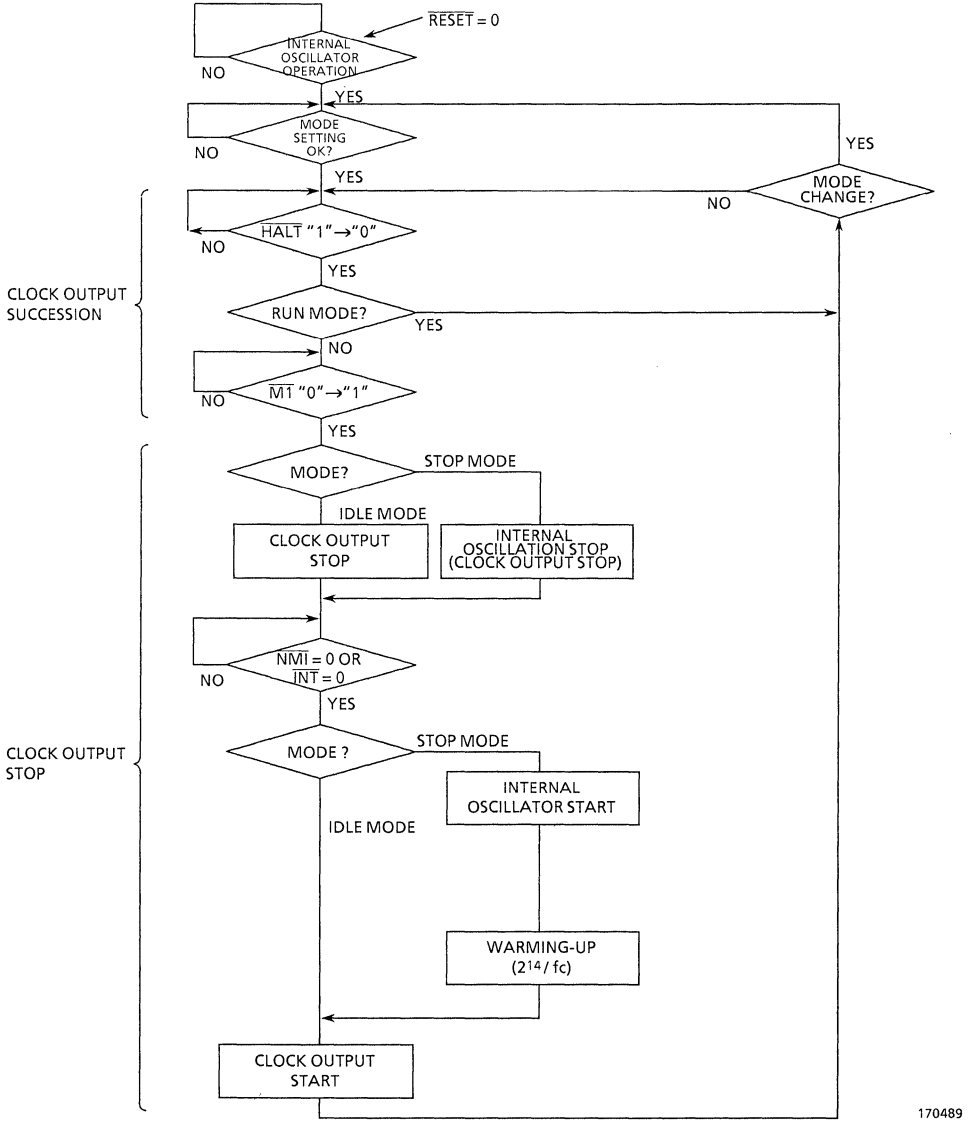


Figure 3.3.4 Status Transition Diagram

[2] Basic Timing

The following describes the CGC basic timing when the CGC clock output pin (CLKOUT) and clock input pin (CLKIN) are connected.

(1) Operation at execution of HALT instruction

The following describes the basic timing in each mode to be provided when the MPU executes a HALT instruction. The MPU sets the $\overline{\text{HALT}}$ signal to "0" synchronized with the falling edge of clock state T4 of the HALT Instruction Opcode fetch cycle (M1). This signal indicates to the CGC that the MPU is going to enter into the halt state.

(a) Run mode (#F0: bit 4 = 1, bit 3 = 1: HALTMR)

Figure 3.3.5 shows the basic timing in the Run mode. In the Run mode, the CGC continues supplying the clock to the outside even when the MPU is in the halt state. Therefore, the MPU continues executing NOPs during the halt state. The systems which need memory address refresh use this mode.

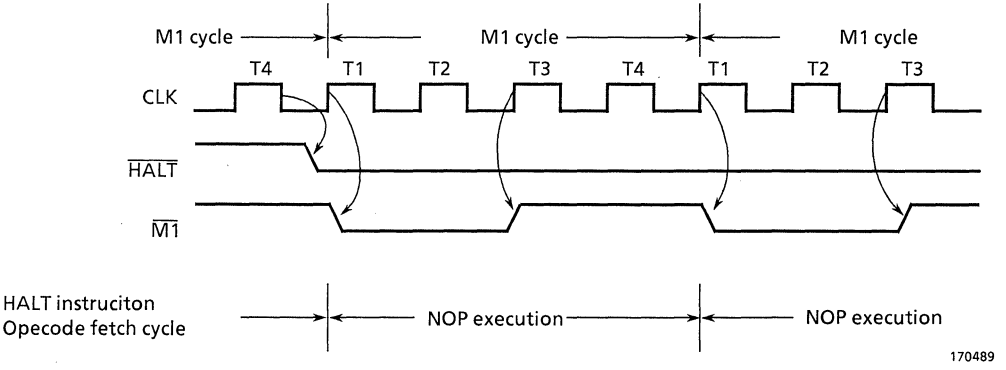


Figure 3.3.5 Basic Timing in Run Mode

(b) Idle 1 mode (#F0: bit 4 = 0, bit 3 = 0: HALTMR), idle 2 mode (#F0: bit 4 = 0, bit 3 = 1: HALTMR), and Stop mode (#F0: bit 4 = 1, bit 3 = 0)

Figure 3.3.6 shows the basic timing in the Idle modes and Stop mode. In these modes, the clock output is stopped with clock state T4 being "0" by the $\overline{\text{HALT}}$ signal and the $\overline{\text{M1}}$ signal which follows the HALT instruction.

However, in the Stop mode, the CGC's internal oscillator also stops.

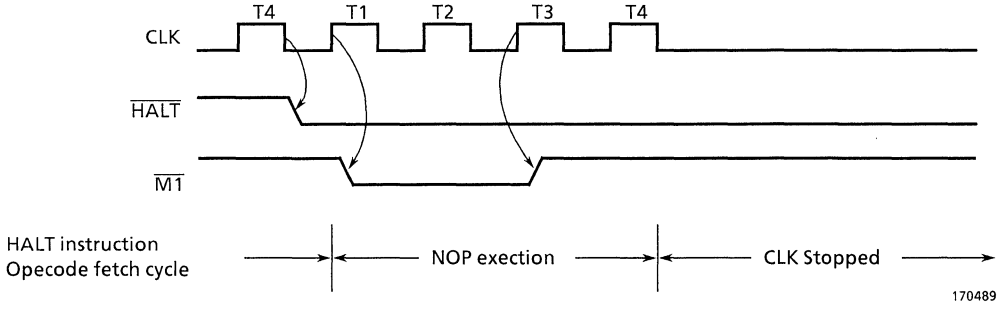


Figure 3.3.6 Basic Timing in Idle and Stop Modes

(2) Clock output restart from each mode

The clock stopped state in the Idle or Stop mode is cleared by setting any of the following signals to "0" (for the system restart operation, see Subsection 3.3.4) :

- $\overline{\text{INT}}$ (level trigger input)
- $\overline{\text{NMI}}$ (edge trigger input)
- $\overline{\text{RESET}}$ (level trigger input)

(a) Clock output restart from Idle mode

Figure 3.3.7 (a) shows the basic timing for the sequence of the output restart from the clock stopped state in the Idle 1 mode. In the restart in the Idle 1 mode, the clock output is restarted in a relatively short delay time because the internal oscillator operates even in the clock stopped state.

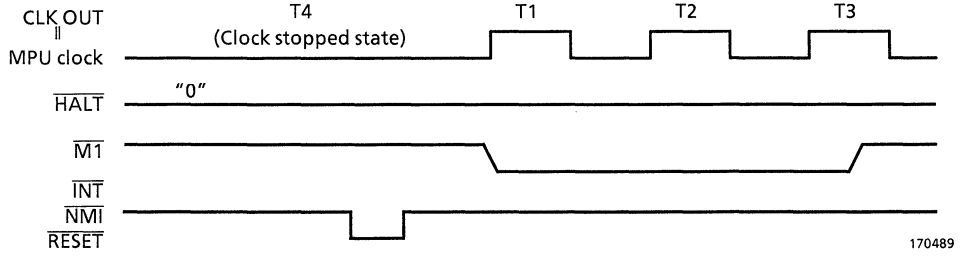
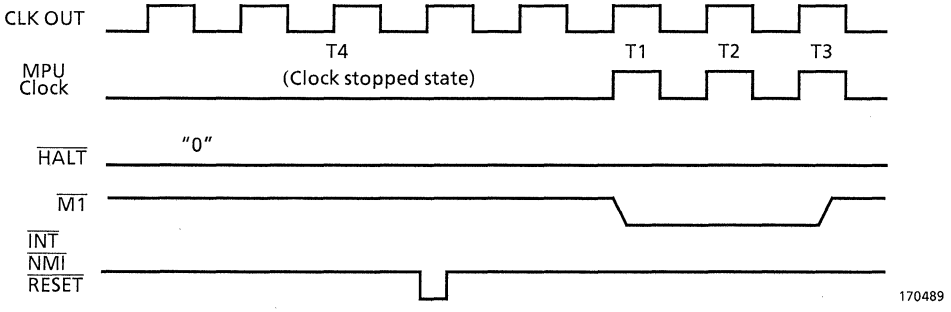


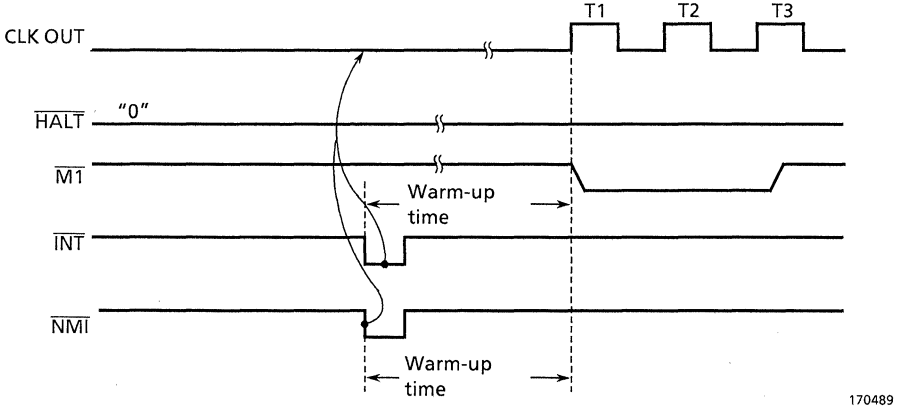
Figure 3.3.7 (a) Basic Timing for Sequence of Restart from Clock stopped State (Idle 1 Mode)

(b) Clock output restart from Stop mode

Figure 3.3.8 shows the basic timing for the sequence of the restart from the clock stopped state in the Stop mode. When restarting by setting the $\overline{INT}$ or $\overline{NMI}$ signal to "0", the warm-up time is automatically created by the internal counter. In the restart by the $\overline{RESET}$ signal, oscillation restarts without warm-up.



Figre 3.3.7 (b) Basic Timing for Sequency of Restart from Clock stopped State (Idle 2 Mode)



Figre 3.3.8 Basic Timing for Sequency of Restart from Clock stopped State (Stop Mode)

3.3.4 Relationship with MPU

The following describes the relationship between the CGC and the MPU mainly in terms of the halt clear operation.

[1] RESET Signal

Figure 3.3.9 shows an example of the timing for the restart from the Stop mode on the TMPZ84C015A using $\overline{\text{RESET}}$ signal for both the MPU and CGC . To reset the MPU, the $\overline{\text{RESET}}$ signal must be set to "0" for at least 3 stable clocks. When the $\overline{\text{RESET}}$ signal goes "1", the MPU releases the halt state after a dummy cycle of 2T clock states to start executing instructions from address 0000H.

To restart the clock output by the $\overline{\text{RESET}}$ signal in the Stop mode, the internal counter to determine the warm-up time does not operate.

Therefore, if the MPU does not restart correctly due to the unstable clock output immediately after the restart of the internal oscillator, or the unstability of the crystal at power-on, the $\overline{\text{RESET}}$ signal must be held at "0" for a time long enough for the MPU to be reset securely.

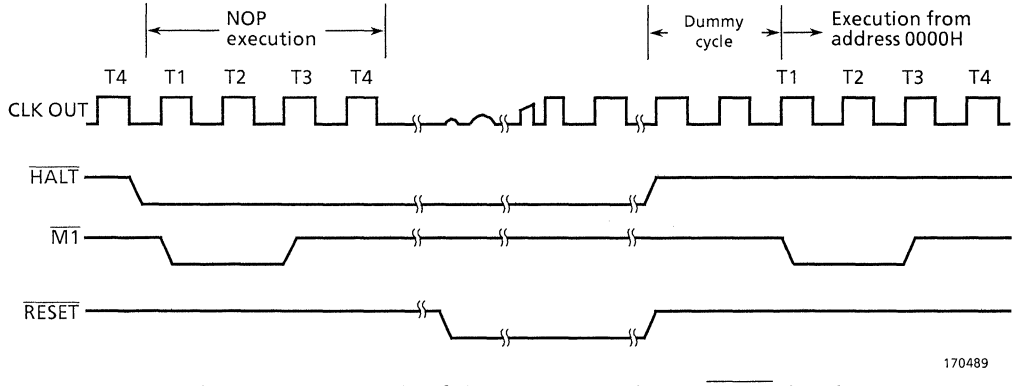


Figure 3.3.9 Example of clock Restart Timing by $\overline{\text{RESET}}$ Signal

[2] Releasing Halt State by Interrupt Signal

The CGC restarts the clock output from the Idle or Stop mode by the input of $\overline{\text{INT}}$ or $\overline{\text{NMI}}$ signal. By this clock, the MPU starts operating. However, when the CGC restarts the clock output, the MPU is still in the halt state executing NOPs. To clear the halt state, the interrupt signal must be entered into the MPU (in the case of the $\overline{\text{INT}}$ signal) for at least one instruction. The MPU interrupt is detected on the rising edge of the last clock of each instruction (NOP for the halt state).

(1) When using non-maskable interrupt (NMI)

MPU's non-maskable interrupt is edge trigger input. The MPU contains the flip-flop to detect an interrupt. The state of this internal NMI flip-flop is sampled on the rising edge of the last clock of each instruction. Therefore, when a short active low ("0") pulse has been inserted before the interrupt detection timing, the interrupt is acknowledged. The $\overline{NMI}$ input of the TMPZ84C015A is connected to the $\overline{NMI}$ input of the MPU via the CGC, performing the same operations as above. (See Figure 3.3.11)

(2) When using maskable interrupt (INT)

With a maskable interrupt, the maskable interrupt enable flip-flop (IFF) must be set to "1" by program before the $\overline{INT}$ input signal is detected "0". Even if the CGC accepts the $\overline{INT}$ signal to restart supply of the clock, no interrupt is acknowledged unless the $\overline{INT}$ signal is kept inserted until one instruction (NOP) has been executed. Figure 3.3.10 shows the timing for clearing the halt state by the interrupt signal.

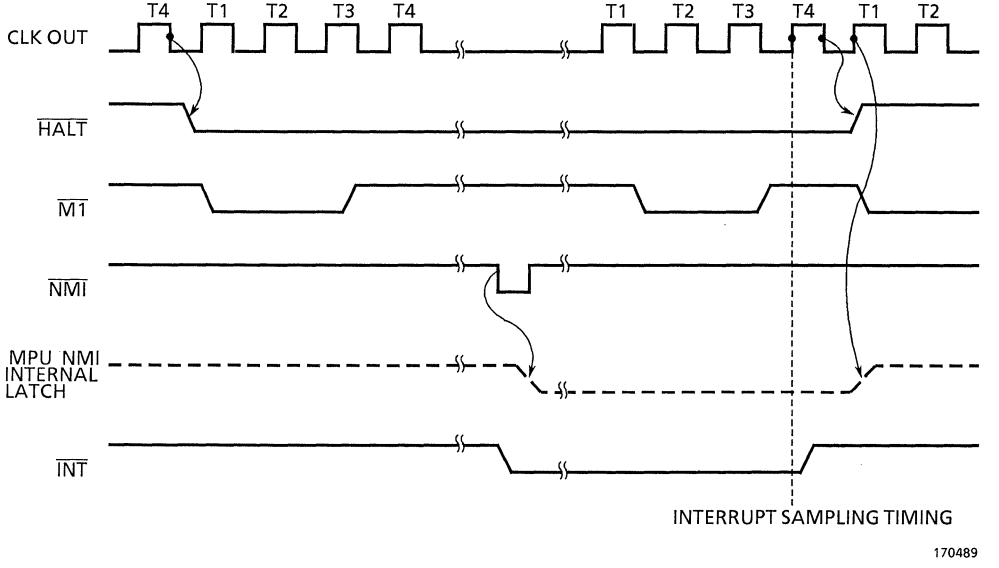


Figure 3.3.10 Timing for Clearing Halt State by Interrupt Signal

[3] Connecting CGC to MPU on TMPZ84C015A

Figure 3.3.11 shows the connection between the CGC and the MPU on the TMPZ84C015A

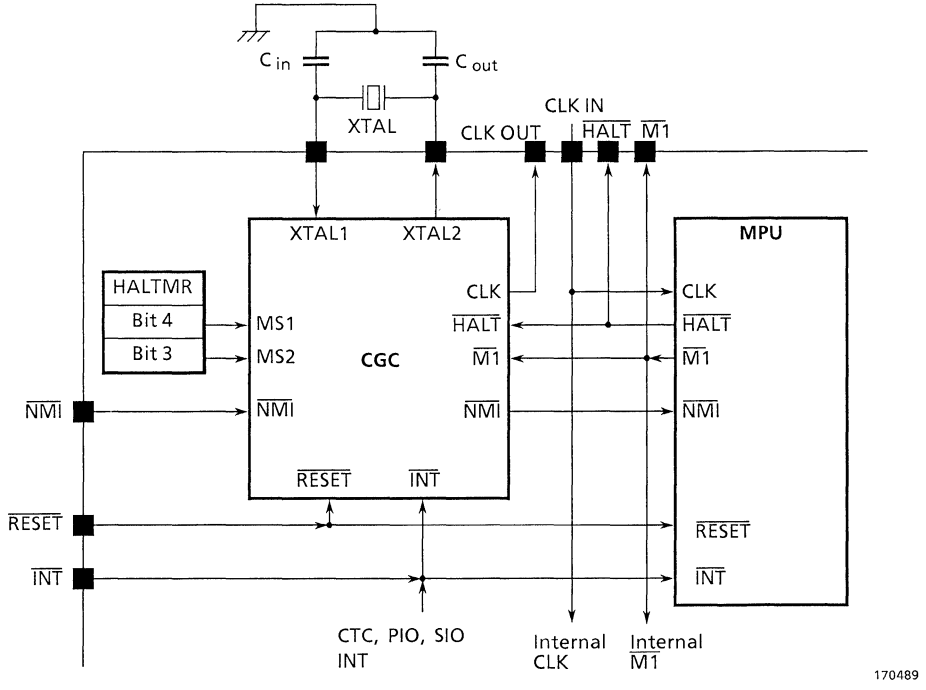


Figure 3.3.11 Connection Between CGC and MPU

3.4 CTC OPERATIONAL DESCRIPTION

The CTC has 4 independent channels. To these channels, addresses are allocated on the TMPZ84C015A's I/O map, permitting the read/write of the channels in the MPU's I/O cycle. (See Figure 3.4.1) This subsection mainly describes the CTC operation to be performed after accessed.

3.4.1 CTC Block Diagram

Figure 3.4.1 shows the block diagram of the CTC.

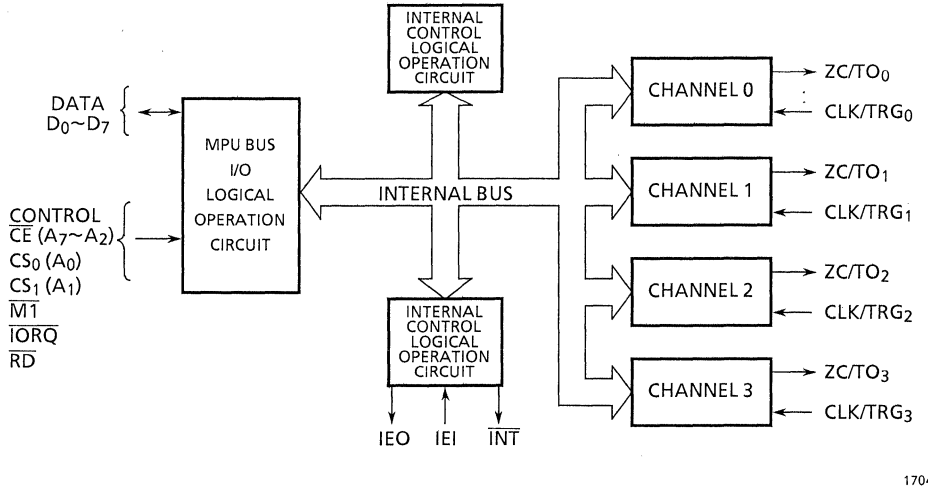


Figure 3.4.1 Block Diagram of CTC

3.4.2 CTC System Configuration

The CTC system consists of the following 4 logic circuits:

(1) MPU bus I/O logic circuit

(2) Internal control logic circuit

(3) Interrupt control logic circuit

(4) Four independent counter/timer channel logic circuits

[1] MPU Bus I/O Logic Circuit

This circuit transfers data between the MPU and the CTC.

[2] Internal Control Logic Circuit

This circuit controls the CTC operational functions such as the CTC selecting chip enable, reset, and read/write circuits.

[3] Interrupt Control Logic Circuit

This circuit performs the MPU interrupt related processing such as priority determination. The order of priority with other LSIs is determined according to the physical location in daisy chain connection.

[4] Counter/Timer Channel Logic Circuit

This circuit consists of the following 2 registers and 2 counters.
Figure 3.4.2 shows the configuration of this circuit.

- Time-constant register (8 bits)
- Channel control register (8 bits)
- Down-counter (8 bits)
- Prescaler (8 bits)

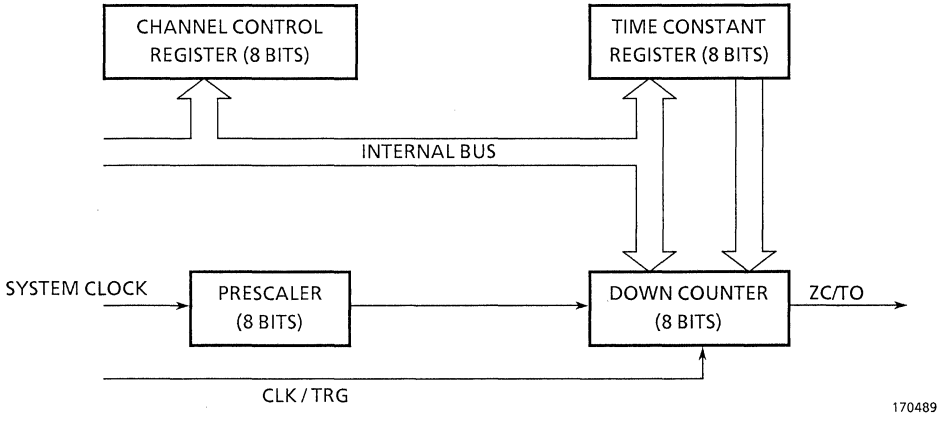


Figure 3.4.2 Configuration of Counter/Timer Channel Logic Circuit

(1) Time-constant register

This register holds the time constant to be written in the down counter. When the CTC is initialized or the down-counter has reached zero, the time constant is loaded into down-counter. The time constant is set immediately after the MPU has written the channel control word in the channel control register. For a time constant, an integer from 1 to 256 can be used.

(2) Channel control register

This register is used to choose the channel mode or condition according to the channel control word sent from the MPU.

(3) Down-counter

The contents of the time-constant register are loaded into the down counter. In the counter mode, these contents are decremented for each edge of the external clock; in the timer mode, they are decremented for each prescaler clock output. The contents of the time-constant register are loaded at initialization or when the down-counter has reached zero.

The contents of the down-counter can be read any time. Also, the system can be programmed so that an interrupt request is generated each time the down-counter has reached zero.

(4) Prescaler

The prescaler, used only in the timer mode, divides the system clock by 16 or 256. The dividing number is programmed by channel control word. The output of the prescaler becomes the clock input to the down-counter.

3.4.3 CTC Basic Operations

[1] Reset

The state of the CTC is unstable after it is powered on. To initialize the CTC, the low level signal needs to be applied to the $\overline{\text{RESET}}$ pin. On any channel, the channel control word and time-constant data must be written to be started before it is started in the counter or timer mode. To program the system to enable interrupts, the interrupt vector word must be written in the interrupt controller. When these data have been written in the CTC, it is ready to start.

[2] Interrupt

The CTC can cause an interrupt when the MPU is operating in the mode 2. The CTC interrupt can be programmed for each channel. Each time the channel's down-counter has reached zero, the CTC outputs the interrupt request signal ($\overline{\text{INT}}$). When the MPU accepts the CTC's interrupt request, the CTC outputs the interrupt vector. Based on this interrupt vector, the MPU specifies the start address of the interrupt processing routine and calls it to start interrupt processing.

The MPU specifies the start address of the interrupt processing routine by the interrupt vector output from the CTC, so that the user can change the vector value to call any desired address.

The interrupt processing is terminated when the MPU executes an RETI instruction. The CTC has the circuit which decodes the RETI instruction. By constantly monitoring the data bus the CTC can detect the termination of the interrupt processing.

The order of interrupt priority with the Z80 peripheral LSIs is determined by the daisy chain connection. That is, the peripheral LSIs are connected one after another and the one physically near the MPU is given a higher priority. The priority of the Z80 peripheral LSIs (CTC, PIO, and SIO) contained in the TMPZ84C015A is determined by the contents of the interrupt priority register (#F4: bits 2 through 0). Inside the CTC, channel 0 is given the highest priority, followed by channels 1, 2 and 3 in this order.

The CTC and other peripheral LSIs on the TMPZ84C015A have the signal lines IEO and IEI. Connect the IEO of a higher peripheral LSI to the IEI of a lower perpheral LSI. Connect the IEI of the highest peripheral LSI to VCC. Leave the IEO of the peripheral LSI unused. In this connection, the CTC interrupt is caused under the following conditions:

- When both IEI and IEO are high, no interrupt is caused. At this time, the $\overline{INT}$ signal is high. An interrupt can be requested in this state.
- When the CTC outputs the interrupt request signal ($\overline{INT}$), the IEO of the CTC becomes low. When the MPU accepts the interrupt, the $\overline{INT}$ goes high again.
- When the IEI goes low, the IEO also goes low.
- While the IEI is low, no interrupt can be requested.
- When the IEI goes low while an interrupt is being serviced, the interrupt processing is aborted.

[3] Operation Modes

The CTC operates in either the counter mode or the timer mode. Mode is selected by writing the channel control word.

(1) Counter mode

In the counter mode, the number of edge of the pulses applied to the channel's CLK/TRG pin is counted. When pulses have been input, the contents of the down-counter are decremented synchronizing with the rising edge of the next system clock. The pulse's rising edge or falling edge to be counted can be specified by the channel control word.

When the contents of the down-counter has reached zero, the high level pulse is output from the ZC/TO pin. When the interrupt is enabled by the channel control word, the $\overline{INT}$ pin goes low and an interrupt is requested. When the contents of the down-counter has reached zero, the time constant data written in the time constant register is automatically loaded into the down-counter. To load a new time constant value into the down-counter, write the data to the time constant register, and it is loaded into the down-counter after the current count operation is terminated.

(2) Timer mode

In the timer mode, the time intervals which are integral multiples of the system clock period. A timer interval is measured according to the system clock. The system clock is supplied to the prescaler which divides it by a factor of 16 or 256. The output of the prescaler provides the clock to decrement the down-counter by 1. The time constant data is automatically loaded into the down-counter each time it has reached zero as in the counter mode. When the contents of the down-counter has reached zero, the high level pulse is output from the ZC/TO pin.

This pulse period is given by the following expression:

tc*P*TC

Where, tc = System clock period
P = Prescaler value (16 or 256)
TC = Time constant data (256 for 00H)

The user can select, by means of the channel control word, to start the timer automatically or to start the timer on the edge of the pulse at CLK/TRG pin. In case the user select the CLK/TRG pin, the user can also select the rising edge or falling edge of the pulse.

3.4.4 CTC Status Transition Diagram and Basic Timing

[1] Transition Diagram

Figure 3.4.3 shows the CTC status transition diagram.

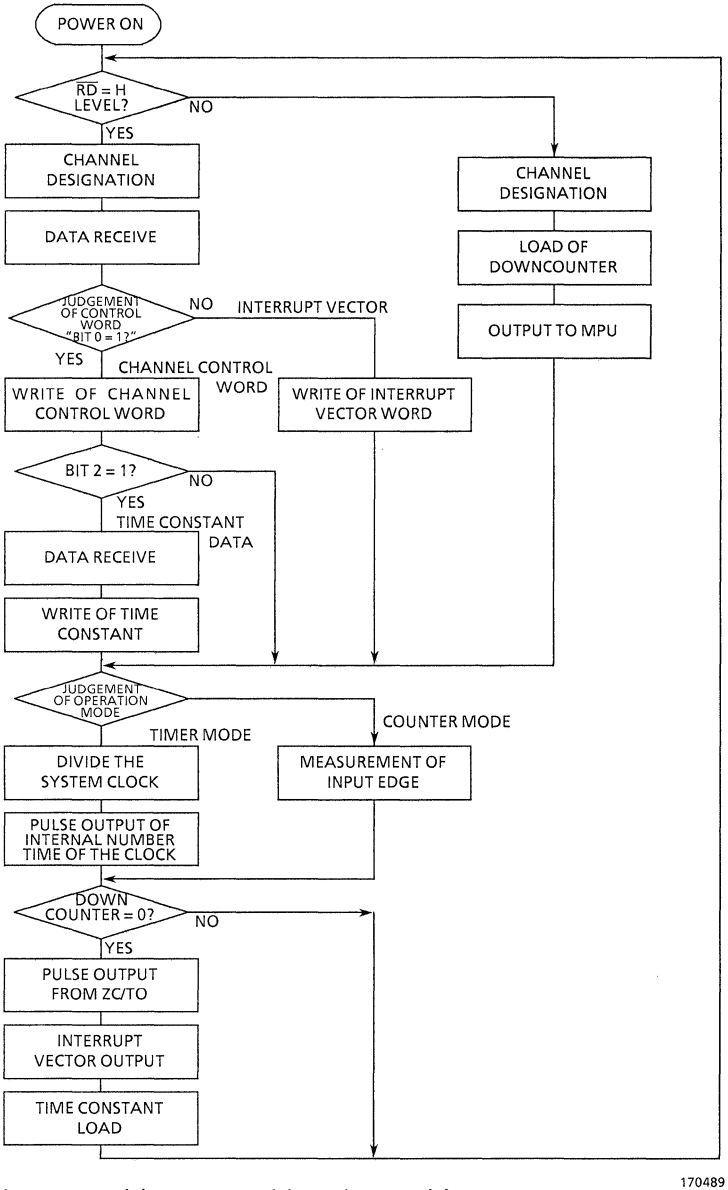


Figure 3.4.3 (a) CTC Transition Diagram (a)

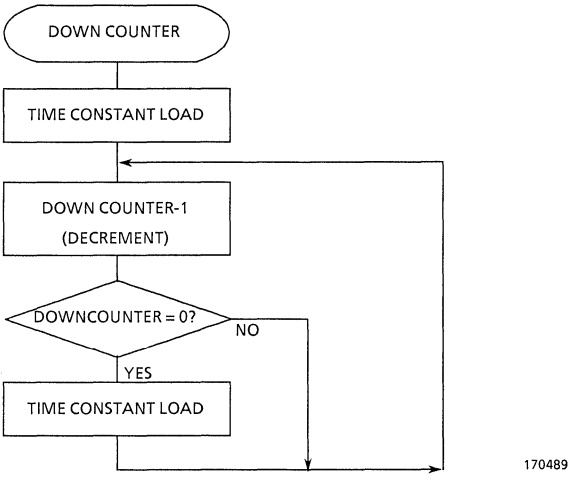


Figure 3.4.3 (b) CTC Transition Diagram (b)

[2] Basic Timing

(1) Write cycle

The write cycle is used to write a channel control word, an interrupt vector, or a time constant. The C MPU drives the $\overline{IORQ}$ pin of the CTC low in the subsequent system clock cycle T2 to start the write cycle.
It is required to make the $\overline{M1}$ pin of the CTC high to indicate that the write cycle is on.

At the start of the cycle, the channel is specified by CS1 (A1) or CS0 (A0) of the CTC. Thus, the CTC's internal registers are ready to accept data in system clock T3. Tw is the state to be automatically inserted by the MPU.

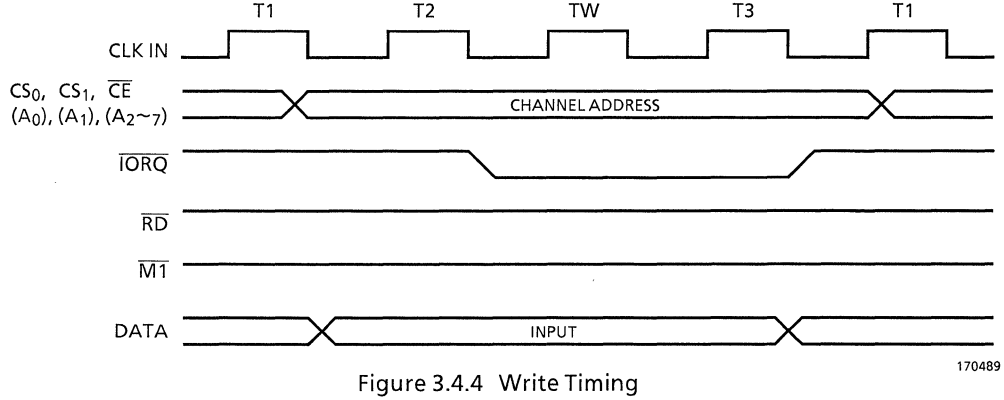


Figure 3.4.4 Write Timing

(2) Read cycle

The read cycle is used to read the contents of the down-counter. During clock cycle T2, the MPU initiates a read cycle by driving the $\overline{RD}$ and $\overline{IORQ}$ pins low. It is required to make the $\overline{M1}$ pin high to indicate that the read cycle is on. At the start of the read cycle, the channel is specified by CS_1 (A_1) or CS_0 (A_0) of the CTC.
At the rising edge of system clock TW, the contents of the down-counter at the time of the rising edge of T2 are put on the data bus. TW is the wait state to be automatically inserted by the MPU.

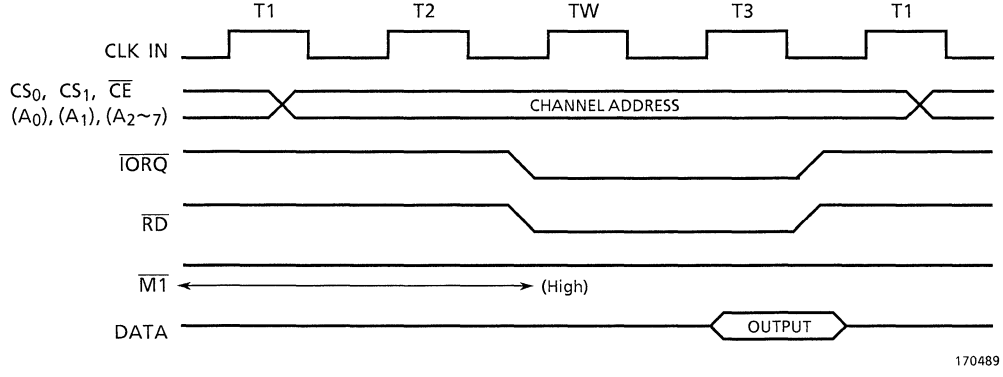


Figure 3.4.5 Read Timing

[3] Counter mode

In the counter mode, the down-counter is decremented synchronizing with the system clock, at the edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the down-counter is decremented one system clock later. When the down-counter has reached zero, a high level pulse is output from the ZC/TO pin.

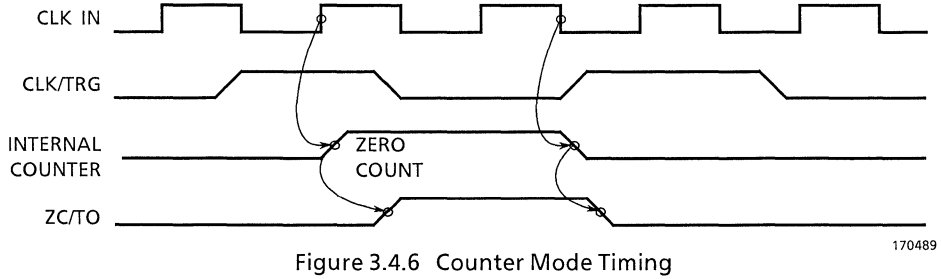


Figure 3.4.6 Counter Mode Timing

[4] Timer mode

The timer starts operating at the second rising edge of the system clock from the rising edge of the pulse applied from the external circuit connected to the CLK/TRG pin. The period of the pulse to be applied to the CLK/TRG pin must be greater than 2 times the system clock period. Also, it is required to insert the setup time between the active edge of the CLK/TRG pin signal and the rising edge of the succeeding system clock. When the interval between these pulses is short, the timer starts one system clock cycle later.

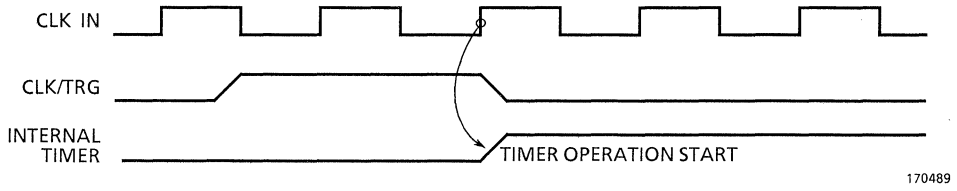


Figure 3.4.7 Timer Mode Timing

[5] Interrupt acknowledge cycle

Having received the interrupt request signal ($\overline{INT}$) from the CTC, the MPU drives the CTC's $\overline{M1}$ pin and $\overline{IORQ}$ pin low to provide the acknowledge signal. The $\overline{IORQ}$ pin goes low 2.5 system clocks later than the $\overline{M1}$ pin. To stabilize the signal lines (IEI and IEO) in daisy chain connection, the interrupt request cannot be changed on each channel while the $\overline{M1}$ pin is low. The $\overline{RD}$ pin is held high to make distinction between the instruction fetch cycle and the interrupt acknowledge cycle. While the $\overline{RD}$ pin is high, the CTC's interrupt control circuit determines the interrupt-requesting channel of highest priority. When the CTC's IEI is high and the $\overline{M1}$ pin and $\overline{IORQ}$ pin go low, the interrupt vector is output from the interrupt requesting channel of highest priority. At this time, 2 system clock cycles are automatically inserted by the MPU as a wait state to maintain the stabilization of the daisy chain connection.

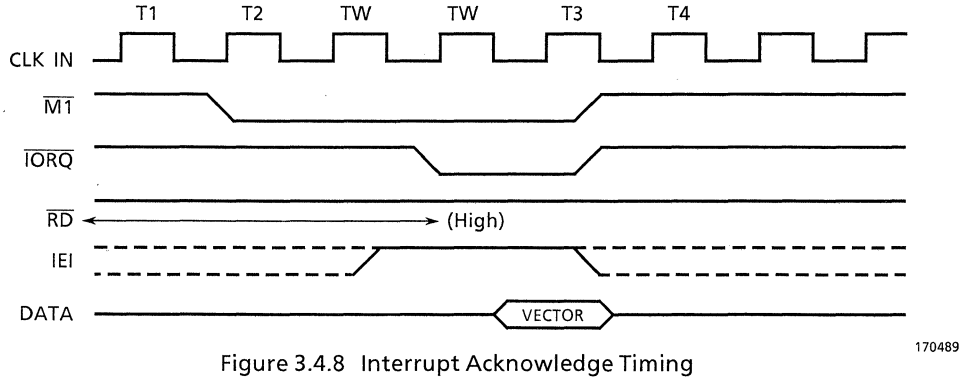


Figure 3.4.8 Interrupt Acknowledge Timing

[6] Return from interrupt processing

Return from the interrupt processing is performed when the MPU executes the RETI instruction. This RETI instruction must be used at the end of the interrupt processing routine. When this instruction is executed by the MPU, the CTC's IEI and IEO return to the state active before the interrupt has been serviced.

The RETI instruction is a 2-byte instruction. Its code is EDH 4DH. The CTC decodes this instruction to check if there is the next interrupt request channel.

In the daisy chain structure, the interrupting LSI's IEI and IEO are held high and low respectively at the time the instruction code EDH has been decoded.

The code following EDH is 4DH, only the peripheral LSI which has sent the last interrupt vector (that is, the LSI whose IEI is high and IEO is low) returns from the interrupt processing. This restarts the processing of the suspended interrupt of the peripheral LSI of the next higher priority.

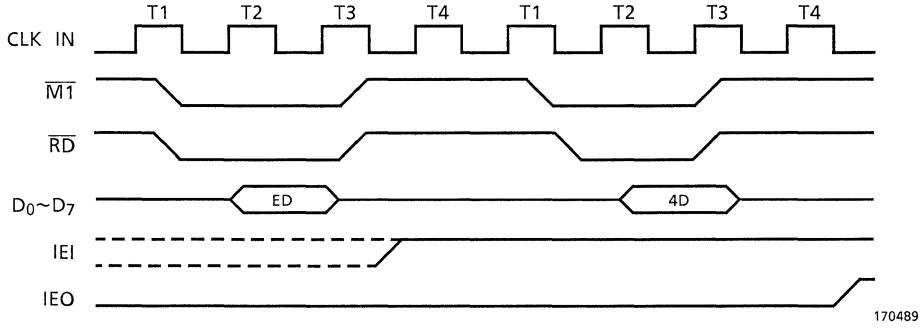


Figure 3.4.9 Interrupt Return Timing

3.4.5 CTC Operational Procedure

To operate the CTC in the counter mode or the timer mode, the channel control word and time-constant data must be written in the CTC. To enable interrupts by the channel control word, the interrupt vector must be written in the CTC.

[1] I/O Address and Channel Control Word

To write the channel control word in the CTC, the channel must be specified by the corresponding channel I/O address.

Table 3.4.1 Channel I/O Addresses

Channel	I/O address
0	#10
1	#11
2	#12
3	#13

170489

The channel control word to be written in the CTC consists of 8 bits. The system data bus D0 through D7 correspond to bit 0 through 7 respectively. Figure 3.4.10 shows the meaning of each bit. Table 3.4.2 shows the function of each bit.

D7	D6	D5	D4	D3	D2	D1	D0
interrupt	Counter/ timer	Prescaler	Edge	Trigger	Time constant	Reset	1

170489

Figure 3.4.10 Channel Control Word

For the channel control word, D0 must always be 1.

Table 3.4.2 Meanings and Function of Channel Control Words (1/3)

Bit	Meaning and function	
	0	1
Bit 7 (D7)	Disables channel interrupt	Enables channel interrupt. In either counter or timer mode, the interrupt is requested every time the down-counter has reached zero. When this bit is set to "1", the interrupt vector must be written in the CTC before the down-counter starts. When the channel control word whose D7 bit is "1" is written in an already operating channel, the interrupt occurs only when the down-counter has reached zero for the first time after the writing of the new channel control word.

170489

(2/3)

Bit	Meaning and function	
	0	1
Bit 6 (D6)	Puts the channel in the timer mode. Puts the system clock into the prescaler and outputs the divided signal to the down-counter.	Puts the channel in the counter mode. The down-counter is decremented for each edge trigger applied to the CLK/TRG pin. In the counter mode, the prescaler is not used.
Bit 5 (D5)	Used only in the timer mode. The prescaler is set to divide the system clock by 16.	Used only in the timer mode. The prescaler is set to divide the system clock by 256.
Bit 4 (D4)	In the timer mode, the timer operation starts on the falling edge of the trigger PULSE (CLK/TRG). In the counter mode, the down-counter is decremented at the falling edge of the external clock pulse (CLK/TRG)	In the timer mode, the timer operation starts on the rising edge of the trigger pulse (CLK/TRG). In the counter mode, the down-counter is decremented at the rising edge of the trigger pulse (CLK /TRG).
Bit 3 (D3)	Used only in the timer mode. The timer oparation starts on the rising edge of the trigger pulse clocks after a time constant is loaded onto the down-counter.	Used only in the timer mode. The timer operation is started at the leading edge of the external trigger pulse that inputs 2 system clocks after a time constant is loaded onto the down counter. when a time lag between the system clock and trigger pulse satisfies a setup time, the prescaler starts to operate from the second leading edge of the trigger pulse. If a time lag between the system clock and trigger pulse dose not satisfy the setup time, the prescaler starts to operate at the leading edge of the trigger pulse after 3 system clocks. If the trigger pulse is input before loading of a time constant, the operation is the same as that when Bit 3 = 0.
Bit 2 (D2)	This bit (0) indicates that there is no time constant written after channel control word. However, when the channel is in the reset state and this bit cannot be changed to "0" in the channel control word which is given first after the channel reset. To change other state without changing a time constant, input a channel control word with this bit changed to 0.	This bit (1) indicates that there is a time constant written immediately after a channel control word. If a time constant is written while the downcounter is operating, a new time constant is set in the time constant register. The counting which is in progress is carried out continuously when the downcounter becomes zero, and a new time constant is loaded onto the downcounter.

170489

(3/3)

Bit	Meaning and function	
	0	1
Bit 1 (D1)	Continues the current channel operation	Stops the down-counter operation. When this bit is set to "1", the channel operation stops but all the channel control register bits remain unchanged. When bit 2 = "1" and bit 1 = "1", the channel operation remains stopped until a new time constant is written. Channel restart is set up after the new time constant is programmed. The channel is restarted acoording to the state of bit 3. When bit 2 = "0" and bit 1 = "1", the channel operation does not start until a new channel control word is written.

170489

[2] Time-Constant Data

In either the time mode or the counter mode, the time-constant data must be loaded into the time constant register. When bit 2 (D2) of the channel control word is "1", the time constant is loaded into the time constant register immediately after the channel control word is written. A time-constant value must be an integer in a range of 1 to 256. When the 8 bits of a time constant are all "0"s, such a time constant is assumed to be 256. Figure 3.4.11 shows the bit configuration of time-constant data.

D7	D6	D5	D4	D3	D2	D1	D0
TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0

170489

Figure 3.4.11 Time-Constant Data

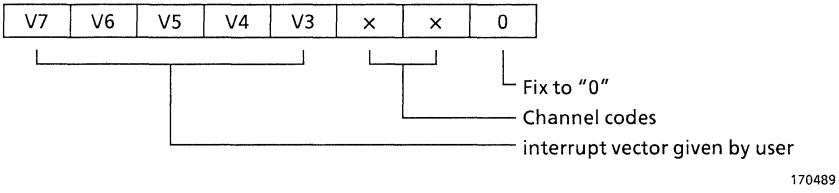
[3] Interrupt Vector

In interrupt in the MPU mode-2, the interrupting channel must give an interrupt vector to the MPU. An interrupt vector is written in the channel-0 interrupt vector register with bit 0 (D0) ="0". The vector is written in the same way as the channel control word is written on channel 0. However, bit 0 (D0) of the vector should always be "0". Bit 7 (D7) through bit 3 (D3) are user-defined values. Bit 2 (D2) and bit 1 (D1) are automatically given and contain the code of the interrupt-requesting channel having the highest priority. Table 3.4.3 shows the channel codes. Figure 3.4.12 shows the interrupt vector bit configuration.

Table 3.4.3 Channel Codes

Bit 2 (D2)	Bit 1 (D1)	Channel number	Interrupt priority
0	0	0	(High)
0	1	1	↕
1	0	2	
1	1	3	(Low)

170489



170489

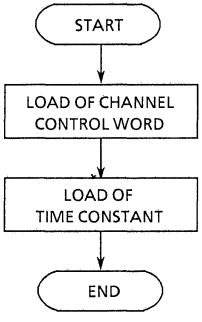
Figure 3.4.12 Interrupt Vector

3.4.6 Using CTC

[1] Counter Mode

The following describes how to use the CTC by referring to a program using channel 0 with interrupt disabled.

(a) The counter programming procedure is shown in Figure 3.4.13



170489

Figure 3.4.13 Counter Programming Procedure

(b) The block diagram for converting the 100 kHz system clock into the 10 kHz equivalent is shown in Figure 3.4.14.

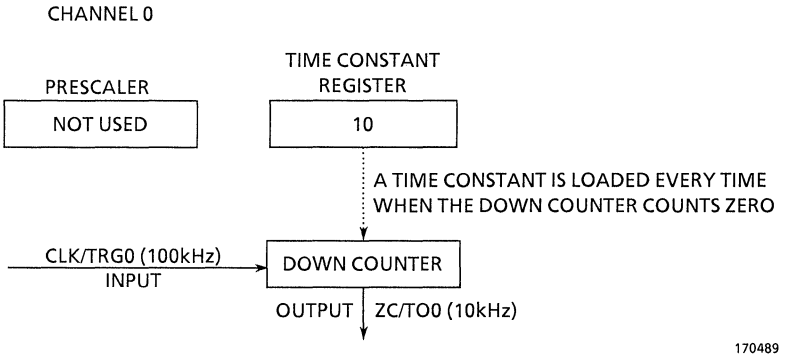


Figure 3.4.14 Down-Counter Block Diagram

(c) The channel control word configuration is shown in Figure 3.4.15

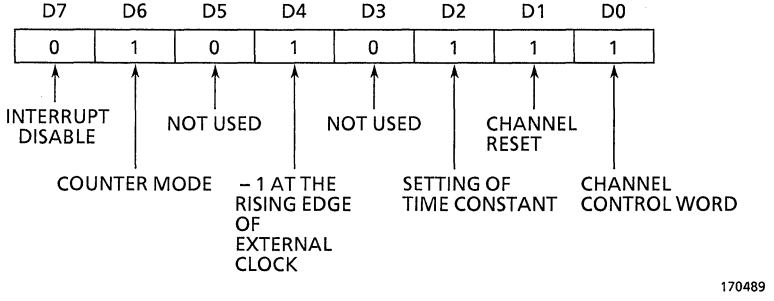


Figure 3.4.15 Channel Control Word Configuration

[2] Timer Mode

(a) The timer programming procedure without using interrupt is shown in Figure 3.4.16.

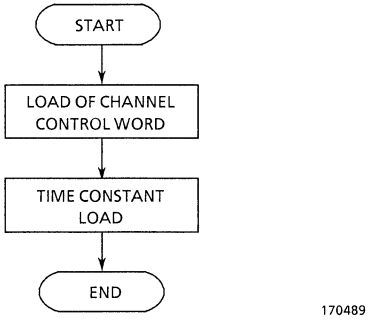


Figure 3.4.16 Timer Programming Procedure

(b) The block diagram for converting the 4 MHz system clock into the 1 kHz equivalent is shown in Figure 3.4.17.

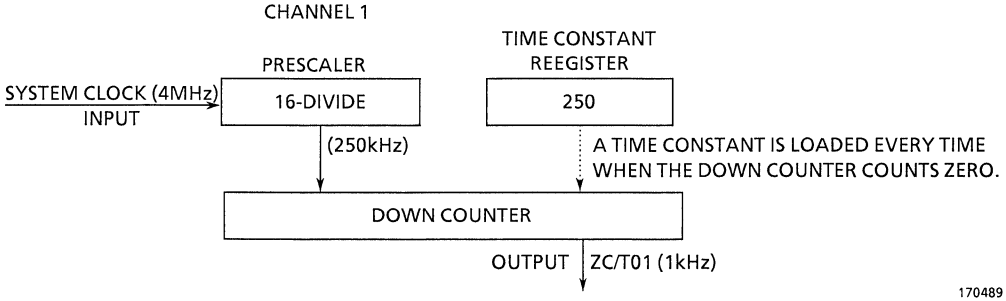


Figure 3.4.17 Timer Block Diagram

(c) The channel control word is shown in Figure 3.4.18.

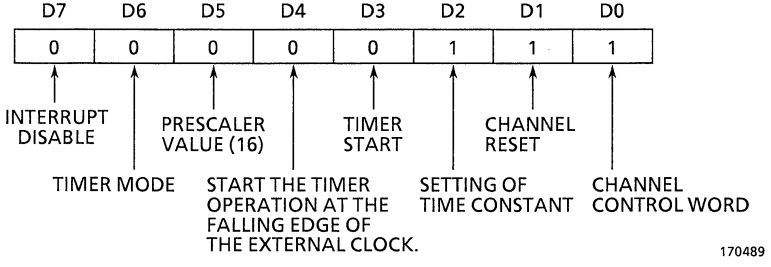


Figure 3.4.18 Channel Control Word

3.5 PIO OPERATIONAL DESCRIPTION

The PIO has two independent, programmable 8-bit ports. These ports are assigned addresses on the TMPZ84C015A's I/O map and therefore can be read/written in the MPU's I/O cycle. This subsection mainly describes the operations that take place after accessing the PIO.

3.5.1 PIO Block Diagram

Figure 3.5.1 shows the PIO block diagram

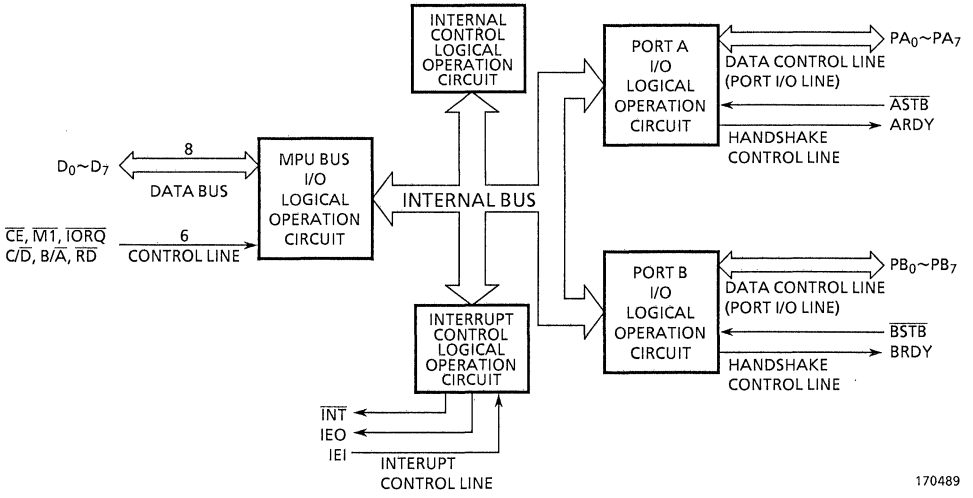


Figure 3.5.1 PIO Block Diagaram

3.5.2 PIO System Configuration

The PIO system consists of the four logic circuits:

(1) MPU bus I/O logic circuit

(2) Internal control logic circuit

(3) Interrupt control logic circuit

(4) Port I/O logic circuit

[1] MPU Bus I/O Logic circuit

The MPU bus I/O logic circuit transfers data between the MPU and the PIO.

[2] Internal Control Logic circuit

The internal control logic circuit controls the PIO operating functions like the PIO selecting chip enable and the read/write circuits.

[3] Interrupt Control Logic circuit

The interrupt control logic circuit performs the MPU interrupt-associated processing such as determining interrupt priorities. The priorities with other LSI peripherals are determined by the physical location in daisy chain connection.

[4] Port I/O Logic Circuit

The port I/O logic circuits are used to directly connect to peripheral devices. Each consists of the following 7 registers and 1 flip-flop. Data are written in the registers by the MPU as specified in the program. Figure 3.5.2 shows the internal configuration of the ports

- Data output register (8 bits)
- Data input register (8 bits)
- Mode control register (2 bits)
- Interrupt vector register (8 bits)
- Interrupt control register (2 bits)
- Mask control register (8 bits)
- Data I/O control register (8 bits)
- Handshake control logic circuit

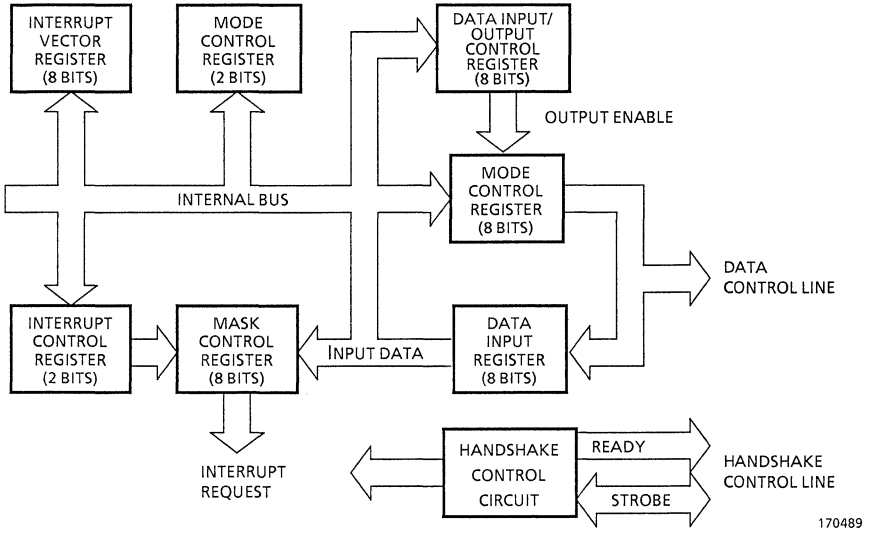


Figure 3.5.2 Port Internal Configuration

(1) Data Output register

This register holds the data to be transferred from the MPU to peripheral devices.

(2) Data input register

This register latches the data to be transferred from peripheral devices to the MPU.
The input data to the MPU is read through this register

(3) Mode control register

This register specifies the operation mode. The operation mode is set by MPU control.

(4) Interrupt vector register

This register holds the vector which makes up the address of the table storing the start address of the interrupt processing routine.

This register is used only for interrupt processing.

(5) Interrupt control register

This register specifies how the I/O ports are to be monitored. This register is used only in the PIO mode 3.

(6) Mask control register

This register specifies which I/O port pin is to be monitored. This register is used only in the PIO mode 3.

(7) Data I/O control register

This register specifies whether each port pin is to be used as output or input. This register is used only in PIO mode 3.

(8) Handshake control logic

This circuit controls the data transfer to the peripheral devices connected to the 8-bit I/O ports.

3.5.3 PIO Basic Operations

[1] Reset

The PIO provides the following two reset capabilities:

(1) Power-on reset

The PIO contains the circuit which automatically resets the PIO at the time of power-on.

(2) Hardware reset

Making the $\overline{\text{RESET}}$ pin low for 2 system clock periods with the $\overline{\text{RD}}$ and $\overline{\text{IORQ}}$ pins being high resets the PIO on the rising edge of the $\overline{\text{RESET}}$ pin. This hardware reset inside the TMPZ84C015A by external pin is possible because the output of the AND circuit between the $\overline{\text{RESET}}$ and $\overline{\text{M1}}$ pins is put on the $\overline{\text{M1}}$ signal of the PIO.

Reset state

(a) The operation mode is set to mode 1 for both ports.

(b) Interrupt is disabled

(c) All the bits of the data I/O register of each port are reset.

(d) All the bits of the mask control register of each port are set and masked.

(e) The port I/O lines of each port are put in the high-impedance state (floating) .

(f) The RDY pin of each port goes low.

The reset state is held until the control word is written. For the function of the control word, see Subsection 3.5.5 "Operational Procedure".

[2] Interrupt

The PIO can cause an interrupt when the MPU is operating in mode 2. The interrupt request signal ($\overline{\text{INT}}$) from the PIO is accepted when the MPU is in the interrupt enabled state (caused after the execution of E1 instruction). Receiving the $\overline{\text{INT}}$ signal, the MPU latches the interrupt vector (8-bit data) sent from the PIO, specifies the start address of the interrupt processing routine based on the vector, and calls the routine to start the processing.

Thus, since the start address of the interrupt processing routine can be specified by the interrupt vector sent from the PIO, the user can change the vector value to call any desired address.

Interrupt processing is terminated when the MPU executes the RETI instruction. The PIO has the circuit to decode the RETI instruction to detect the termination of interrupt processing by constantly monitoring the data bus.

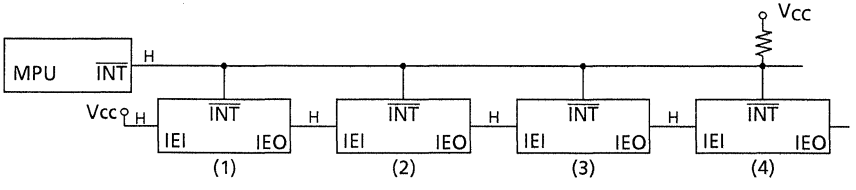
The interrupt priority among the Z80 peripheral LSIs is determined by the daisy chain structure. In daisy chain, the peripheral LSIs are connected one after another as shown in Figure 3.5.3. The more a peripheral LSI is physically located near the MPU, the higher the priority of the peripheral is. Actually, the priority of the Z80 peripheral LSIs (CTC, PIO, and SIO) on the TMPZ84C015A is specified by the contents of the interrupt priority register (# F4 bits 2 through 0). Within the PIO, port A is given higher priority than port B.

The TMPZ84C015A's PIO and peripheral LSIs have the signal lines IEO and IEI connected to the IEO of a higher peripheral LSIs and the IEI of a lower peripheral LSI respectively. However, the IEI of the highest peripheral LSI is connected to the IEI pin and the IEO of the lowest peripheral LSI is connected to the IEO pin. In this state, the PIO interrupt follows the conditions:

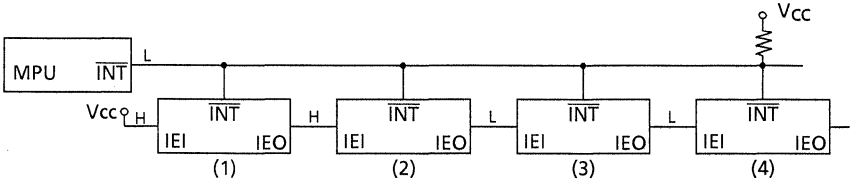
- When both IEI and IEO are high, no interrupt has occurred. This time, the interrupt request signal ($\overline{INT}$) is high. In this state, the PIO can request interrupt.
- When the PIO sends the $\overline{INT}$ signal, it sets the IEO line to the low level. When the interrupt request is accepted by the MPU, $\overline{INT}$ goes back to the low level.
- When the IEI goes low, the IEO also goes low.
- When the IEI is low, the PIO cannot request an interrupt.
- If the IEI goes low during interrupt occurrence, the interrupt processing is suspended.

The operations of the 4 Z80 peripheral LSIs (the states of IEI, IEO and $\overline{INT}$ signal) daisy-chained as shown in Figure 3.5.3 are as follows:

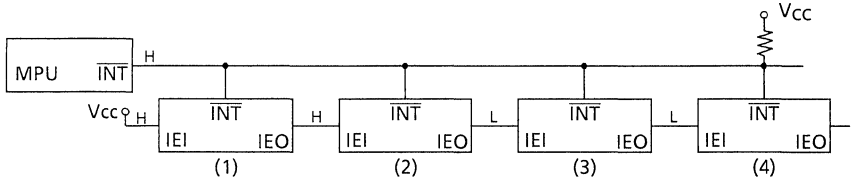
(1) Before interrupt occurrence



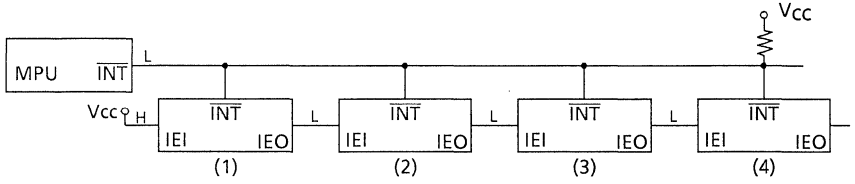
(2) Interrupt request from LSI-2 to the MPU



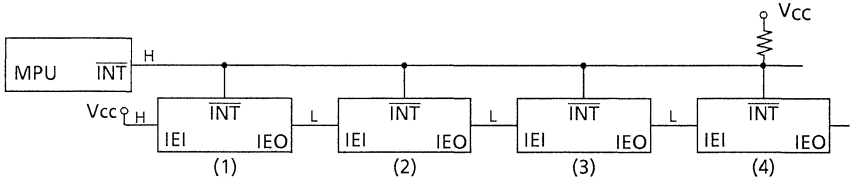
(3) The MPU acknowledges (enables) the interrupt. Interrupt processing for LSI-2 is performed.



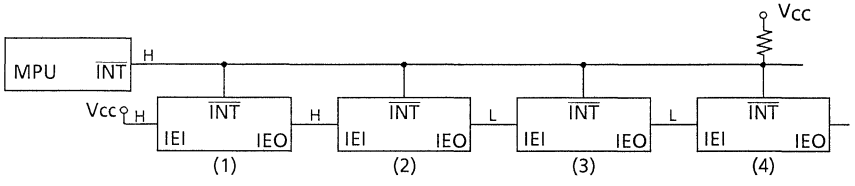
(4) Interrupt request from LSI-1 to the MPU. The interrupt processing for LSI-2 is suspended.



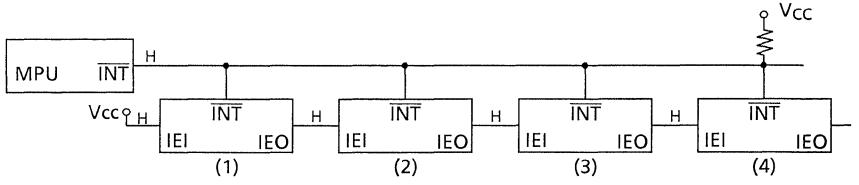
(5) The MPU acknowledges (enables) the interrupt. Interrupt processing for LSI-1 is performed.



(6) Interrupt processing for LSI-1 terminates (upon execution of the RETI instruction). Interrupt processing for LSI-2 is restarted.



(7) Interrupt processing for LSI-2 terminates (upon execution of the RETI instruction).



Interrupt priority is given to LSI-1, LSI-2, LSI-3 and LSI-4 in this order.

170489

Figure 3.5.3 Signal States in Daisy Chain Structure

[3] Operation Modes

The PIO operates in one of the 4 operation modes. The mode is selected by writing the mode control word.

- Mode 0 (byte output mode)
- Mode 1 (byte input mode)
- Mode 2 (byte I/O mode)
- Mode 3 (bit mode)

(1) Mode 0 (byte output mode)

In mode 0, the PIO sends the data received from the MPU to the external device through the port data output register. The contents of this register can be rewritten by using an output instruction. If the data on the bus change, the register contents remain unchanged until the next output instruction is executed. When the MPU executes an output instruction, the write signal is generated in the PIO in the write cycle. Using the signal, the data on the data bus can be latched in the data output register.

(2) Mode 1 (byte input mode)

In this mode, the PIO sends the data received from the external device to the MPU through the port data input register. The data transfer to the MPU is suspended until the MPU has read the current data.

(3) Mode 2 (byte I/O mode)

Mode 2 is a combination of mode 0 and mode 1. This mode is used only for port A. In this mode, all 4 handshake control lines are used. Port A's handshake control lines are used for data output and the port B's handshake control lines are used for data input. For data transfer, port A is used. Port B is set in mode 3 (bit mode) in which no handshake control line is used.

In this mode, the interrupt timing occurs almost at the same time in mode 0 and mode 1. In an input operation, the port B's handshake control lines are used, so that the interrupt vector written in port B is transferred. Therefore, the interrupts in input and output can be controlled by different vectors.

(4) Mode 3 (bit mode)

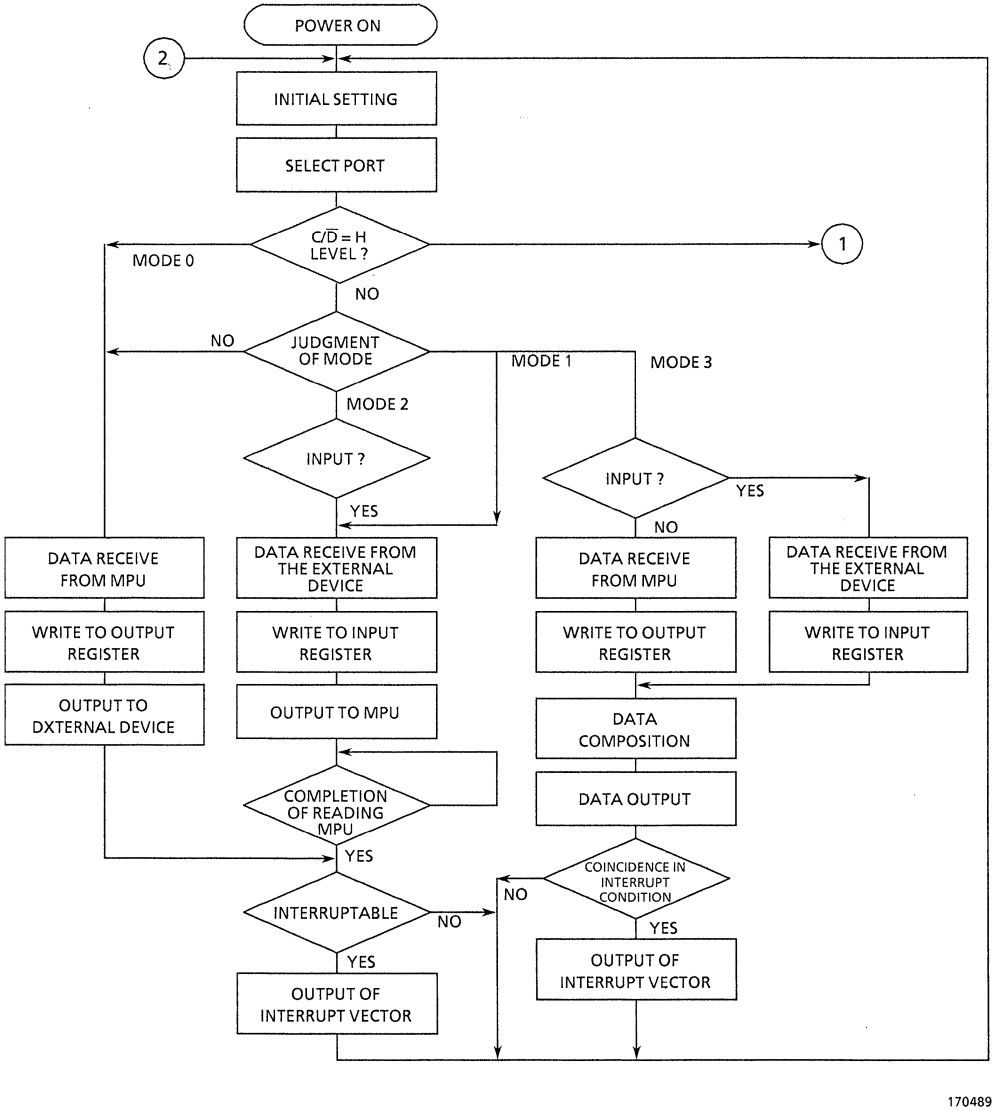
In mode 3, the 8-bit port I/O lines are controlled for each bit. Since no handshake control lines are used, ordinary read/write operations can be performed. I/O operations can be performed on the port as well. In a write operation, the data sent from the MPU to the PIO are latched in the data output register corresponding to the bit set for output in the same timing as in mode 0.

An interrupt occurs in the interrupt enabled state and when the bit set for input satisfies the condition specified in the interrupt control word. However, if port A is operating in mode 2, port B cannot cause an interrupt in the bit mode. Note that, to use the interrupt capability, the mask control register bit corresponding to the bit set for output must be set to "1" to disable its interrupt.

3.5.4 PIO Transition and Basic Timing

[1] Statis Transition

Figure 3.5.4 shows the pio status transition diagram.



170489

Figure 3.5.4 (a) PIO Status Transition

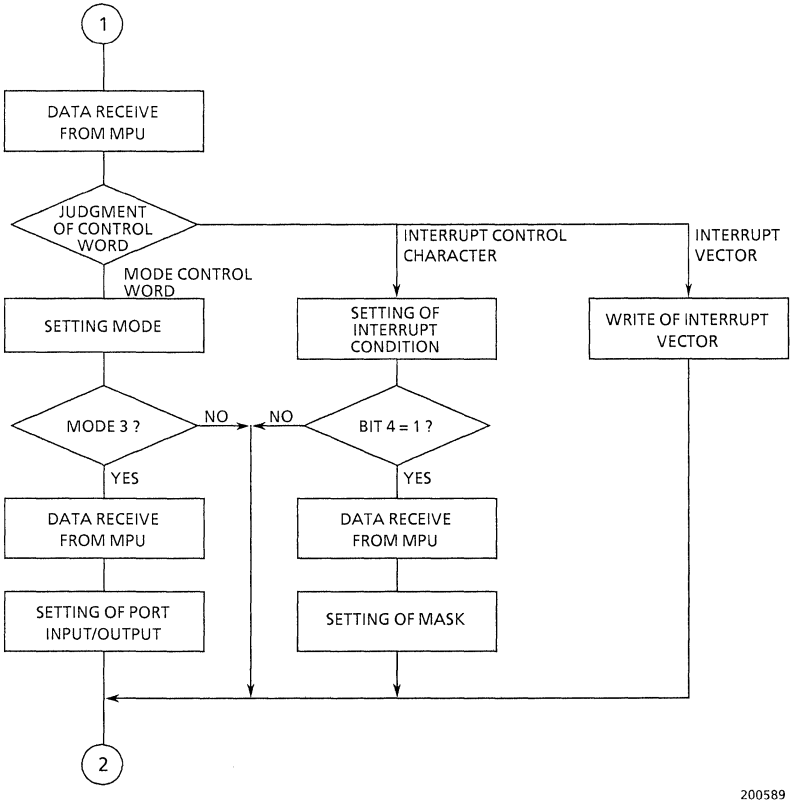


Figure 3.5.4 (b) PIO Status Transition

[2] Write Cycle

The $\overline{IORQ}$, $\overline{RD}$, C/$\overline{D}$ (A0), B/$\overline{A}$ (A1), and $\overline{CE}$ (A7 through A2) signals generate the write signal (*$\overline{WR}$) inside the PIO.

The MPU sets the PIO's $\overline{IORQ}$ signal to the low level at system clock T2, to start the write cycle. At this time, to indicate that this cycle is a write cycle, the PIO's $\overline{M1}$ signal must be set to the high level. At the same time, the MPU sends signals to the PIO's B/$\overline{A}$ (A1) and C/$\overline{D}$ (A0) to specify the port or select control signal or the data. This allows the port data output register of the PIO's selected port to latch the data at system clock T3. TW is a wait state automatically inserted by the MPU.

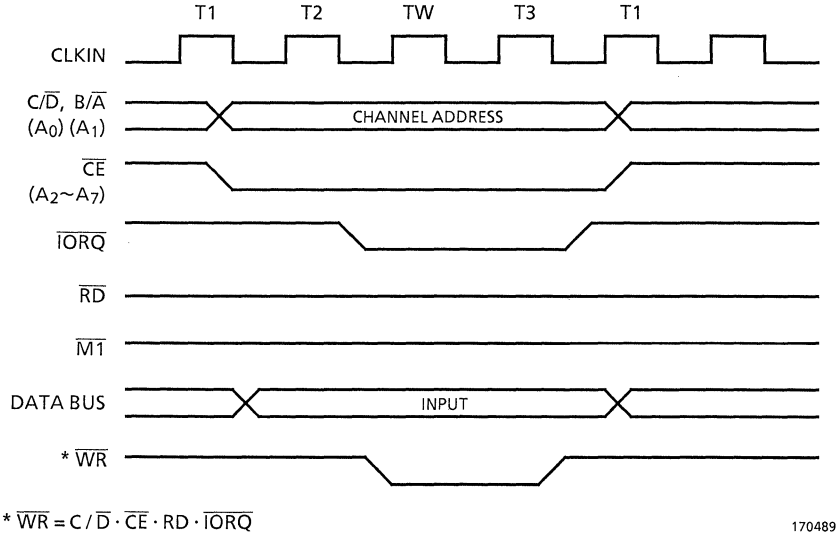


Figure 3.5.5 Write Cycle Timing

[3] Read Cycle

The MPU sets the PIO's, $\overline{RD}$ pin, $\overline{CE}$ signal, and $\overline{IORQ}$ pin to the low level at system clock T2 to start the read cycle. At this time, to indicate that this cycle is a read cycle, the PIO's M1 pin must be set to the high level. The PIO outputs data in the $\overline{CE}$, $\overline{IORQ}$, and RD signals. TW is a wait state automatically inserted by the MPU.

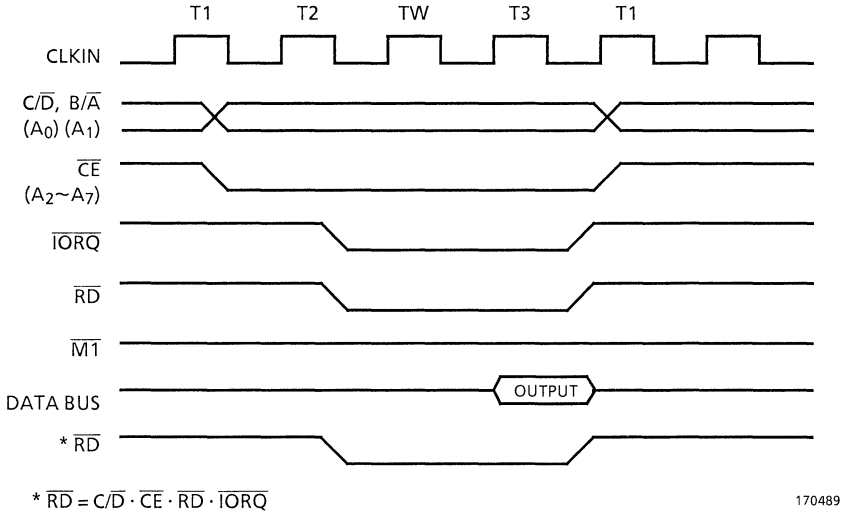


Figure 3.5.6 Read Cycle Timing

[4] Mode 0 (Byte Output Mode)

The mode 0 output cycle starts when the MPU executes an output instruction. When an output instruction is executed, the write signal (*$\overline{WR}$) is generated in the PIO in the write cycle. This signal latches the data on the data bus to the data output register of the selected port. The RDY pin goes high on the first falling edge of the system clock after the rise of the write signal (*$\overline{WR}$). This indicates that the data in the data output register are already on the port I/O pin. The peripheral device sets the RDY pin to the low level on the first falling edge of the system clock after the rise of the $\overline{STB}$ pin to be input to the PIO to indicate that the peripheral device has received the data from the port I/O pin, waiting for the next output instruction. If, at this time, the PIO is enabled for interrupts, it sets the $\overline{INT}$ pin to the low level on the rising edge of the STB signal to output the interrupt request signal to the MPU. Figure 3.5.7 shows the timing chart of mode 0.

[5] Mode 1 (Byte Input Mode)

The input cycle starts when the MPU has completed the previous data read operation. The peripheral device sets the PIO's $\overline{STB}$ pin to the lower level, putting data on the port I/O line. The RDY pin is driven low on the first falling edge of the system clock after the rise of the $\overline{STB}$ pin, disabling the peripheral device to send the next data. If at this time, the PIO is enabled for interrupts,it sets the $\overline{INT}$ pin to the low level on the rising edge of the $\overline{STB}$ pin, making an interrupt request to the MPU. When the MPU executes the input instruction in the interrupt processing routine, the read signal (*$\overline{RD}$) is generated

in the PIO in the read cycle. This signal puts the data in the data input register of the selected port on the data bus. The MPU receives this data. The PIO sets the RDY pin to the high level on the first falling edge of the system clock after the rise of the read signal (*$\overline{RD}$) to wait for the input of the next data. Figure 3.5.8 shows the mode 1 timing chart.

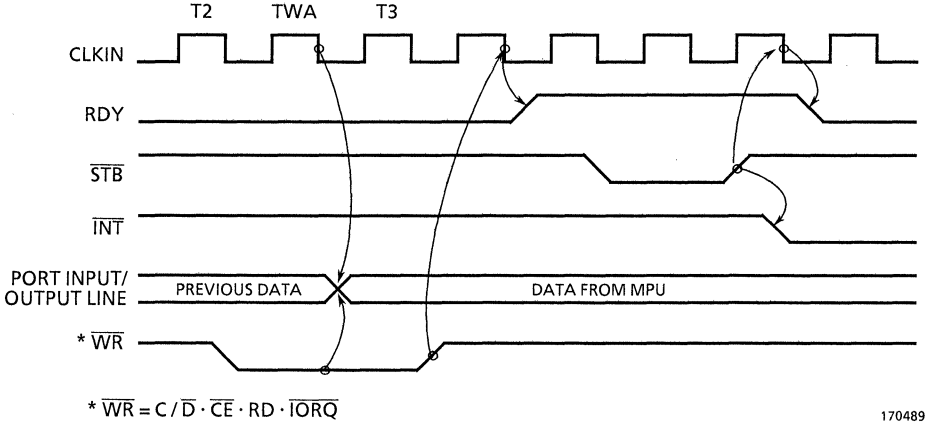


Figure 3.5.7 Mode 0 Timing Chart

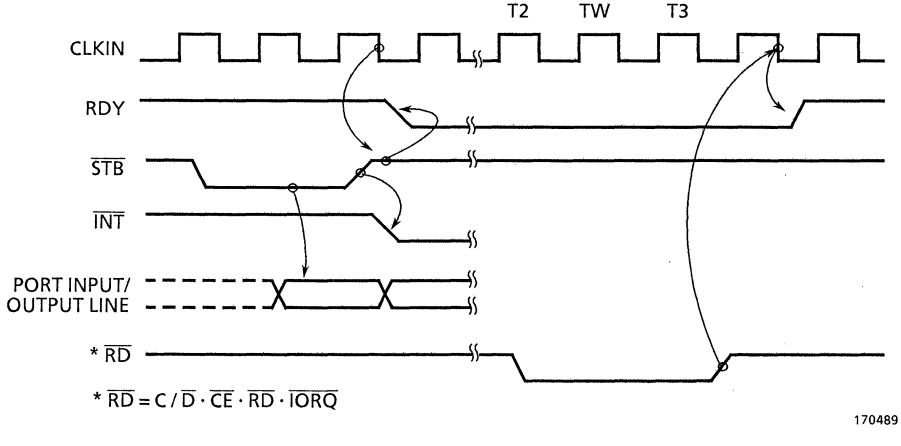


Figure 3.5.8 Mode 1 Timing Chart

[6] Mode 2 (Byte I/O Mode)

Mode 2 is a combination of mode O and mode 1. The timing for output operation is generally the same as in mode 0 except that, in mode 2, data is output only when the $\overline{ASTB}$ pin is low while, in mode 0, data is always on the port I/O line. The peripheral device can receive data on the rising edge of the $\overline{ASTB}$ signal being used as the latch signal.

The input timing is the same as in mode 1.

The port A handshake line is used as output control and the port B handshake line is as input control.

The value of the interrupt vector generated by the $\overline{BSTB}$ signal during a port A input operation is the same as the value of the interrupt vector generated when port B is used in mode 3. Hence, all port B bits are masked by setting the mask control word to disable port B for the interrupt capability.

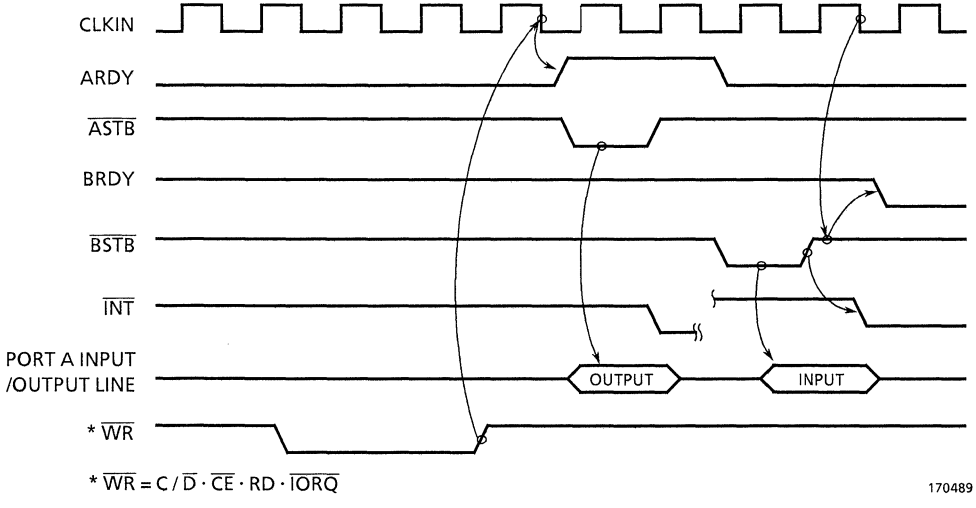


Figure 3.5.9 Mode 2 Timing Chart

[7] Mode 3 (Bit Mode)

In this mode, no handshake line is used. Therefore, the ordinary port read/write operations can be performed, permitting access to the ports any time. The write data from the MPU is latched to the data output register corresponding to the bit set for output in the same timing as in mode 0. Except when port B is used in mode 2, the $\overline{STB}$ pin of the port operating in mode 3 is fixed to the low level. The transfer data consists of the data in the data output register and in the data input register. That is, the data of the bit set for output and the data of the bit set for input make up the transfer data.

An interrupt occurs when the interrupt enabled state is on and the bits set for input satisfy the condition specified by the mask control word, etc. However, if port A is operating in mode 2, port B is disabled for interrupt in the bit mode. Note that, to use the interrupt capability, the bit of the mask register corresponding to the bit set for output must be set to "1" to disable it for interrupts.

An interrupt request occurs when the logic condition becomes true. If the logic condition becomes true immediately before the $\overline{M1}$ pin becomes low or while $\overline{M1}$ pin is low, an interrupt request occurs on the rising edge of the $\overline{M1}$ pin.

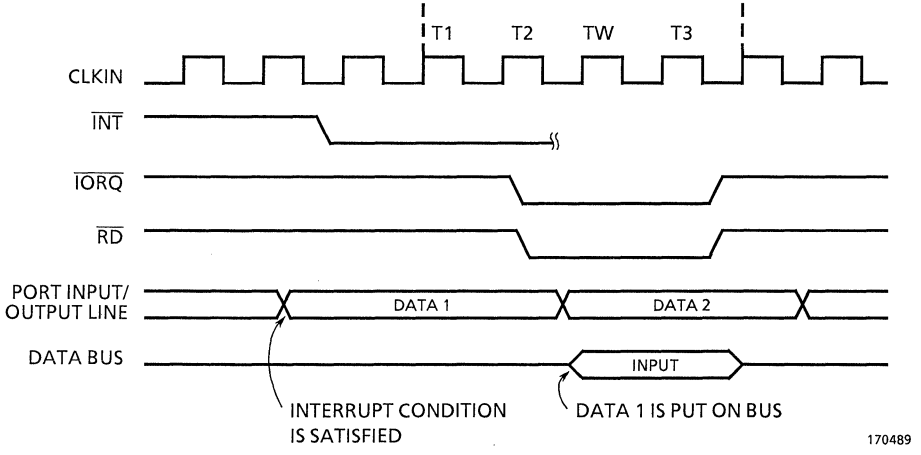


Figure 3.5.10 Mode 3 Timng Chart

[8] Interrupt Acknowledge Cycle

Outputting the interrupt request signal ($\overline{INT}$) , the PIO sets the IEO signal to the low level, disabling the low-priority peripheral LSIs for interrupt requests.

Receiving the interrupt request signal ($\overline{INT}$) from the PIO, the MPU sets the PIO's $\overline{M1}$ and $\overline{IORQ}$ pins to the low level to indicate that the MPU has acknowledged the interrupt request. The $\overline{IORQ}$ pin goes low 2.5 system clocks later than the $\overline{M1}$ pin. To stabilize the daisy-chained signal lines (IEI and IEO), the ports and peripheral LSIs cannot change the interrupt request.

The $\overline{RD}$ pin remains high to make distinction between the instruction fetch and interrupt acknowledge cycles. While the $\overline{RD}$ pin is high, the interrupt control logic in the PIO determines the interrupt requesting port of the highest priority. When the $\overline{IORQ}$ pin goes low with the IEI pin being high, the interrupt vector is put on the data bus from the interrupt requesting port. At the same time, two system clocks are automatically inserted by the MPU as a wait state to stabilize the daisy chain structure.

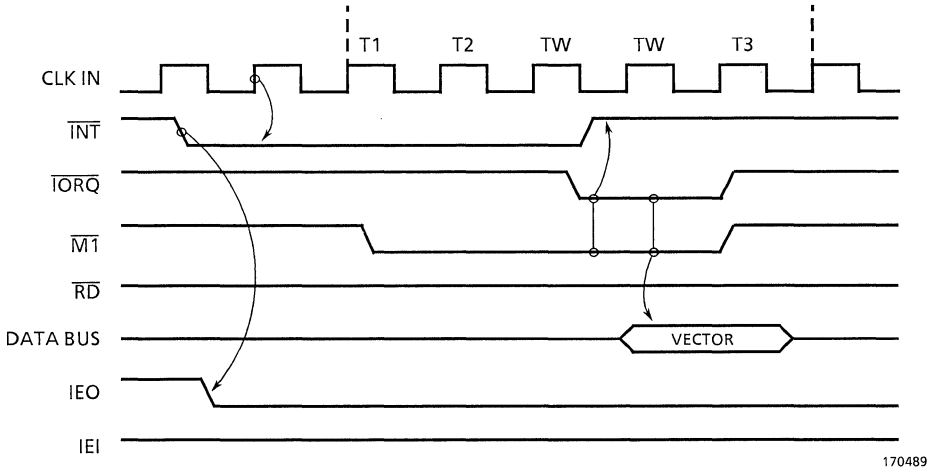


Figure 3.5.11 Interrupt Acknowledge Cycle Timing Charts

[9] Return from Interrupt Cycle

Return from interrupt processing is performed when the MPU executes the RETI instruction. This RETI instruction must be used at the end of the interrupt processing routine. When the MPU executes this instruction, the PIO's IEI and IEO return to the states active before interrupt processing.

The RETI instruction consists of two bytes and its code are EDH and 4DH. The PIO decodes the RETI instruction to determine whether there is any interrupt requesting port. In the daisy chain structure, the IEI and IEO of the interrupting LSI remain high and low respectively at the time the instruction code EDH has been decoded. If the code following EDH is 4DH, only the peripheral LSI which has sent an interrupt vector immediately before, that is, the LSI whose IEI is high and IEO is low, returns from interrupt processing. This restarts the interrupt processing of the suspended peripheral LSI of lower interrupt priority.

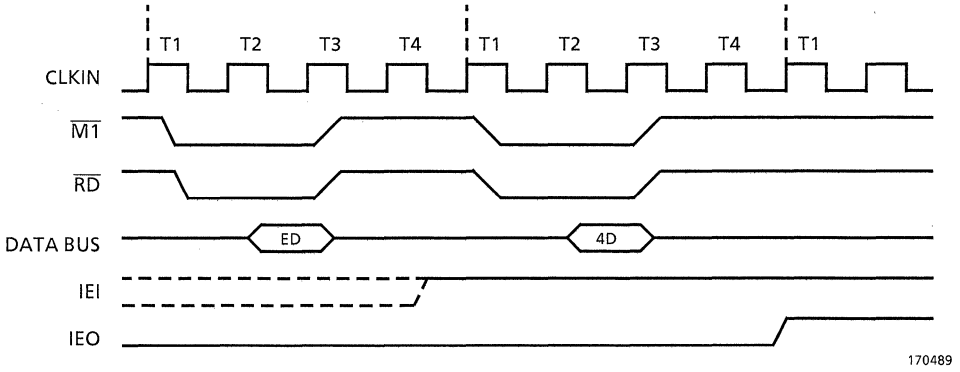


Figure 3.5.12 Interrupt Cycle Return Timing Chart

3.5.5 PIO Operational Procedure

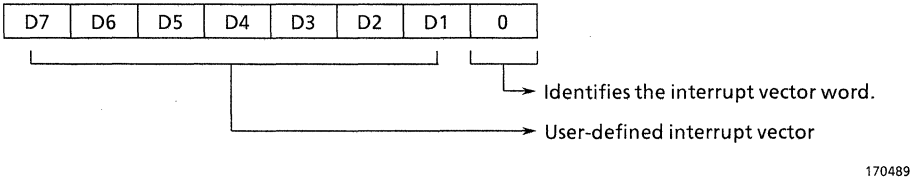
To operate the PIO the control words shown below must be written in it as the initial settings. They must be written in the PIO's ports, A and B, separately. Specify the I/O address listed in Table 3.5.1 to write control words and data in the PIO.

Table 3.5.1 I/OAddresses

I/O function	I/O address
Port A data	#1C
Port A command	#1D
Port B data	#1E
Port B command	#1F

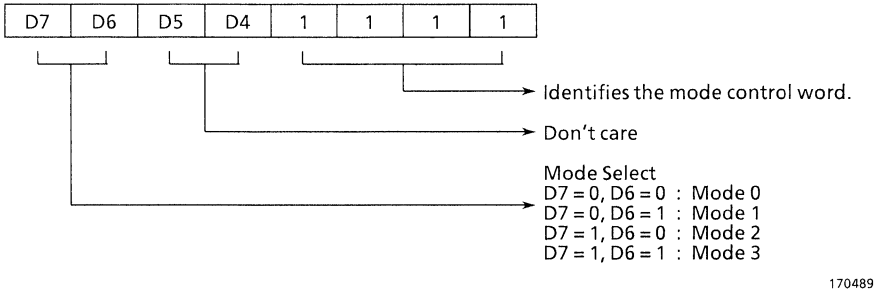
170489

(1) Interrupt vector word



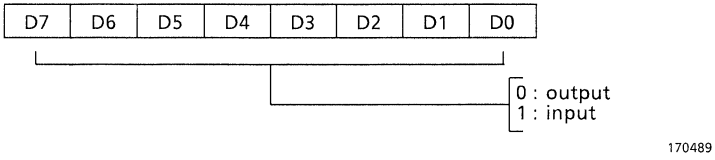
- Using this vector and the contents of the address indicated by the MPU's I register, the MPU generates the start address of the interrupt processing routine.
- D0 through D7 are written in the interrupt vector register.
- This word is not needed when the interrupt capability is not used.

(2) Mode control word



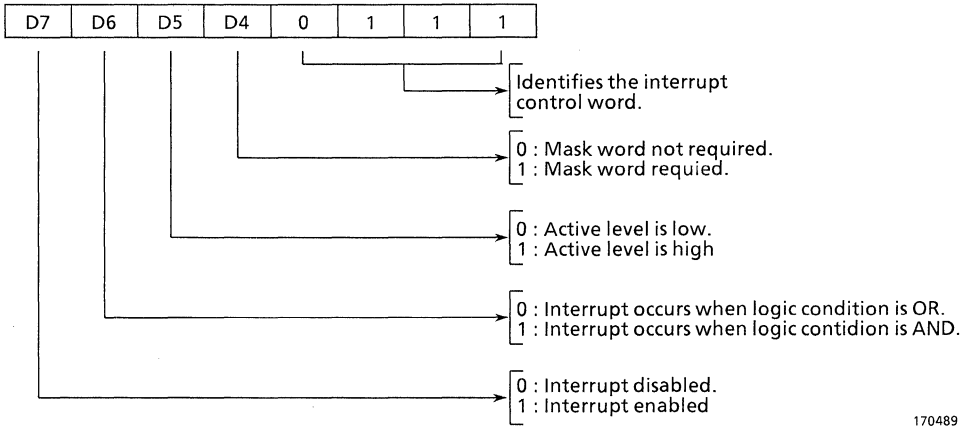
- This word specifies an operation mode.
- D7 and D6 are written in the mode control register.

(3) Data I/O control word



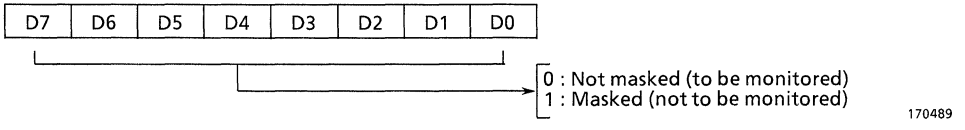
- This word is needed only in mode 3.
- When mode 3 is specified by the mode control word, the data I/O control word is written after it.
- Each port is specified for output or input.
- D0 through D7 are written in the data I/O register.

(4) Interrupt control word



- This word is for interrupt control such as interrupt condition setting.
- D4, D5, and D6 are used only in mode 3.
- With D6=0, interrupt occurs when one of the bits not masked (the bit to be monitored) by the mask control word goes active.
- With D6=1, interrupt occurs when all bits not masked (the bits to be monitored) by the mask control word go active.
- With D4=1, the suspended interrupts are all reset regardless of the mode.
- D5 and D6 are written in the control register.

(5) Mask control word

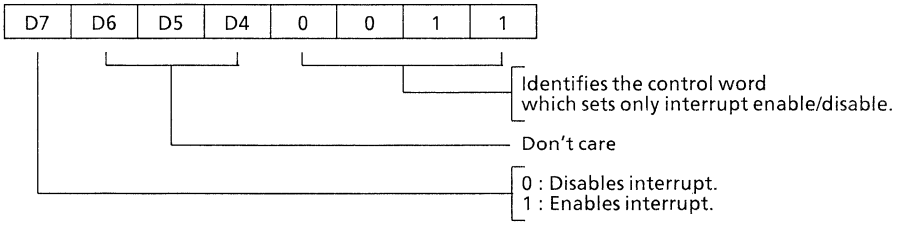


- This word is needed only in mode 3.
- When D4=1 is set by the interrupt control word, the mask control word must be written after it.
- This word specifies whether to monitor the port I/O line specified for input by the data I/O control word.
- When the bit is set to 0, the corresponding input line is monitored and regarded as the input associated with interrupt occurrence.

- When the bit is set to 1, the corresponding input line is masked to provide the input not related to interrupt occurrence.
- The PIO checks only the input line with the bit being 0 to see if the interrupt condition is satisfied. If the condition is satisfied, the PIO requests an interrupt.
- D0 through D7 are written in the mask control register.

When port A is put in mode 2, all 4 handshake lines are used, so that port B must be set in mode 3 which uses no handshake lines. At the same time, all mask control word bits must be set to 1 (masked).

Note : Only interrupt enable/disable can be set by the following control word:



3.5.6 Using PIO

The following is a programming example to operate the PIO's port in mode 3. This program is followed by the main routine and the interrupt processing program.

- The MPU is used in the mode 2 interrupt.
- The table storing the start address of the interrupt processing routine is 0802H.
- Interrupts occur when both PIO's port input lines A6 and A5 go high.
- The I/O addresses of the PIO are the addresses listed in Table 3.5.1.

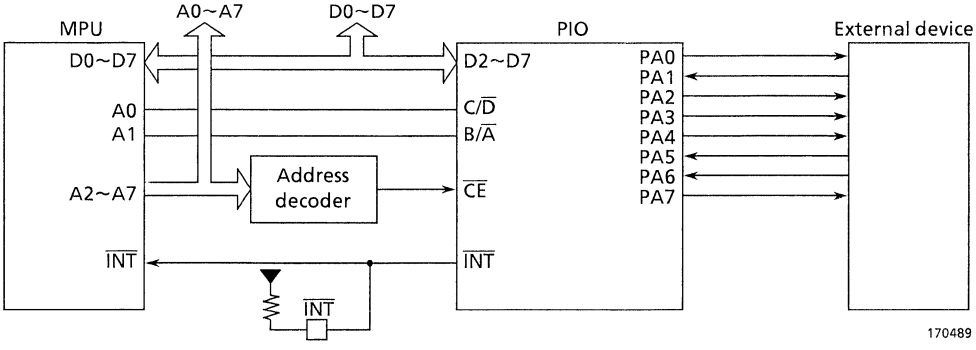


Figure 3.5.13 PIO Connection

```
ORG   100H
LD    SP, 100H      ····· Sets the stack pointer.
LM    2             ····· Sets for MPU mode 2 interrupt.
LD    A, 08H        ····· Writes data in MPU I register.
LD    I, A
;
LD    A, 02H        ····· Writes the interrupt vector word.
OUT   (1DH) , A
LD    A, 0CFH       ····· Writes the mode control word.
OUT   (1DH) , A
LD    A, 62H        ····· Writes the data I/O control word, Sets PIO.
OUT   (1DH) , A
LD    A, 0F7H       ····· Writes interrupt control word.
OUT   (1DH) , A
LD    A, 9FH        ····· Writes the mask control word.
OUT   (1DH) , A
;
EI                  ····· Sets interrupt enable.
```

3.6 SIO OPERATIONAL DESCRIPTION

The SIO has two independent, programmable full-duplex serial ports. These ports are assigned addresses on the TMPZ84C015A's I/O map. This subsection mainly describes the operations that take place after accessing the SIO.

3.6.1 SIO Block Diagram

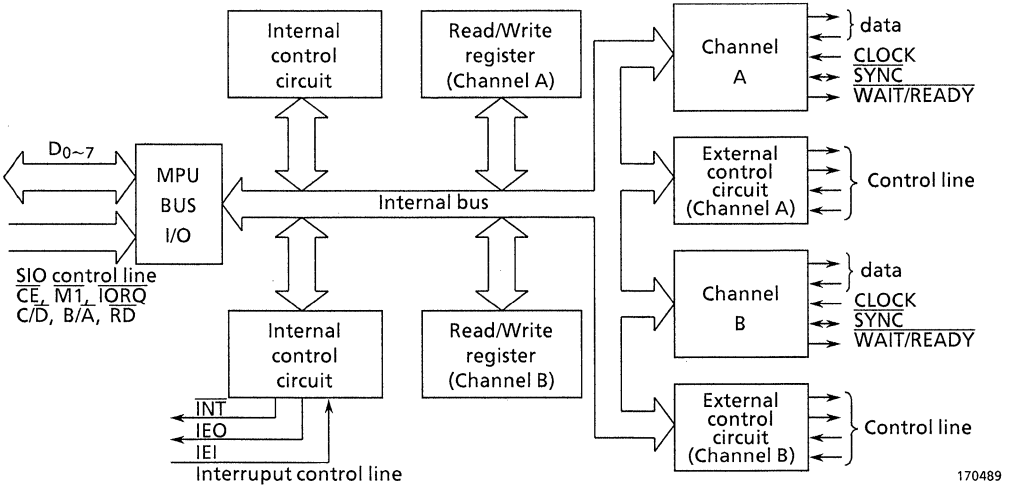


Figure 3.6.1 SIO Block Diagram

3.6.2 SIO System Configuration

As shown in Figure 3.6.1, the SIO consists of the MPU bus interface, the internal controller, the interrupt controller, and two independently operating full-duplex channels. Each channel has the read register, the write register, and the external controller which controls the connection with peripheral LSIs or external devices.

The TMPZ84C015A contains all the functions and pins of the 40-pin, DIP-type TMPZ84C40A (SIO/0), TMPZ84C41A (SIO/1), and TMPZ84C42A (SIO/2). However, when using the SIO of the TMPZ84C015A, the SIO/0, SIO/1, or SIO/2 must be used alone. The pin assignments are as shown in Figure 3.6.2.

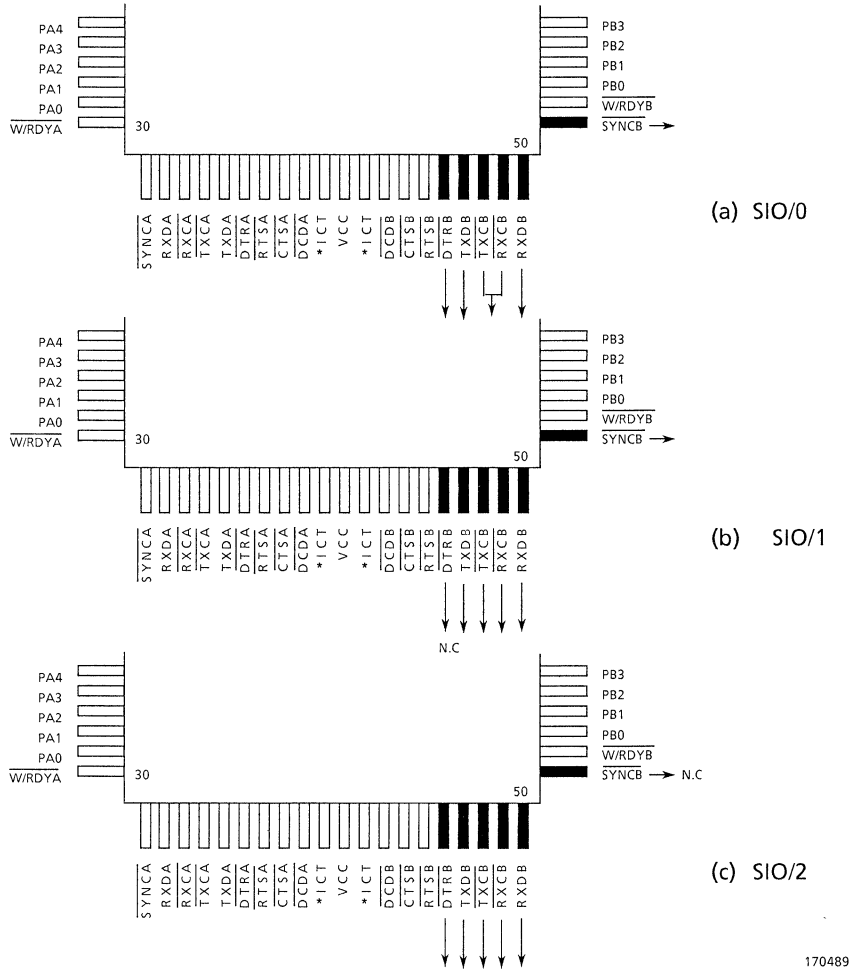


Figure 3.6.2 SIO Pin Assignments

Table 3.6.1 shows the types and functions of the SIO registers. Each channel has 8 write registers and 3 read registers.

(1) Communication data path

Figure 3.6.2 shows the communication path of each channel's transfer data.

1 Receive operation

The receiver has an 8-bit receive register and a 3-stage 8-bit buffer register in FIFO configuration. This saves time in high-speed data block transfers. The receivers also have the receive error FIFO which holds the status information such as parity and framing errors. The receive data follow different paths according to the operation mode and character length as shown in Figure 3.6.3.

Table 3.6.1 (a) Write Regosters

Register		Function
Write register0	(WR0)	Resets CRC. Sets pointers of registers, and commands.
Write register1	(WR1)	Sets the interrupt mode.
Write register2	(WR2)	Sets the vector to be transmitted at interrupt.
Write register3	(WR3)	Provides the parameters to control the receiver.
Write register4	(WR4)	Provides the parameters to control the receiver and transmitter.
Write register5	(WR5)	Controls the transmitter.
Write register6	(WR6)	Sets the sync character or the SDLC address field.
Write register7	(WR7)	Sets the sync character or the SDLC flag

170489

Table 3.6.1 (b) Read Regosters

Register		Function
Read register0	(RR0)	Indicates the receive/transmit buffer state and the pin state.
Read register1	(RR1)	Indicates the error status and the end-of-frame code.
Read register2	(RR2)	Indicates the interrupt vector contents. (Channel B only)

170489

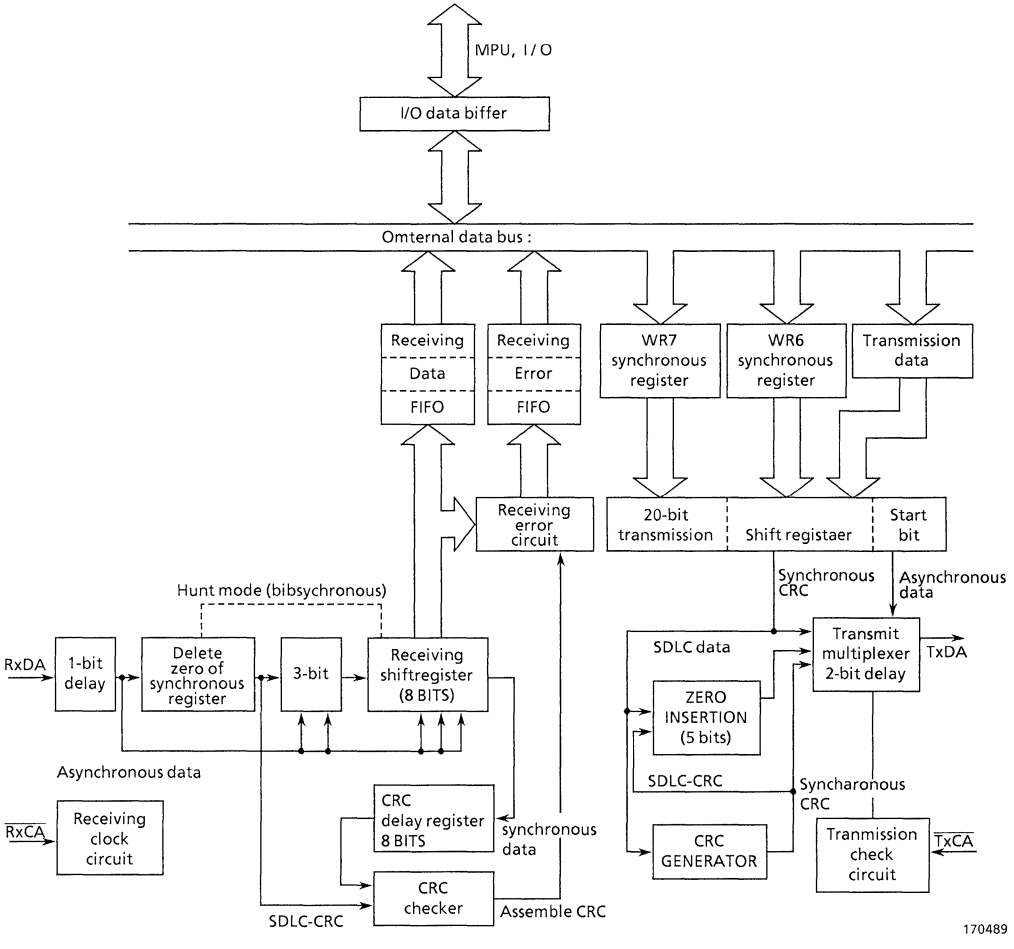


Figure 3.6.3 Transfer Data Path (Channel A)

- Asynchronous mode

  In the asynchronous mode, the receive data enters the 3-bit buffer if the character length is 7 or 8 bits or the 8-bit receive shift register if the character length is 5 or 6 bits.

- Synchronous mode

  In the synchronous mode, the data path depends on the receive processing phase at the time. The receiver operation starts from the hunt phase. In this mode, the receiver searches the receiver data for the bit pattern which matches the specified sync character. If the SIO is set in the monosync mode, the receiver searches for

the bit pattern which matches the sync character set in WR7;if the SIO is set in the bisync mode, the receiver searches for the bit pattern which matches two consecutive sync characters set in WR6 and WR7. When synchronization has been established, the subsequent data enter the 3-bit buffer by bypassing the sync register.

- SDLC mode

  In the SDLC mode, the sync register constantly monitors the receive data performing zero deletion as required. When the sync register detects 5 "1"s consecutively in the receive data, the following bit is deleted if it is "0". If it is "1", the bit that follows is checked. If it is "0", it is assumed as a flag, if it is "1", it is assumed an abort sequence (7 consecutive "1"s).

  The reformatted data are put in the receive shift register via the 3-bit buffer. When synchronization has been established, the subsequent data follow the same path regardless of the character length.

2 Transmission

The transmitter has an 8-bit transmit data register and a 20-bit transmit shift register. The 20-bit transmit shift register holds the data from the WR6, WR7, and transmit data register.

- Asynchronous mode

  In the asynchronous mode, the data in the 20-bit transmit shift register are added with the start and stop bits to be sent to the transmit multiplexer.

- Synchronous mode

  In the synchronous mode, the WR6 and WR7 hold the sync character. The contents of these registers are sent to the 20-bit transmit register as the sync character at the transmission of data blocks or as the idle sync character if a transmitter underrun occurs in data block transmission.

- SDLC mode

  In the SDLC mode, the WR6 holds the station address and the WR7 holds the flag. The flag (WR7) is sent to the 20-bit transmit register at the start and end of each frame. For each of the other data fields, one "0" follows five consecutive "1"s.

(2) I/O functions

To transfer data from the MPU, the SIO must be set in the polling, interrupt, or block transfer mode.

- Polling

  To operate the SIO in the polling mode, all interrupts mode must be disabled. In the polling mode, the MPU reads the status bits D0 and D2 in each channel's RR0 to check for reception or transmission.

- Interrupts

  There are 3 types of SIO interrupt: transmit interrupt, receive interrupt, and external/status interrupt. These interrupts can be enabled by program. The receive interrupt is further divided into the following three:

  - Interrupt on the first received character
  - Interrupt on all received characters
  - Interrupt on special receive conditions

    Higher priority is given to channel A than channel B. On the same channel, higher priority is given to reception, transmission, and external/status in this order.

  The SIO provides the daisy-chained interrupt priority control feature and the interrupt vector generating feature. Further, it provides the "status affected vector" feature. This feature outputs 4 interrupts depending on the interrupt source.

- Block transfer

  The SIO has the block transfer mode to adapt to the MPU"s block transfer and the DMA controller. For block transfer, the $\overline{W/RDY}$ line is used. For the MPU's block transfer, this line is used as the wait line;for the DMA block transfer, it is used as the ready line. The SIO's ready output indicates to the DMA controller that the data is ready to transfer. The SIO's wait output indicates to the MPU that the SIO is not ready for data transfer and therefore requesting the extension of the output cycle.

3.6.3 SIO Basic Operations

(1) Asynchronous mode

For data transfer in the asynchronous mode, the character length, clock rate, and interrupt mode must be set. These parameters are written in the write registers. Note that WR4 must be set before the other registers are set.

Data transfer does not start until the transmit enable bit is set. When the auto enable bit is set, the SIO starts transmission upon the $\overline{CTS}$ pin's going "0", allowing the programmer to send a message to the SIO without waiting for the $\overline{CTS}$ signal. Figure 3.6.4 shows the data format of the asynchronous mode.

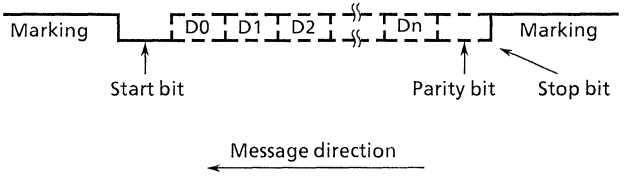


Figure 3.6.4 Data Format of Asynchronous Mode

1 Transmission

Serial data are output from the TxD pin. Its transfer clock rate can be set to one of 1, 1/16, 1/32, and 1/64 times the clock rate ot be supplied to the trandmit clock input ($\overline{TxC}$). The serial data are output on the falling edge of $\overline{TxC}$.

2 Reception

The receiver operation in the asynchronous mode starts when the receive enable bit (D0 of WR3) is set. When the receive data input RxD is set to "0" for the duration of at least 1/2 bit time, the SIO interprets it as the start bit, sampling the input data at the middle of the bit time. The sampling is performed on the rising edge of the $\overline{RxC}$ signal.

When the receiver receives the data whose character length is not 8 bits, it converts the data into the one composed of the necessary bits, the parity bit and the unused bit set to "1".

Example : a 6-bit character

"								"
1	P	D5	D4	D3	D2	D1	D0	

When the external/status interrupt is enabled and a break state is detected in the receive data, the interrupt is generated and the break/abort status bit (D7 of RR0) is set and the SIO monitors the transmit data until the break state is cleared. The interrupt is also generated when the $\overline{DCD}$ signal is in the inactive state for more than the specified pulse width. The DCD status bit is set to "1".

In the polling mode, the MPU must refer the receive character valid bit (D0 of RR0) to read the data. This bit is automatically reset when the receive buffer is read. In the polling mode, the transmit buffer status must be checked before writing data in the transmitter to avoid overwrite.

(2) Synchronous mode

There are 3 kinds of character synchronization : monosync, bisync, and external sync. In each of these synchronous modes, the times 1 clock rate is used for both transmission and reception. The receive data is sampled on the rising edge of the receive clock input ($\overline{RxC}$). The transmit data changes on the falling edge of the transmit clock input.

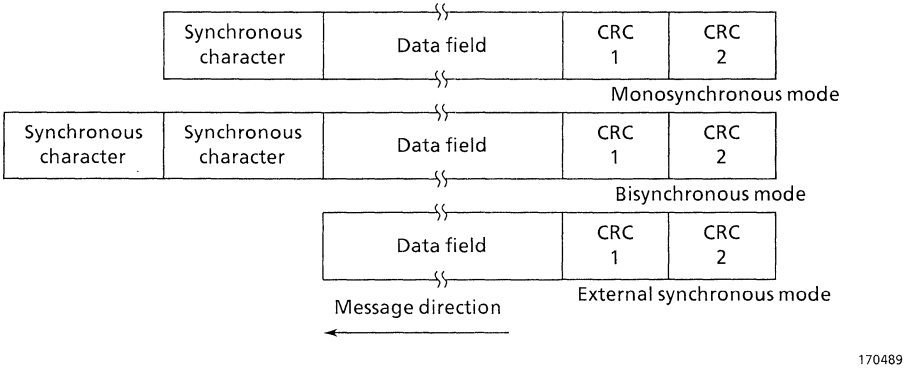


Figure 3.6.5 Data Format of Synchronous Mode

1 Monosync

In this mode, synchronization is established when a match with the sync character (8 bits) set to WR7 is found, enabling data transfer.

2 Bisync

In this mode, synchronization is established when a match with 2 consecutive sync characters set to WR6 and WR7 is found, enabling data transfer. In this mode as well as the monosync mode $\overline{SYNC}$ is active during the receive clock period in which the sync character is being detected.

3 External sync

In this mode, synchronization is performed externally. When synchronization is established, it is indicated by the $\overline{SYNC}$ pin. The $\overline{SYNC}$ input must be kept to "0" until the character synchronization is lost. Character assembly starts from the rising edge of the $\overline{RxC}$ after the falling of the $\overline{SYNC}$.

After reset, the SIO enters the hunt phase to search for the sync character. If synchronization is lost, the SIO sets the enter-hunt-phase-bit (D4 of WR3) to reenter the hunt phase.

- Transmission

(a) Data transfer using interrupt

When the transmit interrupt is enabled, the interrupt is caused upon the transmit buffer's being emptied. For the interrupt processing, other data are written in the transmitter. If these data are not ready for some reason, the transmit underrun condition occurs.

(b) Bisync mode

In the bisync mode, if the transmitter runs out of data during transmission, supply characters are inserted. This is done in two methods. In one method, sync characters are inserted. In the other, characters generated so far are transmitted followed by sync characters. Either of these methods can be selected by the reset transmit underrun/EOM command in WR0.

(c) End of transmission

Break can be performed by setting bit D4 of WR5. When break is performed, the data in the transmit buffer and the shift register are lost. When the external/status interrupt is enabled, the SIO generates the interrupt depending on the transmitter state and outputs the vector. This mode can be used for block transfer.

- Reception

(a) Interrupt on the first received character

This mode is used for ordinary block transfer. In this mode, the SIO generates the interrupt only for the first character;subsequently, it does not generate the interrupt unless special receive conditions are satisfied.

To initialize these settings, command 4 of WR0 (to be enabled by the next receive interrupt) must be set in advance.

(b) Interrupt on all received characters

In this mode, the SIO generates the interrupt for all characters coming into the receive buffer. When the status affect vector has been set, a special vector is generated on a special receive condition.

(c) Special receive condition interrupt

This interrupt occurs when any of the above interrupts is selected. The special receive conditions include parity error, receive overrun error, framing error, and end-of-frame (SDLC). These error status bits are latched, so that they must be reset after they are read. They can be reset by command 6 of WR0 (error reset).

(3) SDLC mode

The SIO supports both the SDLC and HDLC protocols. They resemble each other, so that only the SDLC mode is explained here.

Figure 3.6.6 shows the data format in the SDLC mode. In the SDLC mode, one data block is called a frame and the message in it is put between the open flag and the close. The address field in the frame contains the address of a secondary station. Checking this address, the SIO receives or ignores the frame.

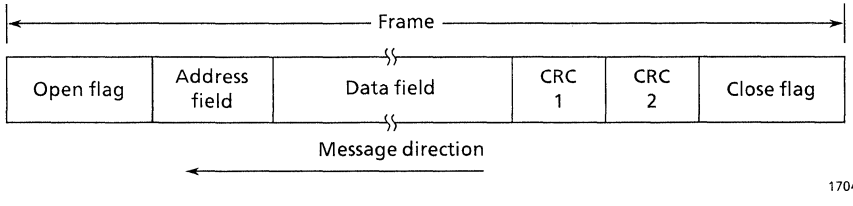


Figure 3.6.6 Data Format in SDLC Mode

- Transmission

(a) Data transfer using interrupt

When the transmit interrupt has been set, the interrupt occurs each time the transmit buffer becomes empty. In the SDLC mode, data are sent to the SIO by this interrupt.

(b) Data transfer using wait/ready

The wait function in the wait/ready capability is used to make the MPU extend the output cycle when the SIO's transmit buffer is not empty. The ready function indicates to the DMA that the SIO's transmit buffer is empty and therefore ready to receive data. If no data has been written in the transmit shift register before transmission, the SIO goes in the underrun state. This capability permits data transfer to the SIO.

(c) Transmit underrun/EOM

The SIO automatically ends the SDLC frame if there is no data to be transmitted to the transmit data buffer. To implement this, the SIO sends a 2-byte CRC when there is no data to send, then the SIO transmits one or more flags. After reset, the transmit underrun/EOM status bit is set to prevent the CRC character from being inserted when there is no data to be sent. Using this function, the SIO starts frame transmission. Here, the transmit underrun/EOM reset command must be set in advance between the transmission of the first data and the data end. Thus, the SIO goes in the reset state at the end of each message with the CRC character being sent automatically.

(d) CRC generation

For CRC calculation, the CRC generator must be reset before transmission (bits D6 and D7 of WR0). CRC calculation starts when the address field is written in the SIO (WR6). The transmit CRC enable bit (D0 of WR5) must be set before the address field is written.

(e) End of transmission

When the transmitter is disabled during transmission, the data currently transmitted is all transmitted to its end. The subsequent data is put in the marking state. When the transmitter is disabled, characters remain in the buffer. However, the abort sequence is made active when the abort command is written in the command register, deleting all data.

- Reception

As in the transmit mode, several parameters must be preset in the receive mode. The address field is written in WR7 and the flag character in WR7. Receiving the open flag, the receiver compares the contents of the following address field with the address set in WR6 or the global address ("1111 1111"). If the contents of the address field in frame matches either of these address, the SIO starts reception.

(a) Interrupt on the first received character

This mode is generally used for the block transfer using the wait/ready capability. In this mode, the SIO generates the interrupt only on the first character. The status flag of this interrupt is latched, so that command 4 (to be enabled by the next received character) of WR0 must be preset for re-initialization. When the external/status interrupt is set, an interrupt occurs every time the $\overline{DCD}$ changes. This interrupt also occurs when the special receive condition is satisfied.

(b) Interrupt on all received characters

In this mode, the SIO generates an interrupt on all received characters. When the status affect vector has been set, the SIO generates a special vector on the special receive condition interrupt.

(c) Special receive condition interrupt

Using the special receive condition, the interrupt on the first received character or the interrupt on all received characters must be selected in advance. The receive overrun status of the special receive condition interrupt is latched. The status bit can be reset by the error reset command (WR0 command).

(d) CRC check

The receive CRC check is reset when the open flag at the head of a frame is received. CRC calculation is performed on the subsequent characters up to the close flag. In the SDLC mode, the transmit CRC is inverted, so that a special check sequence is used. The check must end with "0001 1101 0000 1111." Since SIO handles the CRC character as a data, the MPU must discard it after reading it.

(e) End of transmission

When the SIO receives the close flag, the end-of-frame-bit is set to indicate that the close flag has been received. When the status affect vector has been set, the special receive condition interrupt occurs and the interrupt vector is output. Any frame can be aborted by abort transmission. When the external/status interrupt has been set, the interrupt occurs and the break/abort bit in RR0 is set.

3.6.4 SIO Status Transition Diagram and Basic Timing

[1] Status Transition Diagram

Figure 3.6.7 shows the SIO status transition diagram.

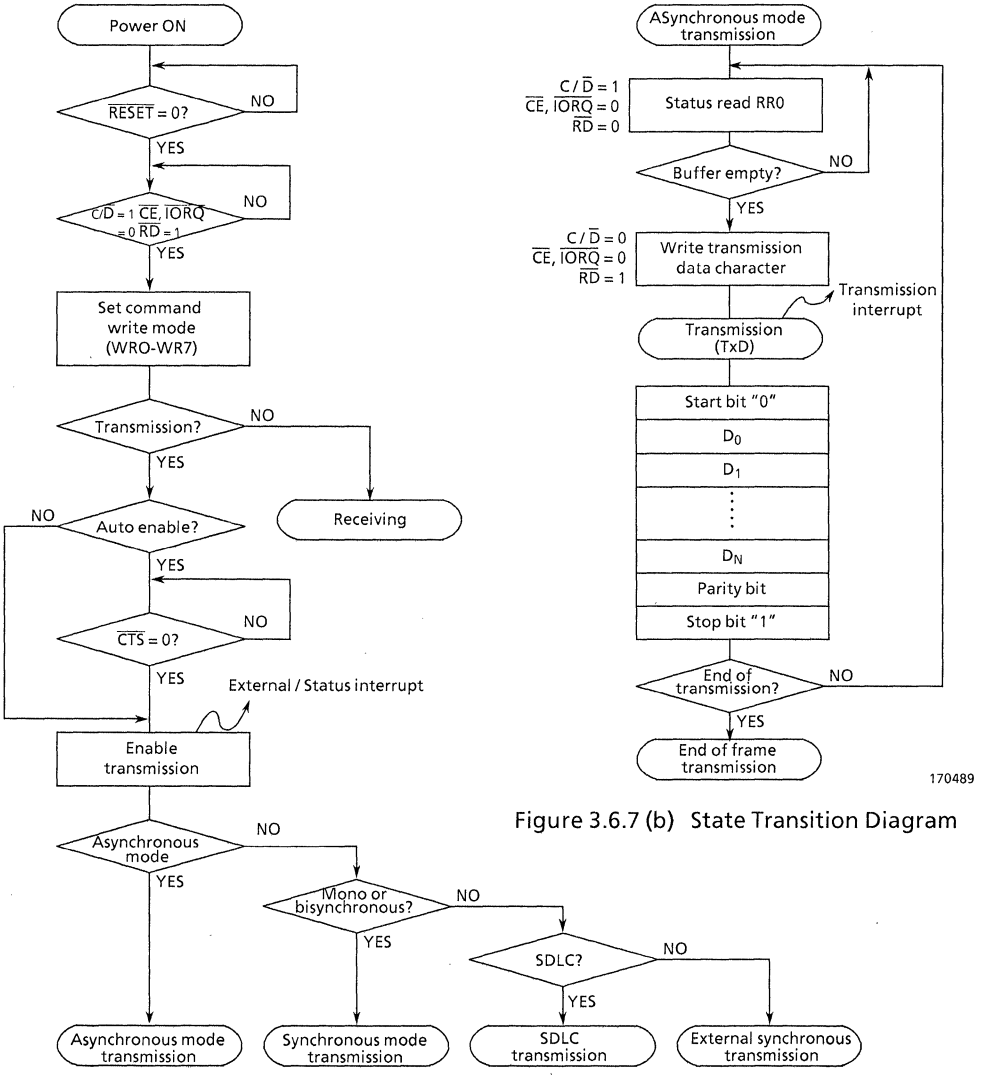


Figure 3.6.7 (b) State Transition Diagram

Figure 3.6.7 (a) SIO Status Transition Diagram

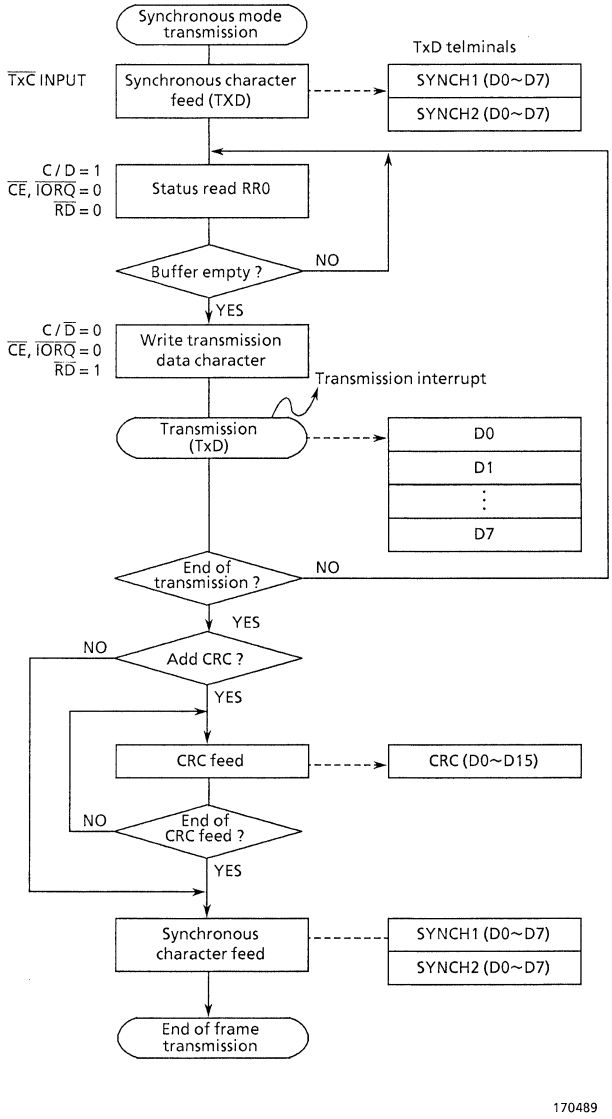


Figure 3.6.7 (c) State Transition Diagram

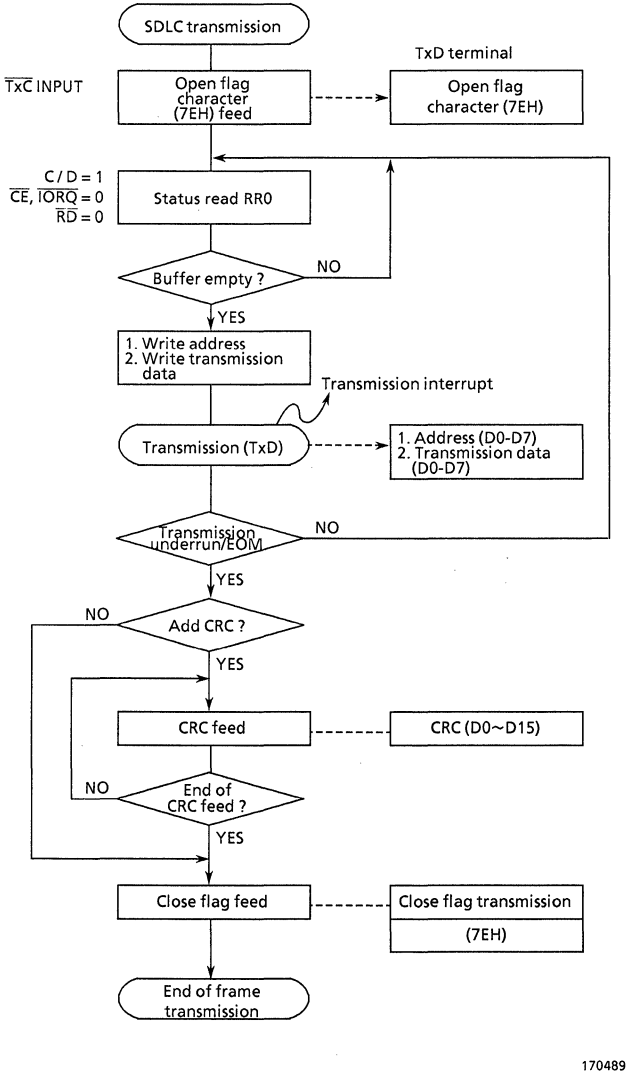


Figure 3.6.7 (d) SIO Status Transition Diagram

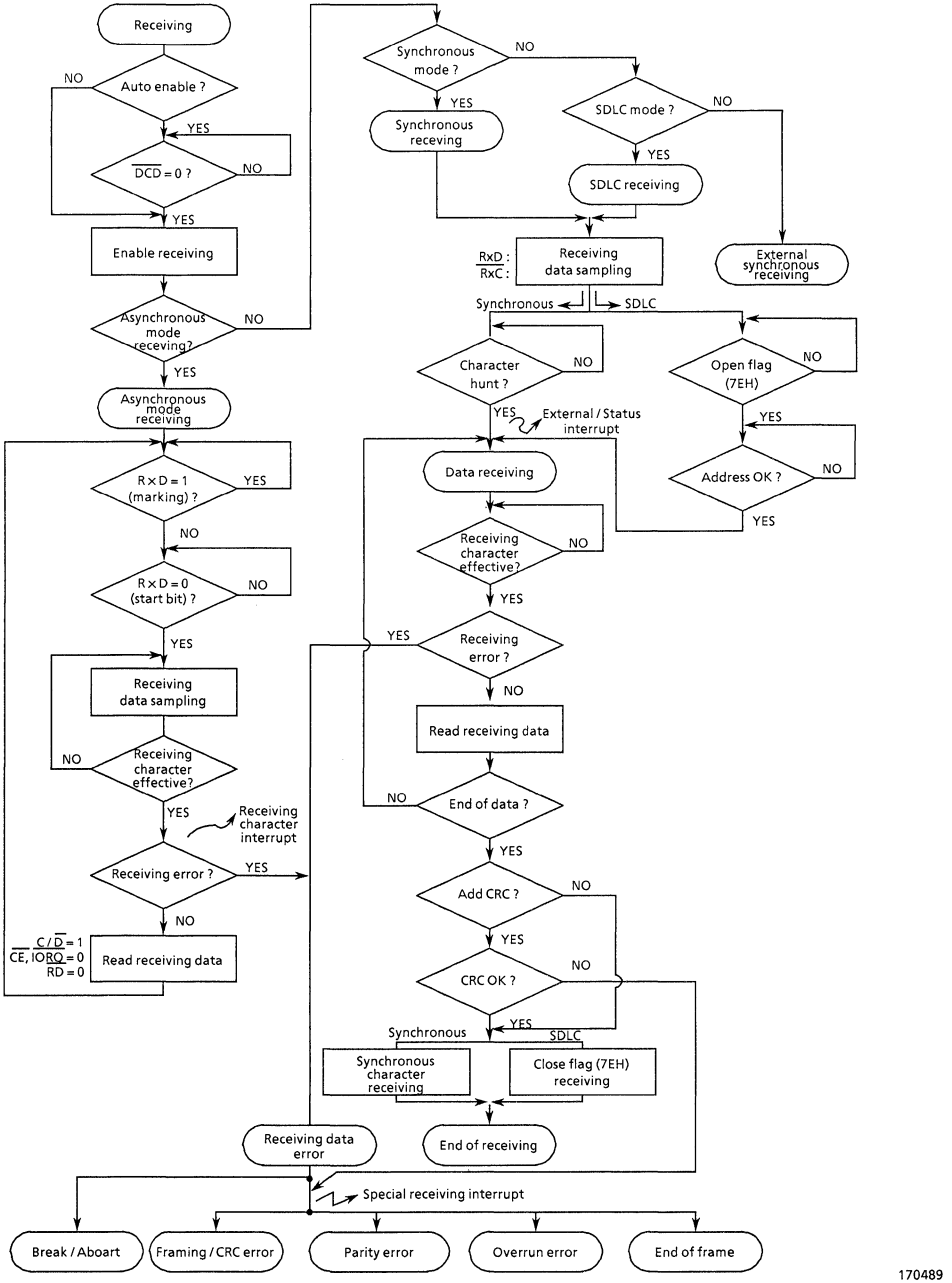


Figure 3.6.7 (e) State Transition Diagram

[2] Basic Timing

Figure 3.6.8 shows the timing in which data or a command is written from the MPU to the SIO. Figure 3.6.9 shows the timing in which data is read from the SIO to the MPU. Figure 3.6.10 shows the interrupt acknowledge timing in which the MPU gives an interrupt response to the SIO's interrupt request to set the $\overline{IORQ}$ pin to "0" several clocks after setting the $\overline{M1}$ pin to "0" as the acknowledge signal. To maintain the interrupt serviced state in daisy chain structure, the interrupt request state cannot be changed while $\overline{M1}$ is active.

Figure 3.6.11 shows the timing in which the return from interrupt is performed.

Figure 3.6.12 shows how the daisy chain structure works. First, suppose that the SIO is servicing interrupt. When the PIO issues an interrupt request immediately before the first byte "EDH" of the RETI instruction is decoded with $\overline{M1}$ being active, "IEO" of the PIO goes "0". However, when "EDH" is decoded, the PIO's interrupt request is not acknowledged. Therefore, the PIO's "IEO" returns to "1". When the second byte "4DH" is decoded, the SIO's "IEO" returns to "1". Therefore, the "IEI" and "IEO" of each peripheral LSI at this point of time all go "1", or out of the interrupt serviced state. The PIO keeps the $\overline{INT}$ pin at "0" until this state is set. Then, the interrupt is serviced starting with the peripheral device of the higher priority.

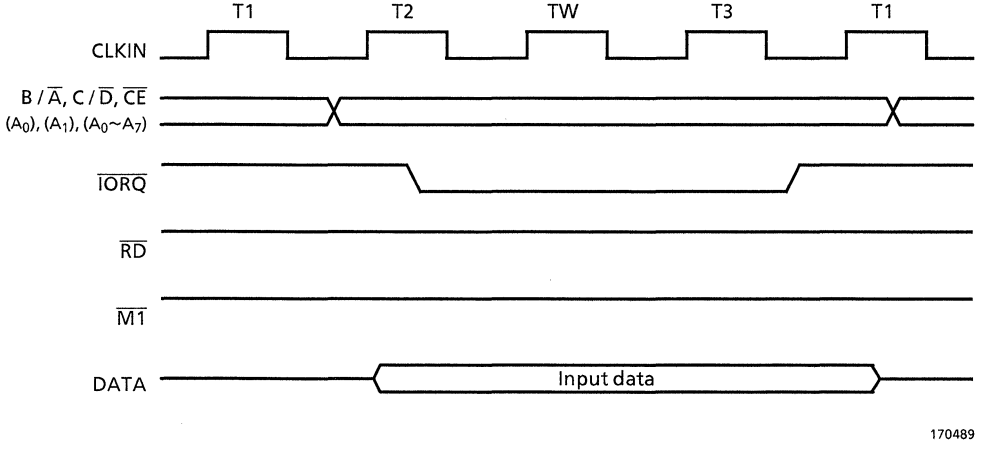


Figure 3.6.8 Write Timing

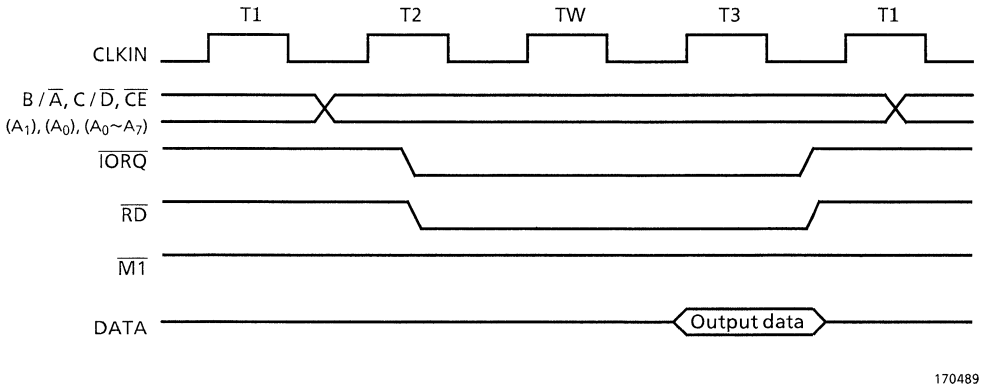


Figure 3.6.9 Read Timing

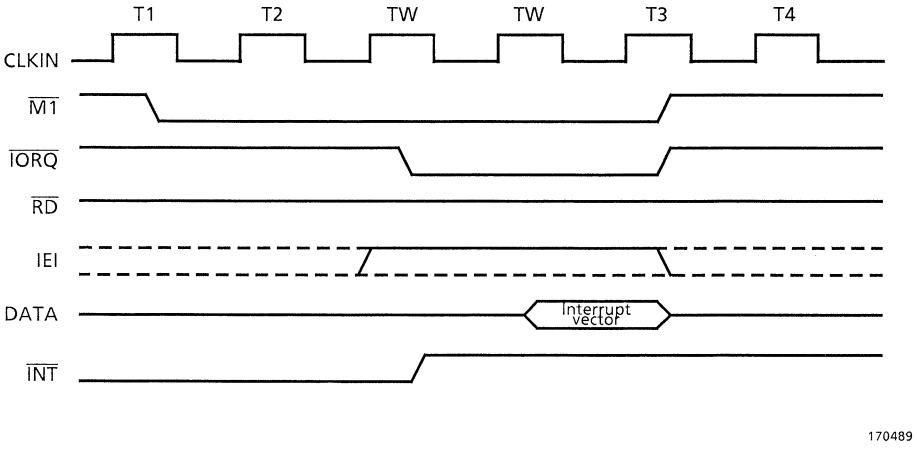


Figure 3.6.10 Interrupt Acknowledge Timing

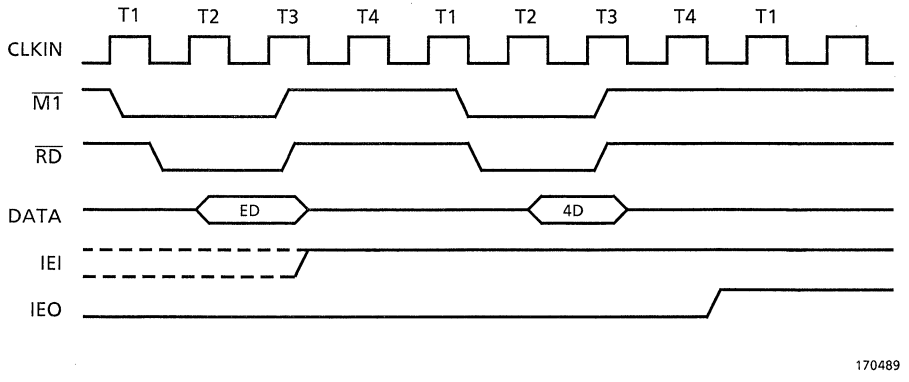
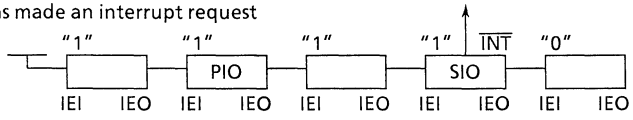
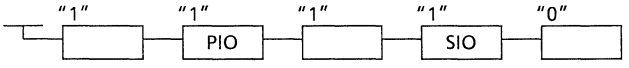


Figure 3.6.11 Return Timing from Interrupt

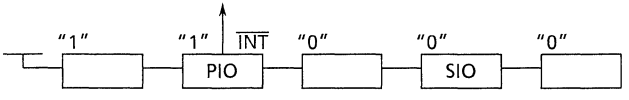
1 The SIO has made an interrupt request



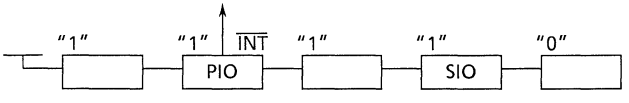
2 The SIO is servicing the interrupt.



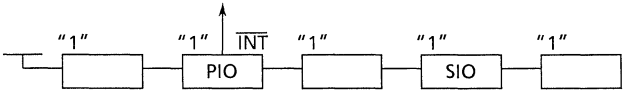
3 The PIO has made an interrupt request immediately before "EDH" is decoded by the SIO. By the PIO's interrupt request, PIO's IEO is set to "0".



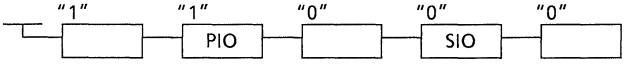
4 Because "EDH" has been decoded, the PIO's interrupt request is not acknowledged. Therefore, PIO's IEO returns to "1".



5 Because "4DH" has been decoded, the SIO's IEO is set to "1".



6 The PIO's interrupt request is acknowledged and the PIO's IEO is set to "0".



170489

Figure 3.6.12 Daisy Chain at Execution of RETI Instruction

3.6.5 SIO Operational Procedure

The following mainly describes the meaning of each bit of the write and read registers. Special attention should be directed to the fact that the parameters of the write register (WR4) should be set before the others.

Some registers can use only a signal channel. The I/O addresses listed in Table 3.6.2 must be specified to write the control word and read/write data on the SIO.

Table 3.6.2 I/O Addresses

I/O function	I/O address
Channel A data	#18
Channel A command	#19
Channel B data	#1A
Channel B command	#1B

170489

[I] Write Registers

1 WR 0; Write register 0

Table 3.6.3 Configuration of Write Register 0

D7	D6	D5	D4	D3	D2	D1	D0
CRC reset code		Primary command bit			Register pointer bit		

170489

Bits D0 through D2: Register pointer bits

These bits specify the register on which read/write is performed by the next byte. When read/write is completed, the register pointer points to WR0.

Bits D3 through D5: Basic command bits

- Command 0 (=000): No operation

  This command only sets the register pointer without making the SIO operate. It is used to invalidate the command in the command chain for the SIO or hold the location at which a command is inserted in the command chain if required.

- Command 1 (=001): Abort sequence generation

  This command is used to generate the abort sequence (7 or more consecutive "1"s). Note that command 1 is used only in the SDLC.

- Command 2 (=010): External/status interrupt reset

  Once an external interrupt or a status interrupt has occurred, the status bit of RR0 is latched. This command is issued to enable the RR0's status bit in order to enable the interrupt again.

- Command 3 (=011): Channel reset

  This command performs generally the same operation as when the $\overline{\text{RESET}}$ pin is set. The difference is that reset is performed only on a single channel. The command for channel A resets the interrupt priority circuit as well.

- Command 4 (=100): Enable the interrupt at the next character reception.

  This command is used to enable an interrupt when the end of block a data block has been detected followed by the reception of the next block.

- Command 5 (=101): Reset transmit interrupt pending

  If the transmit buffer becomes empty in the transmit interrupt enable mode, an interrupt occurs. This command is used to disable the transmit interrupt when there is no data in the transmit buffer.

- Command 6 (=110): Error reset

  The error (parity or overrun error) caused in block transfer is latched in bits D4 and D5 of RR1. This commands is used to clear these bits.

- Command 7 (=111): Return from interrupt

  This command performs the same operation as the operation required to execute the RETI instruction on the SIO's data bus. Therefore, non-Z80 MPUs (that is, systems using no RETI instruction) can use the daisy chain in the SIO. This command is available only on channel A.

Bits D6 and D7: CRC reset code

These 2 bits allow the programmer to select between the receive CRC checker reset, the transmit CRC generator reset, and the transmit underrun/EOM reset.

Table 3.6.4 List of Reset Command Codes

Reset command	D7	D6
No operation	0	0
Reset the receive CRC checker	0	1
Reset the transmit CRC generator	1	0
Reset the transmit underrun / EOM	1	1

170489

2 WR 1; Write register 1

Table 3.6.5 Configuration of Write Register 1

D7	D6	D5	D4	D3	D2	D1	D0
Wait / ready Enable	Wait / ready Select function	Select receiving / transmission	Receiving interrupt mode		Status affect vector	Enable trans-mission interrupt	Enable external / status interrupt

170489

Bit D0: External/status interrupt enable

When this bit is set, an interrupt is generated at the start of sync character transmission even if the execution is terminated upon detection of break/abort, the $\overline{DCD}$, $\overline{CTS}$ or $\overline{SYNC}$ signal has changed, or the transmit underrun/EOM latch is set.

Bit D1: Transmit interrupt enable

When this bit is set, a transmit interrupt is generated upon the transmit buffer becoming empty.

Bit D2: Status affect vector

When this bit is set, bits D1 through D3 (V1 through V3) of WR2 is changed. When this bit is not set, the same interrupt vector as the contents of WR2 issued. Note that this bit is available only on channel B.

Bits D3 and D4: Receive interrupt mode

These bits are used to select a receive interrupt mode.

Bits D5 through D7: Selection wait/ready functions

These 3 bits are used to select a $\overline{W/RDY}$ pin function. The wait or the ready function is selected by program and they are not used simultaneously. The meaning of these bits are:

- When D5 is set to "1" , it indicates that the $\overline{W/RDY}$ pin responds to the receive buffer; when D5 is reset to "0", it indicates that the pin responds to the transmit buffer.
- When D6 is set to "1", the $\overline{W/RDY}$ pin functions as the $\overline{READY}$ pin; when D6 is reset to "0", the pin functions as the $\overline{WAIT}$ pin.

- When D7 is set to "1", the wait/ready function is enabled;when D7 is reset to "0", the function is disabled.

For example, when D7, D6, and D5 are "1", "1", and "0" respectively, and the transmit buffer is full, the $\overline{\text{READY}}$ pin goes "1", when the transmit buffer is empty, the pin goes "0".

Table 3.6.6 shows the summary of the above description of bits D3 and D4 and D5 through D7.

Table 3.6.6 List of Receive Interrupt Mode Codes

Receive interrupt mode	D4	D3
Receive interrupt disable	0	0
Interrupt on first received character or special receive condition*	0	1
Interrupt on received character or special receive condition*	1	0
Interrupt on received character or special receive condition * (except for parity error)	1	1

170489

*Special receive conditions:

- End of frame (in SDLC mode only)
- Receive overrun error
- Parity error
- Framing error

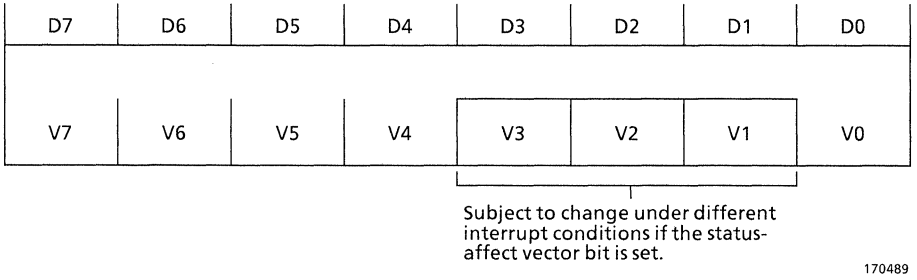
Table 3.6.7 Wait / Ready Select Function (D5 through D7)

Pin state			Buffer state	D7	D6	D5
Pin (Function)		Pin output				
DISABLE	$\overline{\text{WAIT}}$	Floating	-	0	0	–
	$\overline{\text{READY}}$	High	-		1	
ENABLE	$\overline{\text{WAIT}}$	Low	The transmit buffer is full and the SIO data port is selected.	1	0	0
		Floating	The transmit buffer is empty.			
	$\overline{\text{READY}}$	High	The transmit buffer is full.		1	
		Low	The transmit buffer is empty.			
	$\overline{\text{WAIT}}$	Floating	The receive buffer is full.		0	1
		Low	The receive buffer is empty and the SIO data port is selected.			
	$\overline{\text{READY}}$	Low	The receive buffer is full.		1	
		High	The receive buffer is empty.			

170489

3 WR 2; Write register 2

Table 3.6.8 Configuration of Write Register 2



This write register is the interrupt vector register. When bit D2 of WR1 (B channel) is not set, the interrupt vector is issued. When bit D2 of WR1 (B channel) is set, bits D1 through D3 (V1 through V3) are changed depending on the interrupt generation condition. This time, the contents of WR2 remain unchanged. Because WR2 is available only on channel B, WR2 must be programmed even if only channel A of the SIO is used.

Table 3.6.9 shows the WR2 bit states in the interrupt condition with the status affect vector being set.

Table 3.6.9 Channel Interrupt Condition Codes

Channel	Interrupt condition	V3	V2	V1
B	Transmit buffer empty	0	0	0
	Change of external / status	0	0	1
	Received character condition available	0	1	0
	Special receive condition*	0	1	1
A	Transmit buffer empty	1	0	0
	Change of external / status	1	0	1
	Received character available	1	1	0
	Special receive condition*	1	1	1

170489

*Special receive conditions:

- End of frame (in SDLC mode only)
- Receive overrun error
- Parity error
- Framing error

4 WR 3; Write register 3

Table 3.6.10 Configuration of Write Register 3

D7	D6	D5	D4	D3	D2	D1	D0
Receiving bit / character		Auto enable	Enter hunt phase	Enable receiving CRC	Address search mode	Prohibit synchro-nous character load	Enable receiving

170489

Bit D0: Receive enable

When this bit is set, the receive operation starts. Because this bit is used to start the receive operation, it must be set after the receive-associated programming has been all completed.

Bit D1: Sync character load inhibit

When this bit is set in the sync mode, the sync character is not loaded into the receive buffer. This bit is used to remove the sync character and idle sync from the received characters.

Bit D2: Address search mode

When this bit is set in the SDLC mode, any message having a programmed address or an address other than the global address (FFH) is not received by WR6. Therefore, the receive interrupt does not occur unless an address match occurs.

Bit D3: Receive CRC enable

When this bit is set, CRC calculation starts at the start of the last data transfer from the receive shift register to the receiver buffer.

Bit D4: Enter hunt Phase

When the establishment of synchronization is required, set this bit to enter the SIO into the hunt phase. The hunt phase is automatically cleared upon establishment of synchronization.

Bit D5: Auto enable

When this bit is set, the transmitter is enabled at the time the $\overline{CTS}$ pin is "0". When the $\overline{DCD}$ pin is "0", the receiver is enabled.

Bits D6 and D7: Receive character length

These bits are used to specify the number of receive bits which make up one character (character length). Table 3.6.11 shows the number of bits per character.

Table 3.6.11 Receive Character Length Codes

Bits / character	D7	D6
5	0	0
7	0	1
6	1	0
8	1	1

170489

5 WR 4; Write register 4

Table 3.6.12 Configuration of Write Register 4

D7	D6	D5	D4	D3	D2	D1	D0
Clock mode		Synchronous mode		Stop bit		Parity	
						Even / $\overline{\text{Odd}}$	Enable

170489

Bit D0: Parity enable

When this bit is set, 1-bit transmit data is added to the number of bits specified by D6 and D7 of WR3 and the data is received in the resulting number of bits. If a character length other than 8 bits is selected, the added parity bit is set to the MSB side to be transferred to the receive data FIFO. When the 8-bit character length is selected, the parity bit is not transferred to the receive data FIFO.

Bit D1: Parity even/odd

This bit is used to determine whether to perform transfer and check in even or odd parity. (Even parity = "1", odd parity = "0")

Bit D2 and D3: Stop bit length

These bits are used to select the stop bit length in the asynchronous mode. In the synchronous mode, both D2 and D3 must be set to "0".

Table 3.6.13 Stop Bit Length Codes

Stop bit	D3	D2
Sync mode	0	0
1 stop bit / character	0	1
1.5 stop bits / character	1	0
2 stop bits / character	1	1

170489

Bits D4 and D5: Sync mode

These bits are used to select the sync mode.

Table 3.6.14 Sync Mode Codes

Sync mode	D5	D4
8-bit sync mode	0	0
16-bit sync mode (bisync mode)	0	1
SDLC mode (flag character ; 7EH)	1	0
External sync mode	1	1

170489

Bits D6 and D7: Clock mode

These bits are used to select the factor between the transmit/receive clock and the data transfer rate. In the synchronous mode, the ×1 clock mode must be set. In the asynchronous mode, the transmit side and the receive side must have the same factor.

Table 3.6.15 Clock Mode Codes

Clock mode (data transfer rate)	D7	D6
× 1 data transfer rate	0	0
× 16 data transfer rate	0	1
× 32 data transfer rate	1	0
× 64 data transfer rate	1	1

170489

6 WR5; Write register 5

Table 3.6.16 Configuration of Write Register 5

D7	D6	D5	D4	D3	D2	D1	D0
DTR	Transmit bit / character		Break transmission	Enable transmission	CRC-16 / SDLC	RTS	Enable CRC transmission

170489

Bit D0: Transmit CRC enable

When this bit is set at the time the transmit data is loaded from the transmit data buffer into the transmit shift register, the CRC calculation is performed on that data. If this bit is not set, the CRC calculation and transmission are not performed in the transmit underrun state in the synchronous or SDLC mode.

Bit D1: Request to send

When this bit is set, the $\overline{RTS}$ pin goes "0". When this bit is not set, the $\overline{RTS}$ pin goes "1". In the asynchronous mode, the $\overline{RTS}$ pin goes "1" when the transmit buffer becomes empty. In the synchronous or SDLC mode, this bit state is followed by the $\overline{RTS}$ pin state.

Bit D2: CRC-16/SDLC

When this bit is set, the CRC-16 polynomial ($X^{16}+X^{15}+X^2+1$) is selected. When this bit is reset to "0", the CRC-CCITT polynomial ($X^{16}+X^{12}+X^5+1$) is selected.

Bit D3: Transmit enable

When this bit is set, the transmitter is enabled. Even if this bit is reset to "0" after the start of transmission, the sync character and the data being transmitted are transmitted to the last.

Bit D4: Transmit break

When this bit is set, transmitting any data forcibly puts the transmit data line (TxD pin) in the space state. When this bit is reset to "0", the TxD pin is put in the marking state.

Bits D5 and D6: Transmit character length

These bits indicate the character length of transmit data.

Table 3.6.17 Transmit Character Length Codes

Bits / character	D6	D5
Less than 5 bits	0	0
7 bits	0	1
6 bits	1	0
8 bits	1	1

170489

As shown in Table 3.6.17, for the transmission of less than 5 bits (4 bits or 3 bits) per character, D6 and D5 are "0" and "0", which do not indicate how many bits the transmit data consists of. To solve this problem, the data characters must be processed by the format shown in Table 3.6.18. Note that D indicates data.

Table 3.6.18 Data Transfer Format with Transmit Data Consisting of Less than 5 bits

Transmit bits / character	D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	1	1	0	0	0	D
2	1	1	1	0	0	0	D	D
3	1	1	0	0	0	D	D	D
4	1	0	0	0	D	D	D	D
5	0	0	0	D	D	D	D	D

170489

Bit D7: Data terminal ready

This bit indicates the $\overline{DTR}$ pin state. When this bit is set, the $\overline{DTR}$ pin goes "0", when it is reset, the $\overline{DTR}$ pin goes "1".

7 WR6; Write register 6

Table 3.6.19 Configuration of Write Register 6

D7	D6	D5	D4	D3	D2	D1	D0
SYNC							
7	6	5	4	3	2	1	0

170489

This register is programmed as follows:

- In the external sync mode : Transmit sync character
- In the monosync mode : Transmit sync character
- In the bisync mode : First sync character
- In the SDLC mode : Slave station address

8 WR7; Write register 7

Table 3.6.20 Configuration of Write Register 7

D7	D6	D5	D4	D3	D2	D1	D0
SYNC							
15 (7)	14 (6)	13 (5)	12 (4)	11 (3)	10 (2)	9 (1)	8 (0)

170489

This register is programmed as follows:

- In the monosync mode : Receive sync character
- In the bisync mode : Second sync character
- In the SDLC mode : Flag character (7EH)

This register is not used in the external sync mode.

[2] Read Registers

1 RR 0; READ REGISTER 0

Table 3.6.21 Configuration of Read Register 0

D7	D6	D5	D4	D3	D2	D1	D0
Break/ Abort	Trasmission underrun /EOM	CTS	Synchro- nize /Hunt	DCD	Trasmission buffer empty	Interrupt pending	Receiving character effective

Used with the external / status interrupt

170489

Bit D0: Receive character available

This bit is set when the receive buffer holds characters of 1 byte or more. This bit is reset when the buffer becomes empty.

Bit D1: Interrupt pending

This bit is set when an interrupt occurs in the SIO regardless of the interrupt condition type. This bit is available only on channel A.

Bit D2: Transmit buffer empty

This bit is set when the transmit data buffer becomes empty or the SIO is reset. However, in the sync and SDLC modes where the CRC character is being transmitted, bit D2 is reset.

Bit D3: Data carrier detect

This bit indicates the $\overline{DCD}$ pin input state. This bit is latched when the external/status interrupt occurs.

Bit D4: Sync/hunt

The meaning of this bit depends on the operation mode:

(i) Asynchronous mode

Bit D4 indicates the SIO's $\overline{SYNC}$ pin state. When the $\overline{SYNC}$ pin state changes, the external/status interrupt occurs.

(ii) External sync mode

When synchronization has been established by the detection of external synchronization, the last bit of the sync character must be set to "0" at the second $\overline{RxC}$ falling edge from the rising edge of the received $\overline{RxC}$. That is, to set the $\overline{SYNC}$ input to "0" by the external circuit after the detection of synchronization, full 2 receive cycle clocks must be awaited.

When the $\overline{SYNC}$ input goes "0", the sync hunt bit is set. When synchronization is lost or the end of message is detected, the enter hunt phase bit is set.

(iii) Internal sync mode

In the monosync and bisync modes, bit D4 is initialized to "1" by the enter hunt phase command (D4 of WR3). This bit is reset when the SIO detects the sync character.

(iv) SDLC mode

Bit D4 is set when the receiver is disabled or the enter hunt phase command is issued. Then, when the frame open flag is detected, this bit is reset.

Bit D5: Clear to send

This bit indicates the opposite of the $\overline{CTS}$ pin input state.

Bit D6: Transmit underrun/EOM

This bit is set when the SIO is reset (including channel reset). Only the reset transmitter underrun/EOM latch command (WR0 bits D7, D6="1", "1") can reset this bit. When the transmit underrun state occurs, the external/status interrupt is generated. Bit D5 is also used to control transmission in the sync or SDLC mode.

Bit D7: Break/abort

In the asynchronous mode in reception, this bit indicates the break state detection. When the break state is detected, this bit is set, generating the external/status interrupt. This bit is reset by the external/status interrupt reset command.

After break, the external/status interrupt is generated again. In the SDLC mode, bit D7 is set when the abort sequence is detected, generating the external/status interrupt.

2 RR 1; READ REGISTER 1

Table 3.6.22 Configuration of Read Register 1

D7	D6	D5	D4	D3	D2	D1	D0
End of frame	framing error	Receiving overrun error	Parity error	Fraction			Feed all characters

170489

Bit D0: All sent

In the asynchronous mode, this bit is set when all characters are sent from the transmitter or there is no transmit data in the SIO. In the synchronous mode, this bit is always set.

Bits D1 through D3: Fraction codes

Normally, I field is an integral multiple of character length. If it is not, these bits show the number of fraction bits. These codes are effective only for the transmission for which the end of frame bit is set in the SDLC mode.

Example: Figure 3.6.13 shows examples of fractions in which the number of bits/character at the end of I field is 8 bits (1) and 4 bits (2).

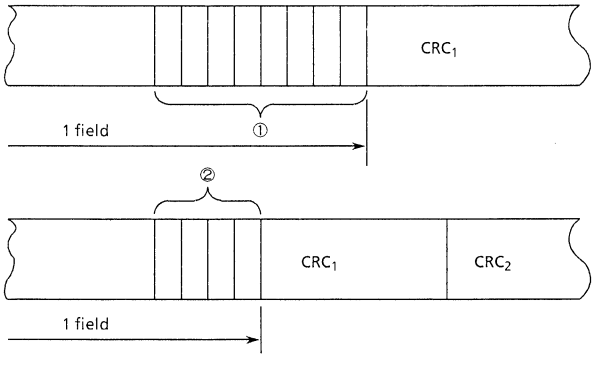


Figure 3.6.13 Examples of Fraction Bits Field

Table 3.6.23 (a) shows the fraction codes for the receive character whose character length is 8 bits.

Table 3.6.23 (a) Bit Patterns by Fraction Bits at End of I Field

Number of fraction bits at end of I field		D3	D2	D1
1 byte before	2 bytes before			
0	3	1	0	0
0	4	0	1	0
0	5	1	1	0
0	6	0	0	1
0	7	1	0	1
0	8	0	1	1
1	8	1	1	1
2	8	0	0	0

170489

The same table can also be provided for each character length when the receive character length of I field is other than 8 bits.

Table 3.6.23 (b) Bit Patterns by Number of Bit / Character (No Fractions)

Bits / character	D3	D2	D1
5 bits / Character	0	0	1
6 bits / Character	0	1	0
7 bits / Character	0	0	0
8 bits / Character	0	1	1

170489

Bit D4: Parity error

This bit is latched when the parity select bit (D0 of WR4) is set and a parity error is detected in the receive data. Latch can be cleared by the error reset command (WR0 bits D5, D4, D3 = "1", "1", "0").

Bit D5: Receive overrun error

The receive data FIFO holds up to 3 characters. When more characters are received without read out by the MPU, the excess character is set to the receive FIFO. When this character is read by the MPU, this receive overrun error is set. Once set, bit D5 latches that state. When the error reset command (command 6 of WR0 bits D3 through D5) is written, this bit is also reset.

Bit D6: CRC/framing error

In the asynchronous mode, this bit is set when a framing error is detected in the received character. Because this bit is not latched, it is always updated.

In the synchronous and SDLC modes, this bit indicates the transmitted CRC check result. This bit is reset when the error reset command (command 6 of WR0 bits D3 through D5) is written.

Bit D7: End of frame

This bit is set when the end flag is detected in the receive data and the CRC check and the fraction code are found normal. This bit is reset when the error reset command (command 6 of WR0 bits D3 through D5) is written. This bit is used only in the SDLC mode and is updated when the first character of the next frame is received.

3 RR 2; Read register 2

Table 3.6.24 Configuration of Read Register 2

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt vector							
V7	V6	V5	V4	V3	V2	V1	V0

Subject to change under different interrupt conditions if the status-affect bit is set

170489

When the status affect vector bit (D2 of WR1 (Channel B)) is set, bits V3 through V1 are changed depending on the interrupt condition at the time. The vector to be read is determined by the interrupt condition having the highest priority at the time of read. When the status affect vector bit is reset, the contents of this register are the same as those of WR2.

3.6.6 Using SIO

The following describes some system examples using the SIO. Figure 3.6.14 shows an inter-processor communication system. In this example, the MPU on the left side controls the data transfer with the modules on the right side. Both diagrams shown in Figure 3.6.14 (a) and (b) are communication systems. As shown, the SIO is used to interface with external devices in data communication. The greatest advantage of the SIO is the smaller number of data lines than parallel communication.

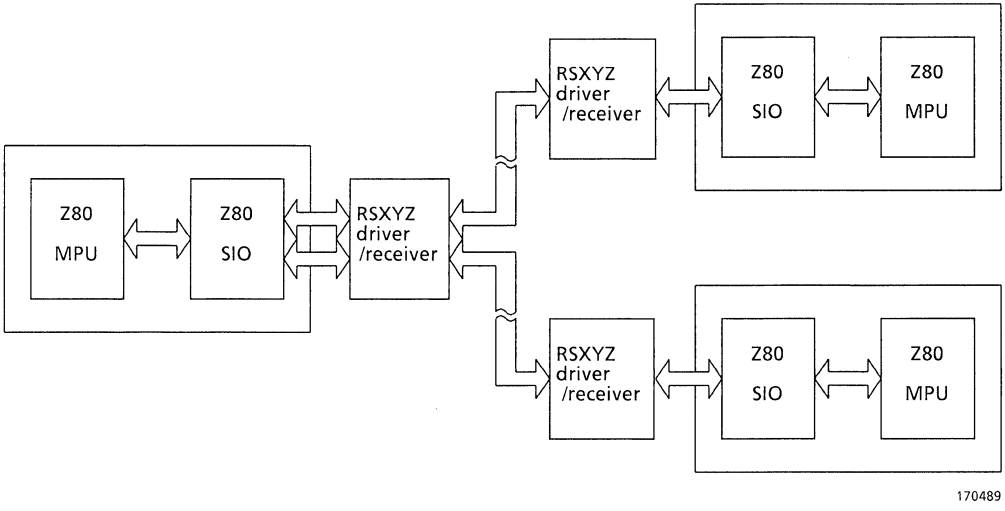


Figure 3.6.14 (a) Example of Data Communication Between Processors

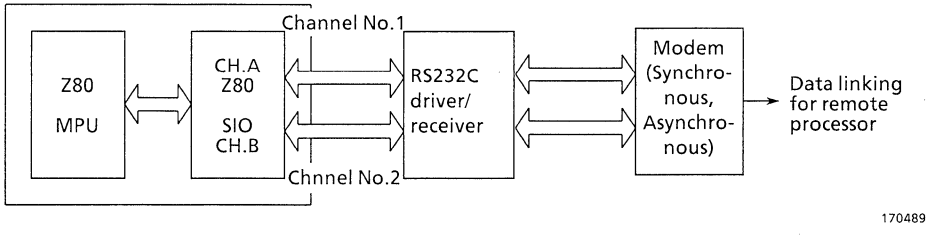


Figure 3.6.14 (b) Example of Data Communication Between Processors

3.7 STANDBY CAPABILITY

When a HALT instruction is executed, the TMPZ84C015A is put in one of the Run, Idle-1, Idle-2, or Stop mode depending on the contents of the halt mode setting register (#F0:bit 4, bit 3:HALTMR). (However, the TMPZ84C015A is put in the Run mode immediately after the reset operation by the $\overline{\text{RESET}}$ pin.) The halt mode setting register is set as follows. For the description and timing of each mode, see Subsection 3.3 "CGC Operations. "

The halt mode setting register is assigned to bits 4 and 3 of address F0 in the I/O address area. The halt mode is releasad by the interrupt (the nonmaskable interrupt by the $\overline{\text{NMI}}$ pin or the maskable interrupt by the $\overline{\text{INT}}$ pin) or by the reset through the $\overline{\text{RESET}}$ pin. A maskable interrupt is accepted when the MPU is in the EI state (in the state after the execution of EI instruction). A nonmaskable interrupt is accepted unconditionally. When an interrupt is accepted, the interrupt processing starts.

When the MPU is in the DI state (after the reset operation and the execution of DI instruction) with maskable interrupt, the TMPZ84C015A returns to the halt mode after executing a HALT instruction (actually a NOP instruction).

3.7.1 Setting Halt Mode

Duplicate control is provided to prevent the stop of the watchdog timer operation which may be caused by the halt mode setting error due to program runaway.

The halt mode is set by the halt mode setting register (HALTMR) and the halt mode control register (#F1:bits 7 through 0:HALTMCR). Figure 3.7.1 shows the contents of the halt mode control register (HALTMCR). Figure 3.7.2 shows the contents of the halt mode setting register (HALTMR).

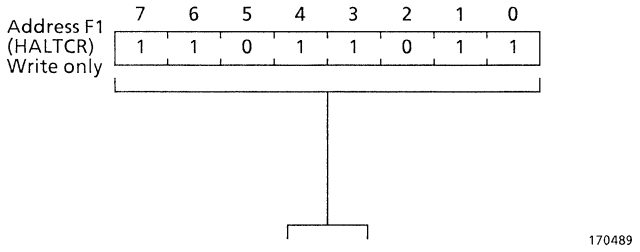


Figure 3.7.1 Halt Mode Control Register (HLTMCR)

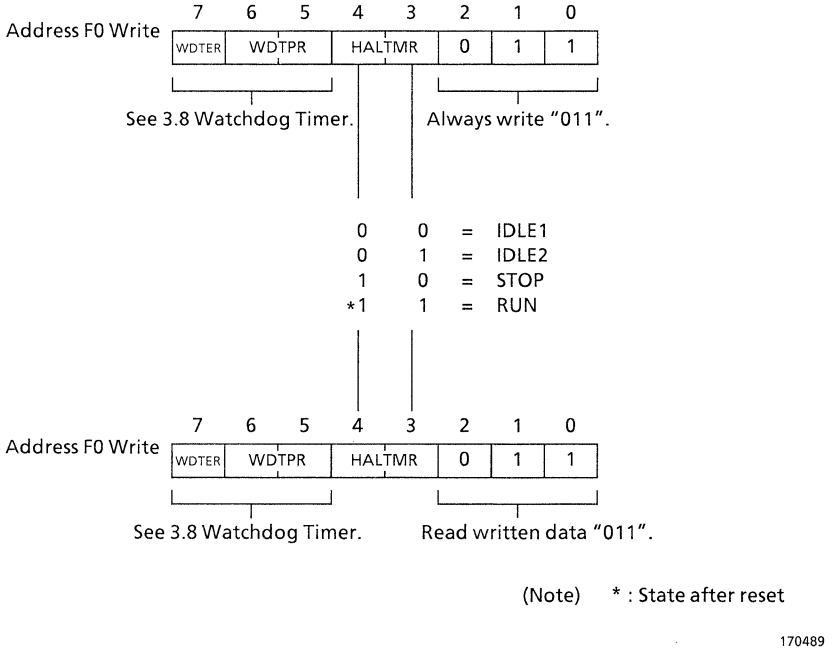


Figure 3.7.2 Halt Mode Set Register

Figure 3.7.3 shows the device states in the halt state with the CLKOUT pin connected to the CLKIN pin.

MODE	CGC	MPU	CTC	PIO	SIO	Watchdog Timer (WDT)	CLKOUT PIN
IDLE1	○	×	×	×	×	×	×
IDLE2	○	×	○	×	×	×	○
STOP	×	×	×	×	×	×	×
RUN	○	○	○	○	○	○	○

170489

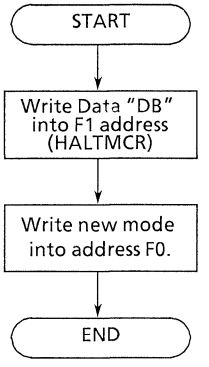
○ ····· Operating
× ····· Stop
Note : CLKOUT and CLKIN must be connected.

Figure 3.7.3 Device States in Halt State

For the halt mode in which the clock is supplied from the CLKIN pin (with the CGC oscillator unused), the Run mode must be used.

3.7.2 Halt Mode Setting procedure

After reset, the halt mode is changed to the Run mode. Figure 3.7.4 shows the procedure to set a new mode.



170489

Figure 3.7.4 Setting Halt Mode

3.8 WATCHDOG TIMER

The watchdog timer (WDT) detects an operation error caused by the program runaway to return to the normal operation.

3.8.1 Block Diagram of Watchdog Timer

Figure 3.8.1 shows the block diagram of the watchdog timer.

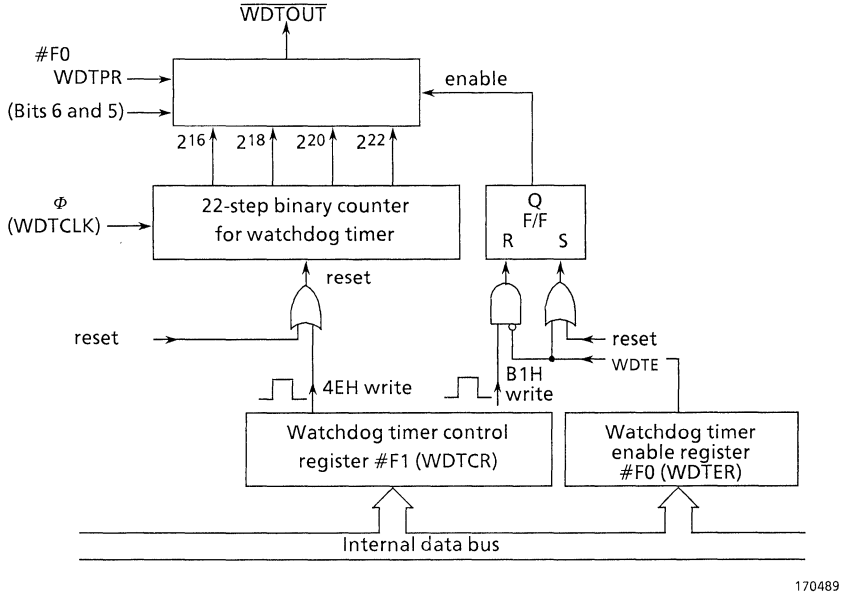


Figure 3.8.1 Block Diagram of Watchdog Timer

3.8.2 Setting watchdog Timer

(1) Enabling the watchdog timer

The watchdog timer can be set by the watchdog timer enable register (#F0:bit 7:WDTER) and the watchdog timer periodic register (#F0:bit 6, bit 5:WDTPR).

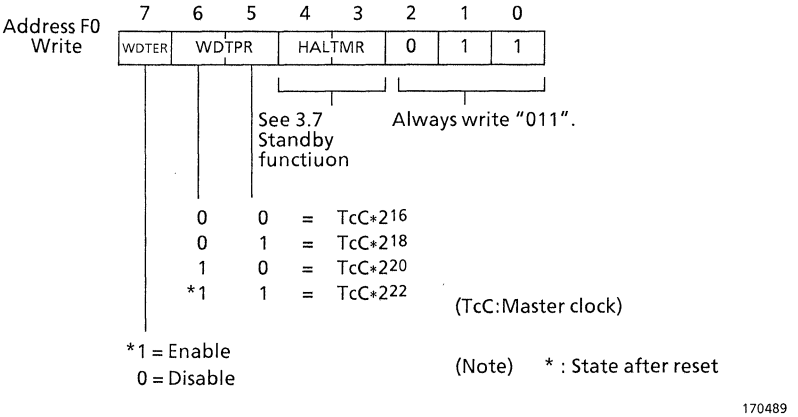


Figure 3.8.2 Enabling Watching Timer

(2) Disabling the watchdog timer

The watchdog timer can be disabled by disabling the watchdog timer enable register (WDTER) then writing data "B1" in the watchdog timer control register (#F1:bit 7 through bit 0:WDTCR).

This function has a duplicate structure to prevent the watchdog timer setting error, which may lead to the watchdog timer operation stop, caused by program runaway.

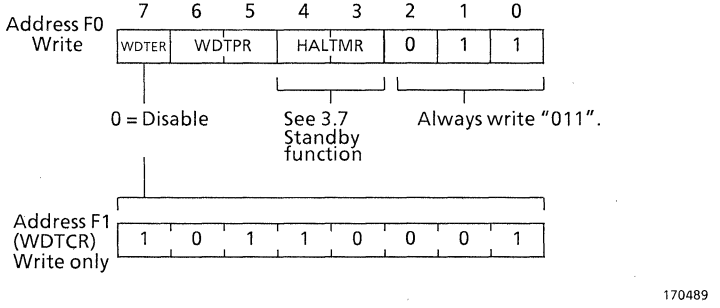


Figure 3.8.3 Disabling Watchdog Timer

(3) Clearing the watchdog timer

The watchdog timer can be cleared by writing data "4E" in the watchdog timer control register (WDTCR).

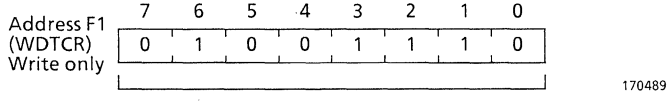


Figure 3.8.4 Clearing Watchdog Timer

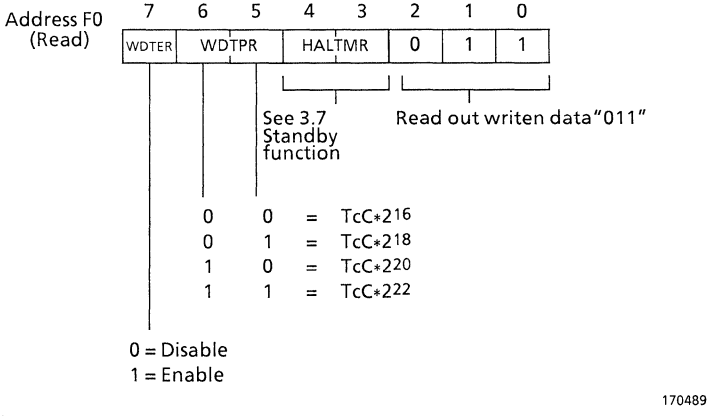


Figure 3.8.5 Reading Watchdog Timer Setting Register

3.8.3 Watchdog Timer Output

When the enabled watchdog timer is used, the "0" level signal is output to the $\overline{\text{WDTOUT}}$ pin after the duration of time specified in the watchdog timer periodic register (WDTPR). The output pulse width is one of the following two types depending on the $\overline{\text{WDTOUT}}$ pin connection:

(1) The $\overline{\text{WDTOUT}}$ connected to the $\overline{\text{RESET}}$ pin:The "0" level pulse of 5TcC (System clock) is output.

(2) The $\overline{\text{WDTOUT}}$ connected to a pin other than $\overline{\text{RESET}}$ pin :The "0" level pulse is kept output until the watchdog timer is cleared by software or reset by the $\overline{\text{RESET}}$ pin.

3.9 INTERRUPT PRIORITY

The programmable interrupt priority register (#F4:bits 2 through 0:INTPR) is provided to determine the interrupt priority for the CTC, SIO, and PIO in the TMPZ84C015A.

3.9.1 Setting Interrupt Priority

Figure 3.9.1 shows the register to determine the daisy chain interrupt priority for the CTC, SIO, and PIO.

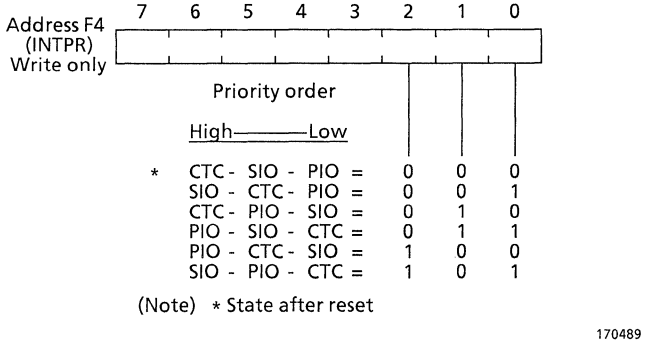


Figure 3.9.1 Interrupt priority Register (INTPR)

Example :

When "101" is written in address F4 (INTPR), the daisy chain interrupt priority is given as shown in Figure 3.9.2.

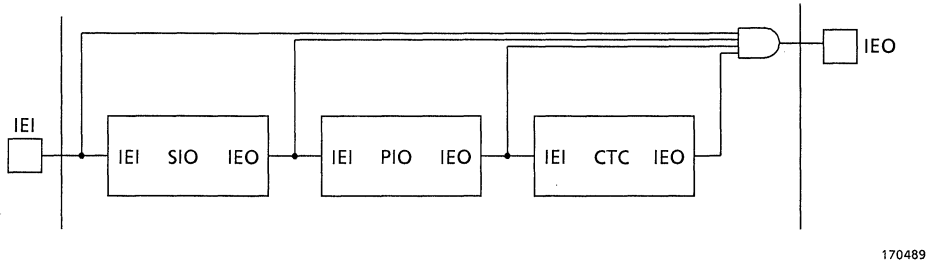


Figure 3.9.2 Daisy Chain Interrupt Priority

4. ELECTRICAL CHARACTERISTICS

4.1 MAXIMUM RATINGS

SYMBOL	ITEM		RATING
VCC	Vcc Supply Voltage with respect to Vss		− 0.5V to + 7.0V
VIN	Input Voltage		− 0.5V to Vcc + 0.5V
PD	Power Dissipation (6MHz VERSION : TA = 85°C) (8MHz VERSION : TA = 70°C)		250mW
TSOLDER	Soldering Temperature (Soldering Time 10 sec)		260°C
TSTG	Storage Temperature		− 55°C to 125°C
TOPR	Operating Temperature	6MHz VERSION	− 40°C to 85°C
		8MHz VERSION	− 10°C to 70°C

300389

4.2 DC Electrical Characteristics

6MHz VERSION: TOPR = −40°C to +85°C, VCC = 5V ± 10%, VSS = 0V
8MHz VERSION: TOPR = −10°C to +70°C, VCC = 5V ± 5%, VSS = 0V

(1/2)

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT.
VILC	Clock Input Low Voltage (CLKIN)		− 0.3	−	0.6	V
VIHC	Clock Input High Voltage (CLKIN)		VCC − 0.6	−	VCC + 0.3	V
VIL	Input Low Voltage (except XTAL1. $\overline{\text{RESET}}$)		− 0.5	−	0.8	V
VIH	Input High Voltage (except XTAL1. $\overline{\text{RESET}}$)		2.2	−	VCC	V
VILR	Input Low Voltage ($\overline{\text{RESET}}$)		− 0.5	−	0.45	V
VIHR	Input High Voltage ($\overline{\text{RESET}}$)		VCC − 0.6	−	VCC	V
VOLC	Output Low Voltage (CLKOUT)	IOL = 2.0mA	−	−	0.6	V
VOHC	Output High Voltage (CLKOUT)	IOH = − 2.0mA	VCC − 0.6	−	−	V
VOL	Output Low Voltage (except CLKOUT)	IOL = 2.0mA	−	−	0.4	V
VOH1	Output High Voltage 1 (except CLKOUT)	IOH = − 1.6mA	2.4	−	−	V
VOH2	Output High Voltage 2 (except CLKOUT)	IOH = − 250μA	VCC − 0.8	−	−	V
ILI	Input Leakage Current	Vss ≦ VIN ≦ Vcc	−	−	± 10	μA
ILO	3-state Output Leakage Current in Float	Vss ≦ Vout ≦ VCC	−	−	± 10	μA

300389

(2/2)

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.	MAX.	UNIT.
ICC1	Power Supply Current	VCC = 5V fCLK = (1) VIHC = VIHR = VIH = VCC-0.2V, VILC = VILR = VIL = 0.2V	AF-6	—	25	35	mA
			AF-8	—	35	45	
ICC2	Stand-by Supply Current (See Note (2))	VCC = 5V fCLK = (1) VIHC = VIH = VIHR = VCC-0.2V, VILC = VIL = VILR = 0.2V	AF-6 /AF-8	—	0.5	50	μA
ICC3	Power Supply Current (IDLE 1 Mode)	VCC = 5V fCLK = (1) VIHC = VIH = VIHR = VCC-0.2V, VILC = VIL = VILR = 0.2V	AF-6	—	1.5	3	mA
			AF-8	—	2	4	
ICC4	Power Supply Current (IDLE 2 Mode)	VCC = 5V fCLK = (1) VIHC = VIH = VIHR = VCC-0.2V, VILC = VIL = VILR = 0.2V	AF-6	—	11	15	mA
			AF-8	—	15	20	mA

300389

Note 1 : fCLK = 1/TcC (MIN)

Note 2 : ICC2 Stand-by Supply Current is guaranteed only when the supplied clock is stopped at a low level during T4 state of the following machine Cycle (M1) next to OP code fetch Cycle of HALT instruction.
Except SYNCA = 0 or SYNCB = 0 state

4.3 AC ELECTRICAL CHARACTERISTICS (1) (in Active State)

6MHz VERSION : TA = −40°C~85°C, VCC = 5V ± 10%, VSS = 0V
8MHz VERSION : TA = −10°C~70°C, VCC = 5V ± 5%, VSS = 0V
(8MHz Version guarantee AC characteristics in only Active State.)

4.3.1 AC Characteristics of MPU (in Active State)

(1/3)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
1	TcC	Clock Cycle Time	162	—	DC	125	—	DC	ns
2	TwCh	Clock Pulse Width (High)	65	—	DC	50	—	DC	ns
3	TwCl	Clock Pulse Width (Low)	65	—	DC	50	—	DC	ns
4	TfC	Clock Fall Time	—	—	20	—	—	15	ns
5	TrC	Clock Rise Time	—	—	20	—	—	15	ns
6	TdCr (A)	Clock ↑ to Address Valid Delay	—	—	90	—	—	85	ns
7	TdA (MREQf)	Address Valid to $\overline{MREQ}$ ↓ Delay	35	—	—	20	—	—	ns
8	TdCf (MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay	—	—	70	—	—	60	ns
9	TdCr (MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay	—	—	70	—	—	60	ns
10	TwMREQh	$\overline{MREQ}$ Pulse Width (High)	65	—	—	45	—	—	ns
11	TwMREQl	$\overline{MREQ}$ Pulse Width (Low)	135	—	—	100	—	—	ns
12	TdCf (MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay	—	—	70	—	—	60	ns
13	TdCf (RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay	—	—	80	—	—	70	ns
14	TdCr (RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay	—	—	70	—	—	60	ns
15	TsD (Cr)	Data Setup Time to Clock ↑	30	—	—	30	—	—	ns
16	ThD (RDr)	Data Hold Time to $\overline{RD}$ ↑	0	—	—	0	—	—	ns
17	TsWAIT (Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	60	—	—	50	—	—	ns
18	ThWAIT (Cf)	$\overline{WAIT}$ Hold Time after Clock ↓	10	—	—	10	—	—	ns
19	TdCr (M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay	—	—	80	—	—	70	ns
20	TdCr (M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay	—	—	80	—	—	70	ns
21	TdCr (RFSHf)	Clock ↑ to $\overline{RFSH}$ ↓ Delay	—	—	110	—	—	95	ns
22	TdCr (RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay	—	—	100	—	—	85	ns
23	TdCf (RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay	—	—	70	—	—	60	ns
24	TdCr (RDf)	Clock ↑ to $\overline{RD}$ ↓ Delay	—	—	70	—	—	60	ns

030489

(2/3)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
25	TsD (Cf)	Data Setup to Clock ↓ during M2, M3, M4 or M5 Cycles	40	—	—	30	—	—	ns
26	TdA (IORQf)	Address Stable prior $\overline{IORQ}$ ↓	110	—	—	75	—	—	ns
27	TdCr (IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay	—	—	65	—	—	55	ns
28	TdCf (IORQr)	Clock ↓ to $\overline{IORQ}$ ↑ Delay	—	—	70	—	—	60	ns
29	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	25	—	—	5	—	—	ns
30	TdCf (WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay	—	—	70	—	—	60	ns
31	TwWR	$\overline{WR}$ Pulse Width	135	—	—	100	—	—	ns
32	TdCf (WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay	—	—	70	—	—	60	ns
33	TdD (WRf)	Data Stable Prior to $\overline{WR}$ ↓	-55	—	—	55	—	—	ns
34	TdCr (WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay	—	—	60	—	—	55	ns
35	TdWRr (D)	Data Stable from $\overline{WR}$ ↑	30	—	—	15	—	—	ns
36	TdCf (HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓	—	—	260	—	—	225	ns
37	TwNMI	$\overline{NMI}$ Pulse Width	80	—	—	80	—	—	ns
38	TsBUSREQ (Cr)	$\overline{BUSREQ}$ Setup Time to Clock ↑	50	—	—	40	—	—	ns
39	ThBUSREQ (Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	10	—	—	10	—	—	ns
40	TdCr (BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay	—	—	90	—	—	80	ns
41	TdCf (BUSACKr)	Clock ↓ to $\overline{BUSACK}$ ↑ Delay	—	—	90	—	—	80	ns
42	TdCr (Dz)	Clock ↑ to Data Float Delay	—	—	80	—	—	70	ns
43	TdCr (CTz)	Clock ↑ to Control Outputs Float Delay ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)	—	—	70	—	—	60	ns
44	TdCr (Az)	Clock ↑ to Adress Float Delay	—	—	80	—	—	70	ns
45	TdCr (A)	$\overline{MREQ}$ ↑, $\overline{IORQ}$ ↑, $\overline{RD}$ ↑, and $\overline{WR}$ ↑ to Address Hold Time	35	—	—	20	—	—	ns
46	TsRESET (Cr)	$\overline{RESET}$ to Clock ↑ Setup Time	60	—	—	45	—	—	ns
47	ThRESET (Cr)	$\overline{RESET}$ to Clock ↑ Hold Time	10	—	—	10	—	—	ns
48	TsINTf (Cr)	$\overline{INT}$ to Clock ↑ Setup Time	70	—	—	55	—	—	ns

030489

(3/3)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
49	TsINTr (Cr)	$\overline{INT}$ to Clock ↑ Hold Time	10	—	—	10	—	—	ns
50	TdM1f (IORQf)	$\overline{M1}$ ↓ to $\overline{IORQ}$ ↓ Delay	365	—	—	270	—	—	ns
51	TdCf (IORQf)	Clock ↓ to $\overline{IORQ}$ ↓ Delay	—	—	70	—	—	60	ns
52	TdCr (IORQr)	Clock ↑ to $\overline{IORQ}$ ↑ Delay	—	—	70	—	—	60	ns
53	TdCf (D)	Clock ↓ to Data Valid Delay	—	—	130	—	—	115	ns

030489

4.3.2 AC Characteristics of CGC (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
54	TcC CLK	Output-Clock Cycle	—	162	—	—	125	—	ns
55	TwCh CLK	Output-Clock Width(High)	—	70	—	50	—	—	ns
56	TwCl CLK	Output-Clock Width(Low)	—	70	—	50	—	—	ns
57	TfC CLK	Output Clock fall time	—	12	—	—	—	12	ns
58	TrC CLK	Output Clock rise time	—	12	—	—	—	12	ns
59	TRST (INT) S	CLKOUT restart time by $\overline{INT}$ (STOP Mode)	—	2^{14} + 2.5TcC	—	—	2^{14} + 2.5TcC	—	ns
60	TRST (NMI) S	CLKOUT restart time by $\overline{NMI}$ (STOP Mode)	—	2^{14} + 2.5TcC	—	—	2^{14} + 2.5TcC	—	ns
61	TRST (INT) I	CLKOUT restart time by $\overline{INT}$ (IDLE 1 / 2 Mode)	—	2.5 *TcC	—	—	2.5 *TcC	—	ns
62	TRST (NMI) I	CLKOUT restart time by $\overline{NMI}$ (IDLE 1 / 2 Mode)	—	2.5 *TcC	—	—	2.5 *TcC	—	ns
63	TRST (RESET) I	CLKOUT restart time by $\overline{RESET}$ (IDLE 1 / 2 Mode)	—	TcC	—	—	TcC	—	ns

030489

4.3.3 AC Characteristics of CTC (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
64	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall (in case of generating only interrupt immediately before M1 cycle)	—	—	160	—	—	130	ns
65	TdIEI (IEOf)	Delay from IEI fall to IEO fall	—	—	70	—	—	50	ns
66	TdIEI (IEOr)	Delay from IEI rise to IEO rise (after ED decode)	—	—	150	—	—	120	ns
67	IsCLK (INT)	CLK / TRG setup to TL ↑ for detection of interrupt tsCTR (c) Satisfied	TcC + 120 + T68 + T48	—	—	TcC + 100 + T68 + T48	—	—	ns
		tsCTR (c) not Satisfied	2TcC + 120 + T68 + T48	—	—	2TcC + 100 + T68 + T48	—	—	
68	TcCTR	CLK / TRG Frequency (counter mode)	2TcC	—	—	2TcC	—	—	ns
69	TrCTR	CLK / TRG rising time	—	—	40	—	—	30	ns
70	TfCTR	CLK / TRG falling time	—	—	40	—	—	30	ns
71	TwCTR1	CLK / TRG Pulse Width (Low)	120	—	—	90	—	—	ns
72	TwCTRh	CLK / TRG Pulse Width (High)	120	—	—	90	—	—	ns
73	TsCTR (Cs)	CLK/TRG ↑ to Clock ↑ Setup Time for Immediate count (counter mode)	150	—	—	110	—	—	ns
74	TsCTR (CT)	CLK/TRG ↑ to Clock ↑ Setup Time for enabling of Prescaler on following clock ↑ (timer mode)	150	—	—	110	—	—	ns
75	TdC (ZC / TOr)	Clock ↑ to ZC/TO ↑ Delay	—	—	190	—	—	110	ns
76	TdC (ZC / TOf)	Clock ↓ to ZC/TO ↓ Delay	—	—	140	—	—	110	ns

030489

4.3.4 AC Characteristics of PIO (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
77	TdM1 (IEO)	Delay from $\overline{M1}$ fall to IEO fall	—	—	100	—	—	100	ns
78	TsIEI (IO)	IEI Set-up time for $\overline{IORQ}$ fall (INTA cycle)	100	—	—	80	—	—	ns
79	TdIEI (IEOf)	Delay from IEI fall to IEO fall	—	—	70	—	—	50	ns
80	TdIEI (IEOr)	Delay from IEI rise to IEO rise	—	—	150	—	—	120	ns
81	TdC (RDYr)	Delay from clock fall to READY rise	—	—	170	—	—	150	ns
82	TdC (RDYf)	Delay from clock fall to READY fall	—	—	120	—	—	110	ns
83	TwSTB (C)	$\overline{STROBE}$ pulse width	120	—	—	100	—	—	ns
84	TsSTB (C)	Set-up time of $\overline{STROBE}$ rise for clock fall (in case of making READY to active by next cycle)	150	—	—	100	—	—	ns
85	TdIO (PD)	Delay from $\overline{IORQ}$ rise to port data stable (Mode 0)	—	—	160	—	—	140	ns
86	TsPD (STB)	Port Data set-up time for $\overline{STROBE}$ rise (Mode 1)	190	—	—	150	—	—	ns
87	TdSTB (PD)	Output Port data delay time from $\overline{STROBE}$ fall (Mode 2)	—	—	180	—	—	150	ns
88	TdSTB (PDr)	Delay from $\overline{STROBE}$ rise to data float (Mode 2)	—	—	160	—	—	120	ns
89	IsSTRB	$\overline{STROBE}$ setup to TL↑ for detection of interrupt	350 + T48	—	—	350 + T48	—	—	ns
90	IsPD	Port data stable setup to TL↑ for detection of interrupt (Mode 3)	430 + T48	—	—	350 + T48	—	—	ns
91	ThPD (STB)	Data Hold time for $\overline{STROBE}$ rise (Mode 1)	40	—	—	40	—	—	ns

030489

4.3.5 AC Characteristics of SIO (in Active State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
92	TsM1 (C)	$\overline{M1}$ ↑ to clock ↑ Setup time	75	—	—	50	—	—	ns
93	TsIEI (IO)	IEI ↓ to $\overline{IORQ}$ ↓ Setup time (INTACK cycle)	120	—	—	100	—	—	ns
94	TdM1 (IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (interrupt before $\overline{M1}$)	—	—	160	—	—	120	ns
95	TdIEI (IEOr)	IEI ↑ to IEO ↑ Delay (after ED decode)	—	—	150	—	—	120	ns
96	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	—	—	50	ns
97	TdIO (W/RWf)	$\overline{IORQ}$ ↓ or $\overline{CE}$ ↓ to $\overline{W/RDY}$ ↓ Delay (Wait mode)	—	—	175	—	—	130	ns
98	TdC (W/RRf)	Clock ↑ to $\overline{W/RDY}$ ↓ delay (Ready Mode)	—	—	100	—	—	80	ns
99	TdC (W/RWZ)	Clock ↓ to $\overline{W/RDY}$ float delay (Wait mode)	—	—	110	—	—	90	ns
100	TwPh	Pulse Width (High)	200	—	—	200	—	—	ns
101	TwPl	Pulse Width (Low)	200	—	—	200	—	—	ns
102	TcTxC	$\overline{TxC}$ cycle time	330	—	∞	250	—	∞	ns
103	TwTxCl	$\overline{TxC}$ Width (Low)	100	—	∞	80	—	∞	ns
104	TwTxCh	$\overline{TxC}$ Width (High)	100	—	∞	80	—	∞	ns
105	TdTxC (TxD)	$\overline{TxC}$ ↓ to TxD delay (x1 mode)	—	—	220	—	—	180	ns
106	TdTxC (W/RRf)	$\overline{TxC}$ ↓ to $\overline{W/RDY}$ ↓ delay (Ready mode)	5	—	9	5	—	9	CLK Periods
107	TcRxC	$\overline{RxC}$ cycle time	330	—	∞	250	—	∞	ns
108	TwRxCl	$\overline{RxC}$ Width (Low)	100	—	∞	80	—	∞	ns
109	TwRxCh	$\overline{RxC}$ Width (High)	100	—	∞	80	—	∞	ns
110	TsRxD (Rxc)	$\overline{RxD}$ to $\overline{RxC}$ ↑ Setup time (x1 mode)	0	—	—	10	—	—	ns
111	ThRxD (Rxc)	$\overline{RxC}$ ↑ to RxD hold time (x1 mode)	100	—	—	80	—	—	ns
112	TdRxC (W / RRf)	$\overline{RxC}$ ↑ to $\overline{W/RDY}$ ↓ delay (Ready mode)	10	—	13	10	—	13	CLK Periods

030489

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
113	TdRxC (SYNC)	$\overline{RxC}$ ↑ to $\overline{SYNC}$ ↓ delay (Output mode)	4	—	7	4	—	7	CLK Periods
114	TsSYNC (Rxc)	$\overline{SYNC}$ ↓ to $\overline{RxC}$ ↑ Setup (External SYNC modes)	-100	—	—	-100	—	—	ns
115	IsTxc	$\overline{TxC}$ ↓ Setup to TL ↑ for detection of interrupt	5*T1 +T48	—	9*T1 +T48	5*T1 +T48	—	9*T1 +T48	ns
116	IsRxc	$\overline{RxC}$ ↑ Setup to TL ↑ for detection of interrupt	10*T1 +T48	—	13*T1 +T48	10*T1 +T48	—	13*T1 +T48	ns

030489

4.3.6 AC Characteristic of WDT (in Active State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			TMPZ84C015AF-8 (8MHz)			UNIT
			MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	
117	Tdc (WDTf)	Clock ↑ to $\overline{WDTOUT}$ ↓ Delay	—	—	160	—	—	120	ns
118	Tdc (WDTr)	Clock ↑ to $\overline{WDTOUT}$ ↑ Delay	—	—	165	—	—	125	ns
119	TcWDT	$\overline{WDTOUT}$ Out put period							
		WDT Mode 0	—	$T1*2^{16}$	—	—	$T1*2^{16}$	—	ns
		WDT Mode 1	—	$T1*2^{18}$	—	—	$T1*2^{18}$	—	ns
		WDT Mode 2	—	$T1*2^{20}$	—	—	$T1*2^{20}$	—	ns
		WDT Mode 3	—	$T1*2^{22}$	—	—	$T1*2^{22}$	—	ns

030489

Note 1 : Timing Measurements are made at the following voltage.

Input VIH = 2.4V, VIL = 0.4V, VIHC = VCC − 0.6V, VILC = 0.6V

Output VOH = 2.2V, VOL = 0.8V (Exept CLKOUT)

CL = 100pF

4.4 AC TIMING CHARTS (1) (in Active State)

4.4.1 AC Timing Charts of MPU (in Active State)

Figures 4.4.1 through 4.4.8 show the basic timing charts. The circled numbers in these charts correspond to the numbers in the number column of the AC Electrical Characteristics Tables.

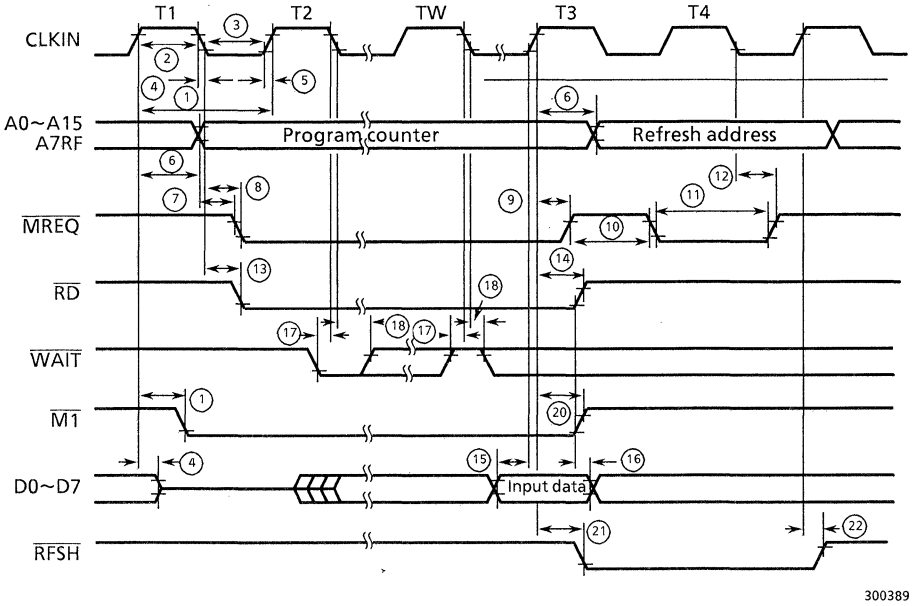


Figure 4.4.1 Opcode Fetch Cycle

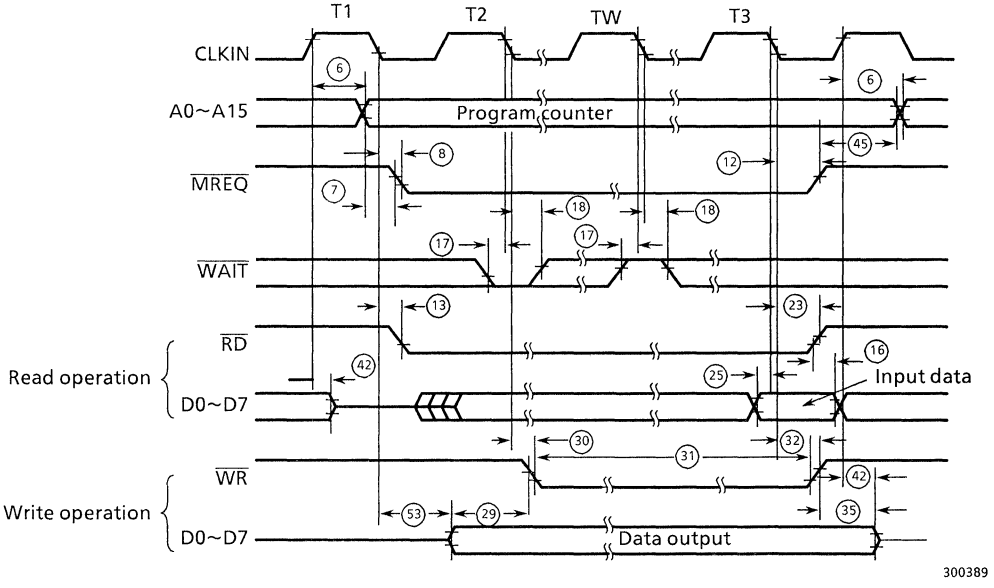
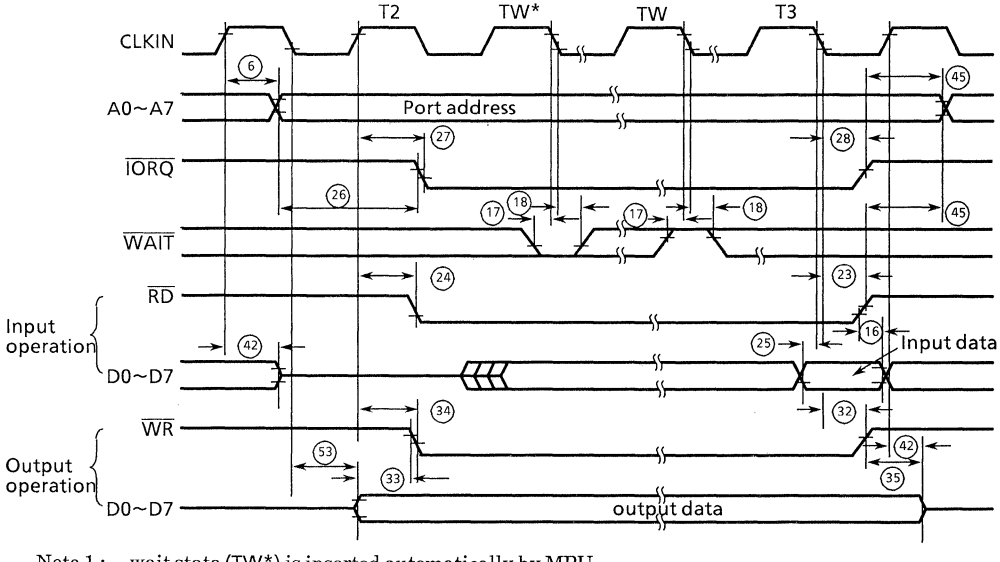
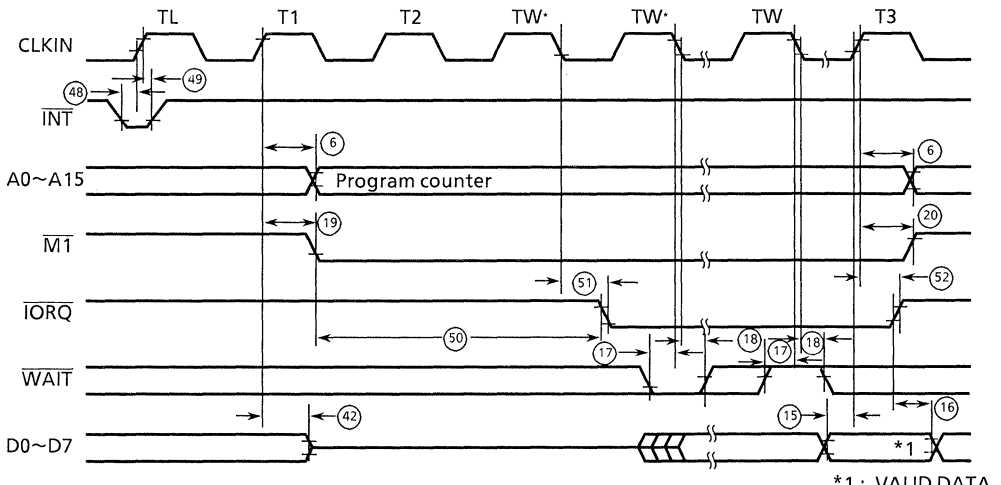


Figure 4.4.2 Memory Read / Write Cycle



Note 1 : wait state (TW*) is inserted automatically by MPU.

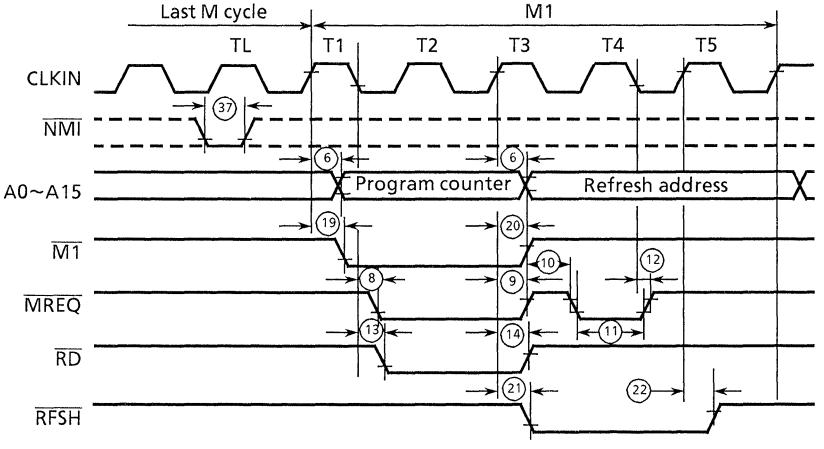
Figure 4.4.3 I / O Cycle



Notes 1. TL : Last state of instruction
2. 2 - wait state (TW*) is inserted automatically by MPU

300389

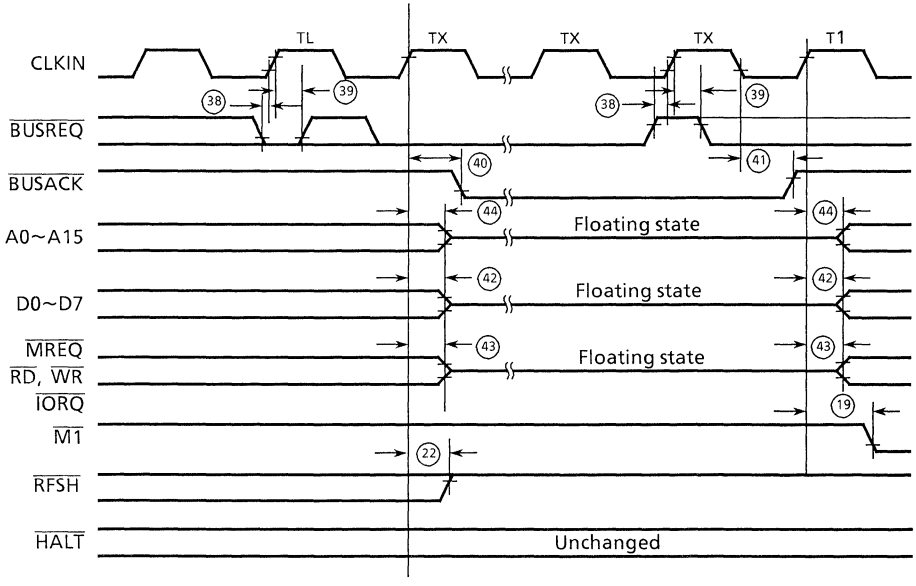
Figure 4.4.4 Interrupt Request/Acknowledge Cycle



Note : $\overline{NMI}$ is asynchronous, but its falling edge signal must occur synchronously with the rising edge of previous TL state for correct response to the subsequent machine cycles.

300389

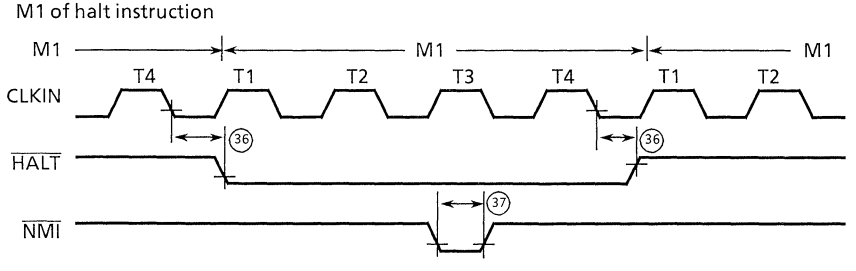
Figure 4.4.5 Non-maskable Interrupt Request Cycle



Notes 1. TL: Last state of a given machine cycle
2. Tx: Clock used by peripheral LSI that made request.

300389

Figure 4.4.6 Bus Request / Acknowlege Cycle



Note: $\overline{INT}$ signal is also used for releasing HALT state.

300389

Figure 4.4.7 HALT Acknowledge Cycle

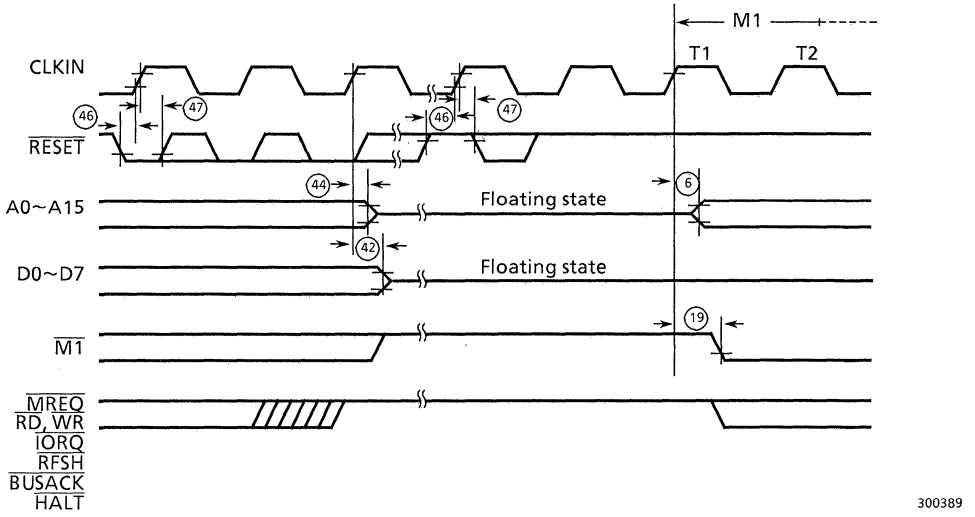


Figure 4.4.8 Reset Cycle

4.4.2 AC Timing Charts of CGC (in Active State)

The following Figures show the timings in each operation mode with the CLKOUT pin connected to the CLKIN pin.

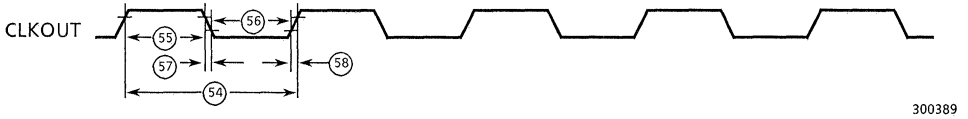


Figure 4.4.9 CLKOUT Waveform

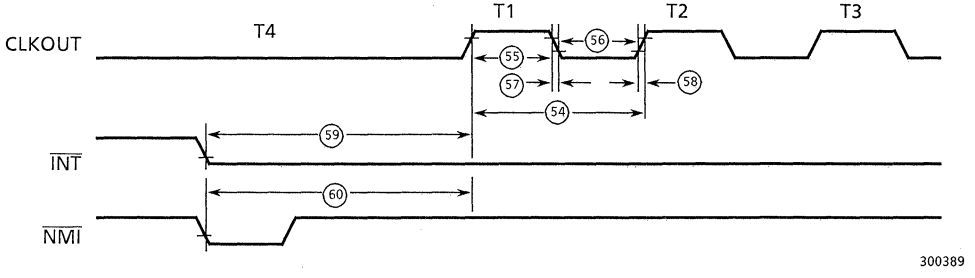


Figure 4.4.10 Clock Restart Timing (STOP Mode)

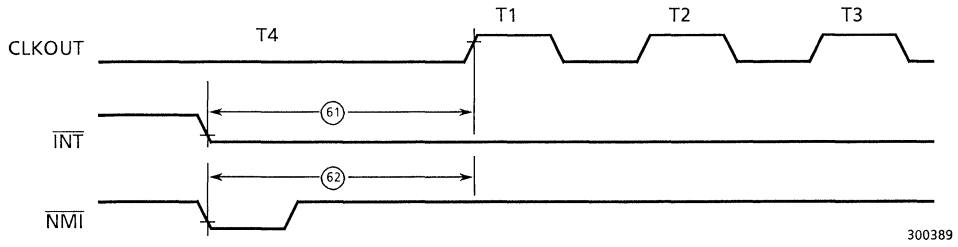


Figure 4.4.11 Clock Restart Timing (IDLE1 Mode)

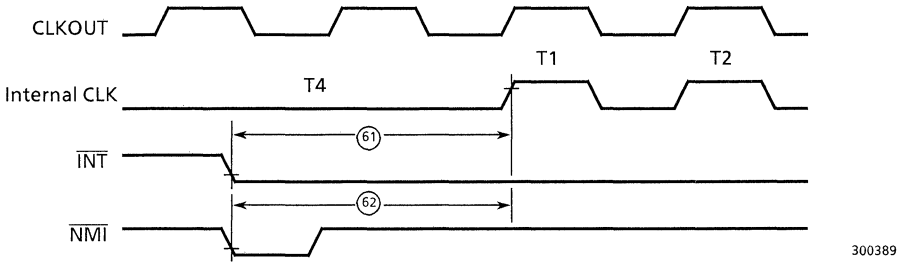


Figure 4.4.12 Clock Restart Timing (IDLE2 Mode)

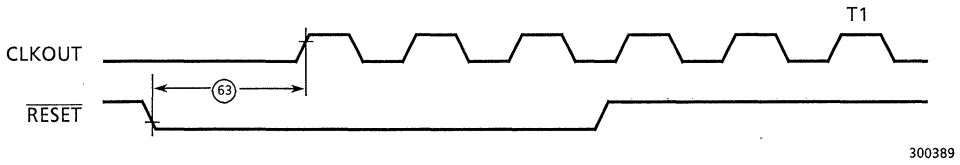


Figure 4.4.13 Timing of Clock Start by $\overline{\text{RESET}}$ (IDLE1 and IDLE2 Modes)

4.4.3 AC Timing Charts of CTC (in Active State)

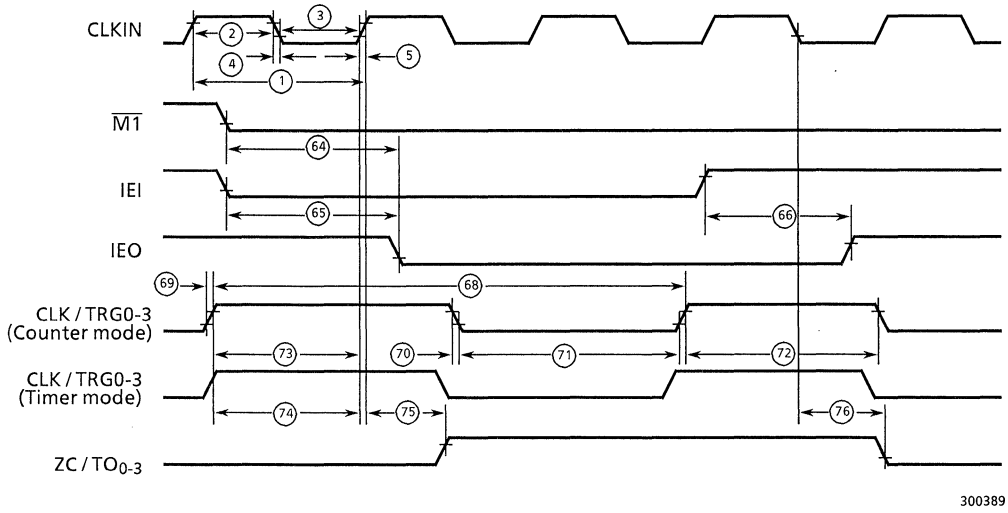


Figure 4.4.14 CTC Timing Diagram

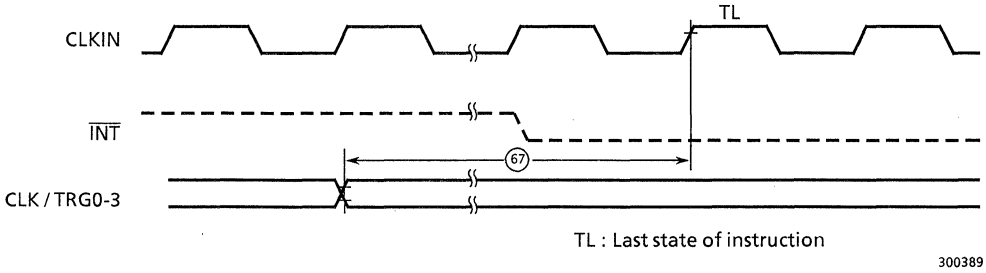


Figure 4.4.15 CTC Interrupt Occurrence Timing

4.4.4 AC Timing Charts of PIO (in Active State)

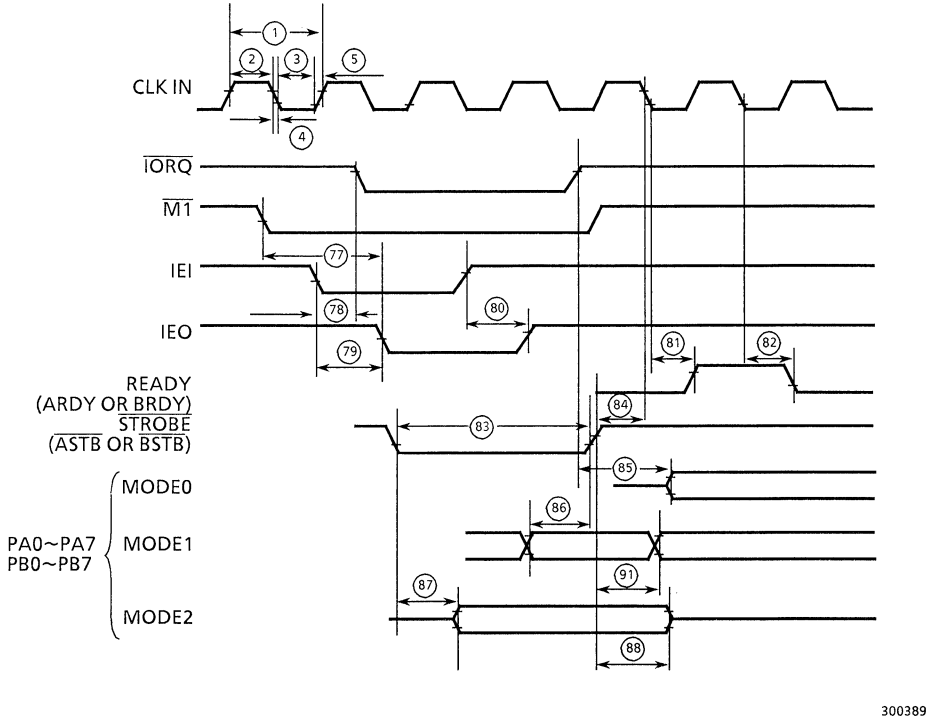


Figure 4.4.16 PIO Timing Diagram

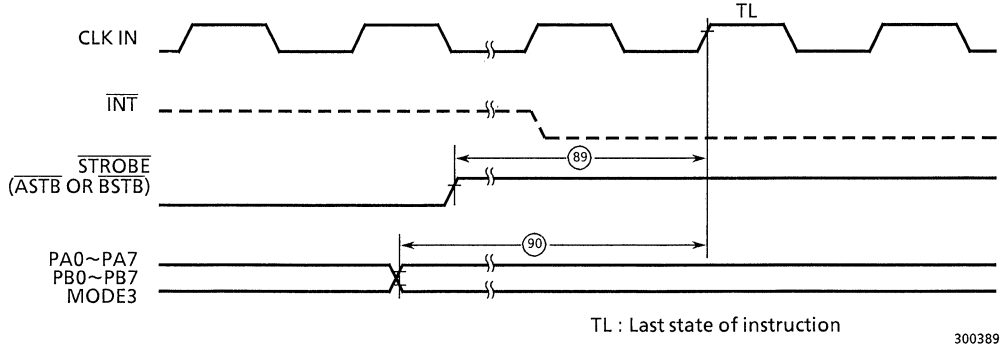


Figure 4.4.17 PIO Interrupt Occurrence Timing

4.4.5 AC Timing Charts of SIO (in Active State)

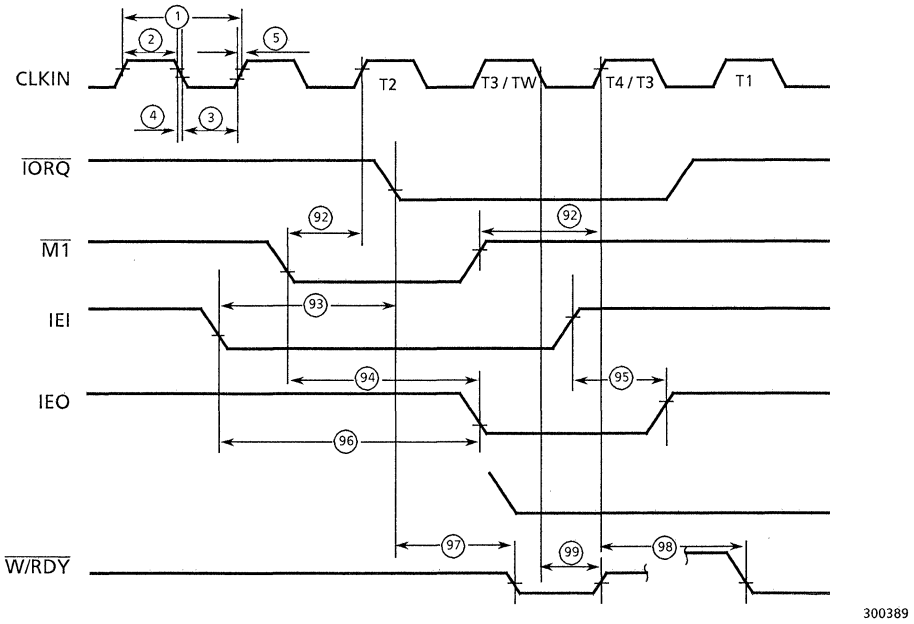


Figure 4.4.18 (a) SIO Timing Diagram

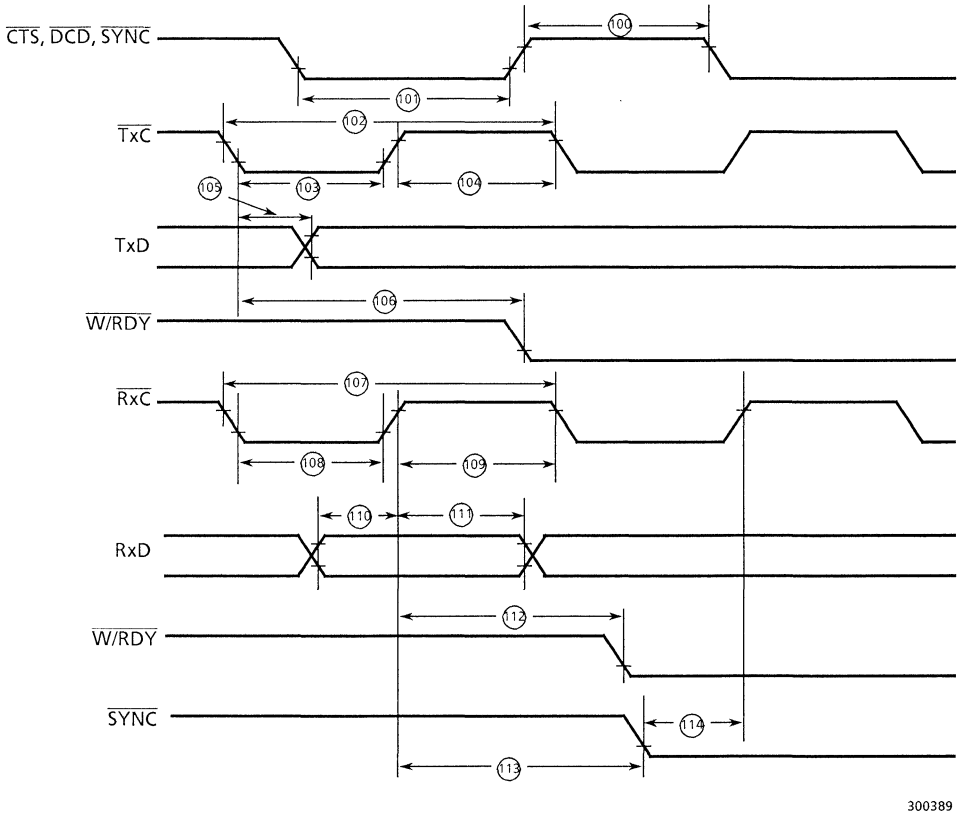


Figure 4.4.18 (b) SIO Timing Diagram

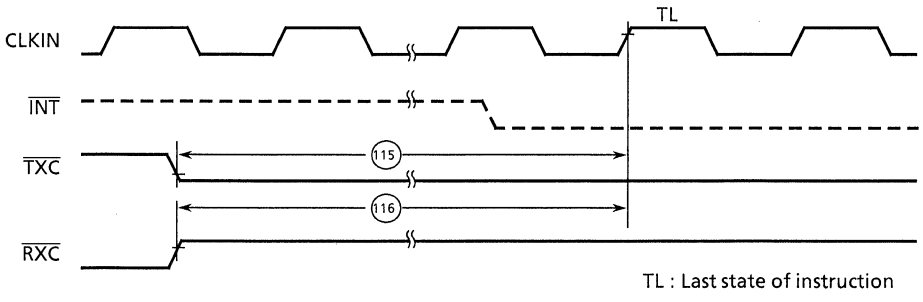


Figure 4.4.19 SIO Interrupt Occurrence Timing

4.4.6 AC Timing Charts of WDT (in Active State)

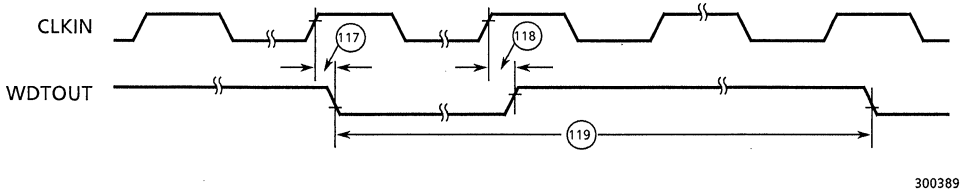


Figure 4.4.20 WDT Timing Diagram

4.5 AC ELECTRICAL CHARACTERISTICS (2) (in Inactive State)

6MHz VERSION : TA = −40°C~+85°C, VCC = 5V+10%, VSS = 0V

4.5.1 AC Characteristics of CGC (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
1	TcCCLK	Output Clock Cycle	—	162	—	ns
2	TwChCLK	Output Clock Width (High)	—	70	—	ns
3	TwClCLK	Output Clock Width (Low)	—	70	—	ns
4	TfCCLK	Output Clock fall Time	—	12	—	ns
5	TrCCLK	Output Clock rise Time	—	12		ns
6	TRST (INT) S	Clock (CLKOUT) restart Time by $\overline{INT}$ (STOP mode)	—	2^{14} + 2.5TcC	—	ns
7	TRST (NMI) S	Clock (CLKOUT) restart Time by $\overline{NMI}$ (STOP mode)	—	2^{14} + 2.5TcC	—	ns
8	TRST (INT) I	Clock (CLKOUT) restart Time by $\overline{INT}$ (IDLE 1 / 2 mode)	—	2.5 *TcC	—	ns
9	TRST (NMI) I	Clock (CLKOUT) restart Time by $\overline{NMI}$ (IDLE 1 / 2 mode)	—	2.5 *TcC	—	ns
10	TRST (RESET) I	Clock (CLKOUT) restart Time by $\overline{RESET}$ (IDLE 1 / 2 mode)	—	1TcC	—	ns
11	TsHALT (M1r)	$\overline{HALT}$ Set up Time	10	—	—	ns

300389

4.5.2 AC Characteristics of CTC (in Inactive State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
12	TcC	ClocK Cycle Time	162	—	—	ns
13	Twch	ClocK width (High)	65	—	—	ns
14	Twcl	ClocK width (Low)	65	—	—	ns
15	TfC	ClocK falling time	—		20	ns
16	TrC	ClocK rising time	—	—	20	ns
17	Th	Hold Time	10	—	—	ns
18	TcCS (C)	CS (A1, A0) Set up time to clock ↑	100	—	—	ns
19	TSCE (C)	CE (A7~A2) Set up time to clock ↑	150	—	—	ns
20	TsIO (C)	$\overline{IORQ}$ ↓ Set up time to clock ↑	70	—	—	ns

300389

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
21	TsRD (C)	$\overline{RD}$ ↓ Set up time to clock ↑	70	—	—	ns
22	TdC (DO)	Clock ↑ to Data Valid Delay	—	—	130	ns
23	TdC (DOz)	$\overline{IORQ}$, $\overline{RD}$ ↑ to Data Float Delay	—	—	90	ns
24	TdCr (M1f)	Data Input Set up time to clock ↑	40	—	—	ns
25	TsM1 (C)	$\overline{M1}$ Set up time to clock ↑	70	—	—	ns
26	TdM1 (IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (in case of generating only interrupt immediately before $\overline{M1}$ cycle)	—	—	130	ns
27	TdIO (DOI)	$\overline{IORQ}$ ↓ to Data out Delay (INTA Cycle)	—	—	110	ns
28	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	ns
29	TdIEI (IEOf)	IEI ↑ to IEO ↑ Delay (after ED decode)	—	—	150	ns
30	TdC (INT)	Clock ↑ to $\overline{INT}$ ↓ Delay	—	—	TcC + 120	ns
31	TdA (IORQf)	CLK / TRG ↑ to $\overline{INT}$ ↓ Delay TsCTR (c) Satisfied TsCTR (c) not Satisfied	— —	TcC + 120 + T37 2TcC + 120 + T37	— —	ns
32	TcCTR	CLK / TRG Frequency	—	2TcC	—	ns
33	TrCTR	CLK / TRG rising time	—	—	40	ns
34	TfCTR	CLK / TRG falling time	—	—	40	ns
35	TwCTRl	CLK / TRG pulse width (Low)	120	—	—	ns
36	TwCTRh	CLK / TRG pulse width (High)	120	—	—	ns
37	TsCTR (CS)	CLK/TKG ↑ to clock ↑ Setup Time for Immediate Count (counter mode)	150	—	—	ns
38	TsCTR (CT)	CLK/TRG ↑ to clock ↑ Setup Time for enabliling of Prescaler on following clock ↑ (timer mode)	150	—	—	ns
39	TdC (ZC / TOr)	Clock ↑ to ZC/TO ↑ Delay	—	—	140	ns
40	TdC (ZC / TOf)	Clock ↓ to ZC/TO ↓ Delay	—	—	140	ns

200589

4.5.3 AC Characteristics of PIO (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
41	TsCS (RI)	$\overline{CE}(A_7$ to $A_2)$, $B/\overline{A}(A_1)$, $C/\overline{D}(A_0)$ Set up Time to $\overline{RD}$, $\overline{IORQ}$	50	—	—	ns
42	Th	Hold Time	40	—	—	ns
43	TsRI (C)	$\overline{RD}$, $\overline{IORQ}$ Set up time to clock ↑	70	—	—	ns
44	TdRI (DO)	$\overline{RD}$, $\overline{IORQ}$ ↓ to Data out Delay	—	—	300	ns
45	TdRI (DOs)	$\overline{RD}$, $\overline{IORQ}$ ↑ to Data float Delay	—	—	70	ns
46	TsDI (C)	Data Set up time to clock ↑	40	—	—	ns
47	TdIO (DOI)	$\overline{IORQ}$ ↓ to Data out Delay (INTA Cycle)	—	—	120	ns
48	TsM1 (Cr)	$\overline{M1}$ ↓ Set up time to clock ↑	70	—	—	ns
49	TsM1 (Cf)	$\overline{M1}$ ↑ Set up time to clock ↓ (M1 cycle)	0	—	—	ns
50	TdM1 (IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (Interrupt Immedi-ately receding $\overline{M1}$ ↓)	—	—	100	ns
51	TsIEI (IO)	IEI Set up time to $\overline{IORQ}$ ↓ (INTA Cycle)	100	—	—	ns
52	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	ns
53	TdIEI (IE0r)	IEI ↑ to IEO ↑ Delay (after ED Decade)	—	—	150	ns
54	TdIO (C)	$\overline{IORQ}$ ↑ Set up time to Clock ↓	170	—	—	ns
55	TdC (RDYr)	Clock ↓ to READY ↑ Delay	—	—	170	ns
56	TdC (RDYf)	Clock ↓ to READY ↓ Delay	—	—	120	ns
57	TwSTB (C)	$\overline{STROBE}$ Pulse width	120	—	—	ns
58	TsSTB (C)	Set up time of $\overline{STROBE}$ rise for clock fall (in case of making READY to active by next cycle)	150	—	—	ns
59	TdIO (PD)	Delay from $\overline{IORQ}$ rise to Port data stable (Mode 0)	—	—	160	ns
60	TsPD (STB)	Port data set up time for $\overline{STROBE}$ rise (Mode 1)	190	—	—	ns
61	TdSTB (PD)	Output Port data delay time from $\overline{STROBE}$ fall (Mode 2)	—	—	180	ns
62	TdSTB (PDr)	Delay from $\overline{STROBE}$ rise to Port data float (Mode 2)	—	—	160	ns
63	TdPD (INT)	Delay from port data match to $\overline{INT}$ fall (Mode 3)	—	—	430	ns
64	TdSTB (INT)	Delay from $\overline{STROBE}$ rise to $\overline{INT}$ fall	—	—	350	ns

300389

4.5.4 AC Characteristics of SIO (in Inactive State)

(1/2)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
65	TsCS (C)	$\overline{CE}$ (A7 to A2), C / $\overline{D}$ (A0), B / $\overline{A}$ (A1) Set up time to clock ↑	60	—	—	ns
66	TsRD (C)	$\overline{IORQ}$, $\overline{RD}$ Set up Time to clock ↑	60	—	—	ns
67	TdC (DO)	Clock ↑ to Data output Delay	—	—	150	ns
68	TsDI (C)	Data input set up time to clock ↑ (write cycle or $\overline{M1}$ cycle)	30	—	—	ns
69	TdRD (DOz)	$\overline{RD}$ ↑ to Data Out Float Delay	—	—	90	ns
70	TdIO (DOI)	$\overline{IORQ}$ ↓ to Data out put Delay (INTACK cycle)	—	—	120	ns
71	TsM1 (C)	$\overline{M1}$ Set up time to clock ↑	75	—	—	ns
72	TsIEI (IO)	$\overline{IEI}$ Set up time to $\overline{IORQ}$ ↓ (INTACK Cycle)	120	—	—	ns
73	TdM1 (IEO)	$\overline{M1}$ ↓ to IEO ↓ Delay (interrupt before $\overline{M1}$)	—	—	160	ns
74	TdIEI (IEOr)	IEI ↑ to IEO ↑ Delay (After ED Decode)	—	—	150	ns
75	TdIEI (IEOf)	IEI ↓ to IEO ↓ Delay	—	—	70	ns
76	TdC (INT)	Clock ↑ to $\overline{INT}$ ↓ Delay	—	—	150	ns
77	TdIO (W / RWf)	$\overline{IORQ}$, $\overline{CE}$ (A7 to A2) ↓ to $\overline{W/RDY}$ ↓ Delay (Wait Mode)	—	—	175	ns
78	TdC (W / RRf)	Clock ↑ to $\overline{W/RDY}$ ↓ Delay (Ready Mode)	—	—	100	ns
79	TdC (W / RWz)	Clock ↓ to $\overline{W/RDY}$ Float Delay (Wait Mode)	—	—	110	ns
80	Th, Th (CS)	Any unspecified hold When set up is specified	0	—	—	ns
81	TwPh	Pulse width (High)	200	—	—	ns
82	TwPl	Pulse width (Low)	200	—	—	ns
83	TcTxC	$\overline{TxC}$ Cycle time	330	—	∞	ns
84	TwTxCl	$\overline{TxC}$ width (Low)	100	—	∞	ns
85	TwTxCh	$\overline{TxC}$ width (High)	100	—	∞	ns
86	TdTxC (TxD)	$\overline{TxC}$ ↓ to TxD Delay (x1 Mode)	—	—	220	ns
87	TdTxC (W / RRf)	$\overline{TxC}$ ↓ to $\overline{W/RDY}$ ↓ Delay (Ready mode)	5	—	9	CLK Periods
88	TdTxC (INT)	$\overline{TxC}$ ↓ to $\overline{INT}$ ↓ Delay	5	—	9	CLK Periods
89	TcRxC	$\overline{RxC}$ Cycle time	330	—	∞	ns

300389

(2/2)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
90	TwRxCl	$\overline{RxC}$ width (Low)	100	—	∞	ns
91	TwRxCh	$\overline{RxC}$ width (High)	100	—	∞	ns
92	TsRxD (RxC)	RxD to $\overline{RxC}$ ↑ Set up time (x1 mode)	0	—	—	ns
93	ThRxD (RxC)	$\overline{RxC}$ ↑ to RxD Hold time (x1 mode)	100	—	—	ns
94	TdRxC (W / RRf)	$\overline{RxC}$ ↑ to $\overline{W/RDY}$ ↓ Delay (Ready mode)	10	—	13	CLK Periods
95	TdRxC (INT)	$\overline{RxC}$ ↑ to $\overline{INT}$ ↓ Delay	10	—	13	CLK Periods
96	TdRxC (SYNC)	$\overline{RxC}$ ↑ to $\overline{SYNC}$ ↓ Delay (output modes)	4	—	7	CLK Periods
97	TsSYNC (RxC)	$\overline{SYNC}$ ↓ to $\overline{RxC}$ ↑ Set up Time (External sync modes)	-100	—	—	ns
98	TsAdd (Cr)	Address Set up time to clock ↑	150	—	—	ns
99	TsIO (Cr)	$\overline{IORQ}$ ↓ Set up time to clock ↑	70	—	—	ns
100	TdRD (Cr)	$\overline{RD}$ ↓ Set up time to clock ↑	70	—	—	ns
101	TdCr (Do)	Data out Delay to clock ↑	—	—	130	ns
102	TdIORDr (DoZ)	Data Float Delay to $\overline{IORQ}$ ↑ , $\overline{RD}$ ↑	—	—	90	ns
103	TsWR (Cr)	$\overline{WR}$ ↓ Set up time to clock ↑	70	—	—	ns
104	TsDI (Cr)	Data Input Set up time to clock ↑	0	—	—	ns
105	TdIOWRf (D)	Data Hold time to $\overline{IORQ}$ ↑ , $\overline{WR}$ ↑	20	—	—	ns

300389

4.5.5. AC Characteristics of WDT (in Inactive State)

NO.	SYMBOL	PARAMETER	TMPZ84C015AF-6 (6MHz)			UNIT
			MIN.	TYP.	MAX.	
106	TdC (WDTf)	Clock ↑ to $\overline{\text{WDTOUT}}$ ↓ Delay	—	—	160	ns
107	TdC (WDTr)	Clock ↑ to $\overline{\text{WDTOUT}}$ ↑ Delay	—	—	165	ns
108	TCWDT	$\overline{\text{WDTOUT}}$ out put period				
		WDT Mode 0	—	$T12_*$ 2^{16}	—	ns
		WDT Mode 1	—	$T12_*$ 2^{18}	—	ns
		WDT Mode 2	—	$T12_*$ 2^{20}	—	ns
		WDT Mode 3	—	$T12_*$ 2^{22}	—	ns

300389

Note1 : Timing Measurements are made at the following voltage.

Input	VIH = 2.4V,	VIL = 0.4V
	VIHC = VCC − 0.6V,	VILC = 0.6V
Output	VOH = 2.2V,	VOL = 0.8V (Except CLKOUT)
	CL = 100pF	

4.6 AC TIMING CHARTS (2) (IN INACTIVE STATE)

4.6.1 AC Timing Charts of CGC (in Inactive State)

The following Figures show the timing charts in each operation mode with the CLKOUT pin connected to the CLKIN pin.

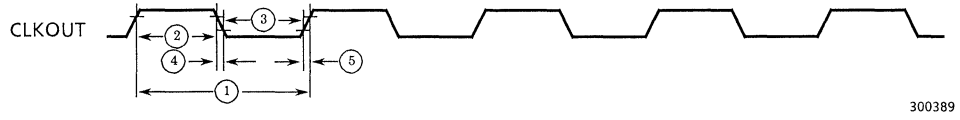


Figure 4.6.1 CLKOUT waveform

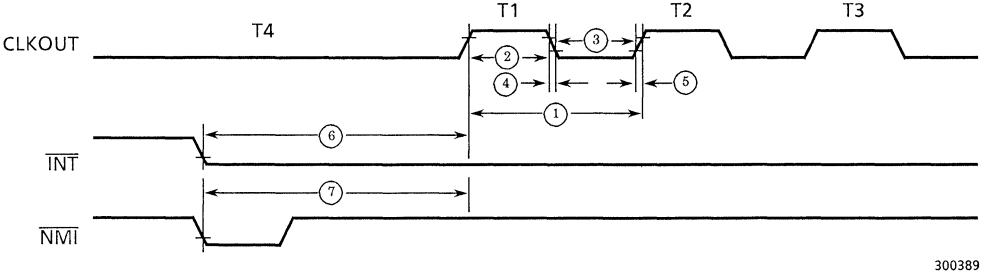


Figure 4.6.2 Clock restart timing (STOP mode)

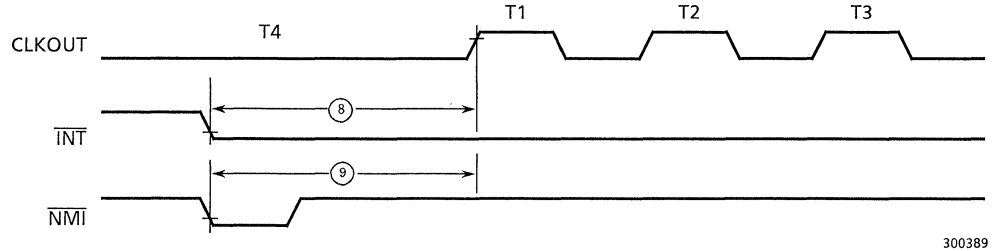


Figure 4.6.3 Clock restart timing (IDLE1 mode)

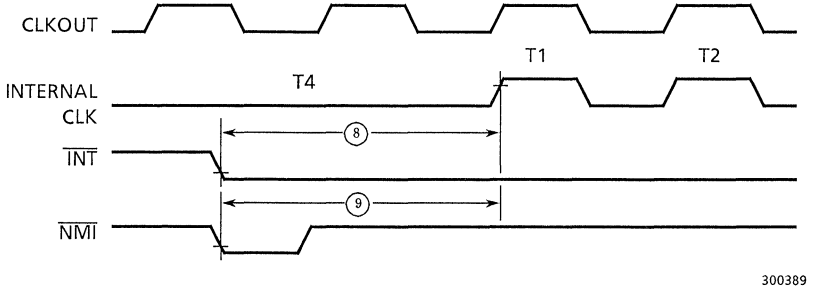


Figure 4.6.4 Clock restart timing (IDLE2 mode)

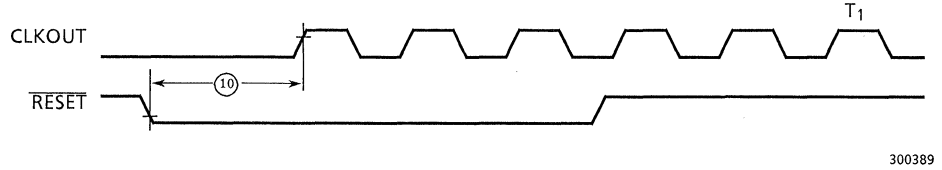


Figure 4.6.5 Timing of clock restart by $\overline{\text{RESET}}$ (IDLE1, IDLE2 mode)

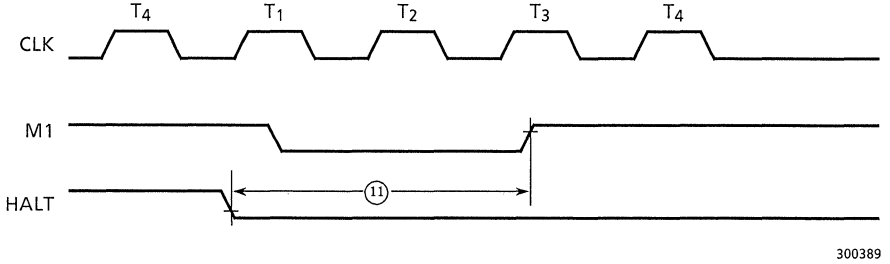


Figure 4.6.6 Clock suspension timing (IDLE1,IDLE2 and STOP modes)

4.6.2 AC Timing Charts of CTC (in Inactive State)

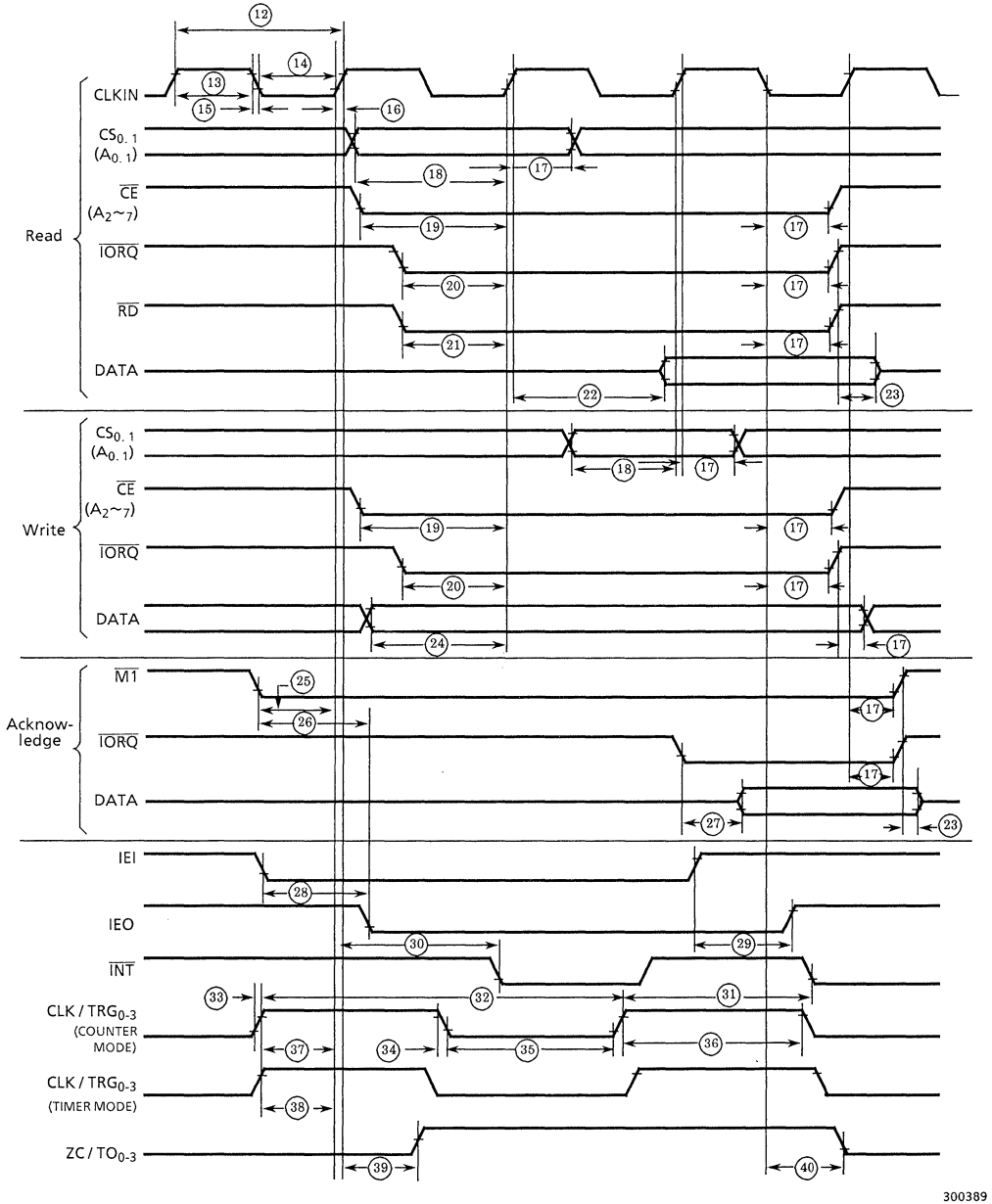


Figure 4.6.7 CTC timing diagram (Inactive)

4.6.3 AC Timing Charts of PIO (in Inactive State)

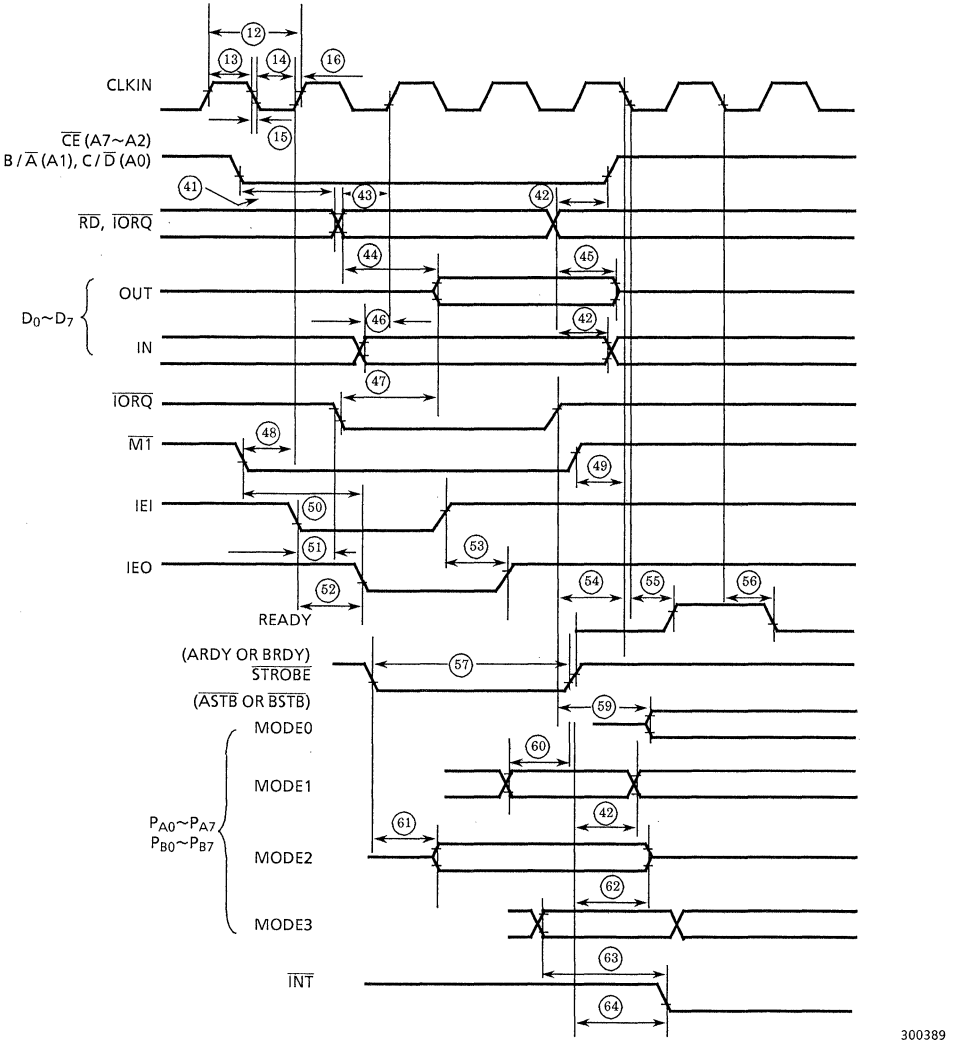


Figure 4.6.8 PIO timing diagram (Inactive)

4.6.4 AC Timing Charts of SIO (in Inactive State)

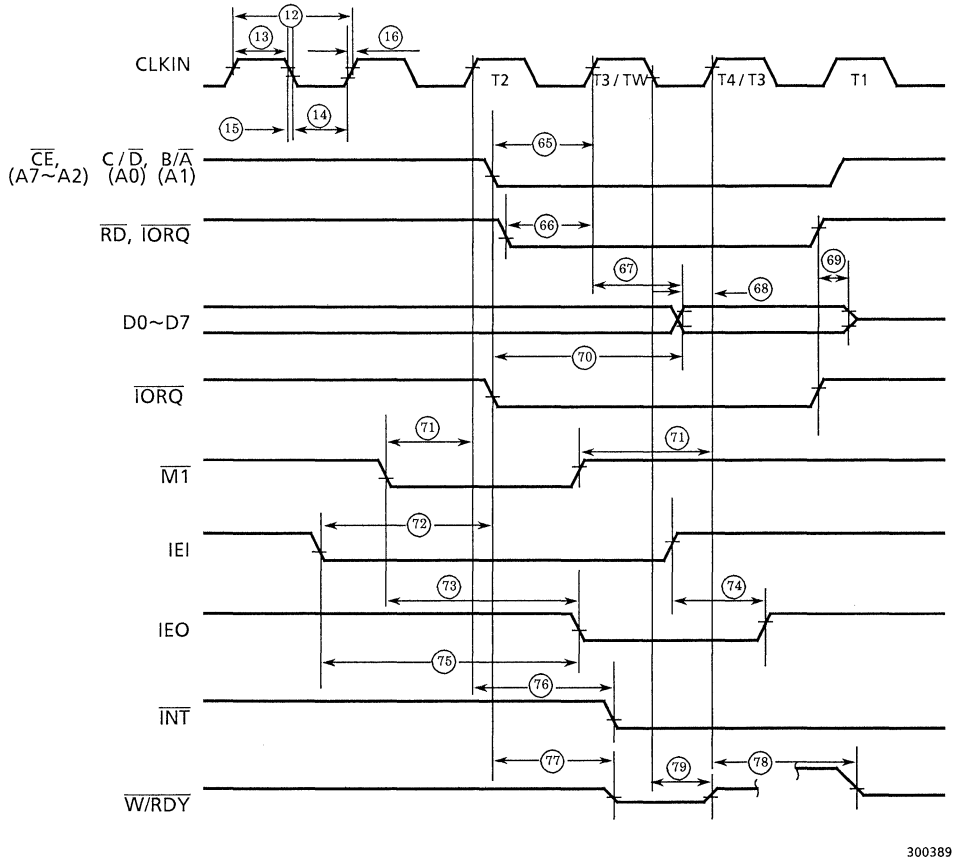


Figure 4.6.9 SIO timing diagram (a) (Inactive)

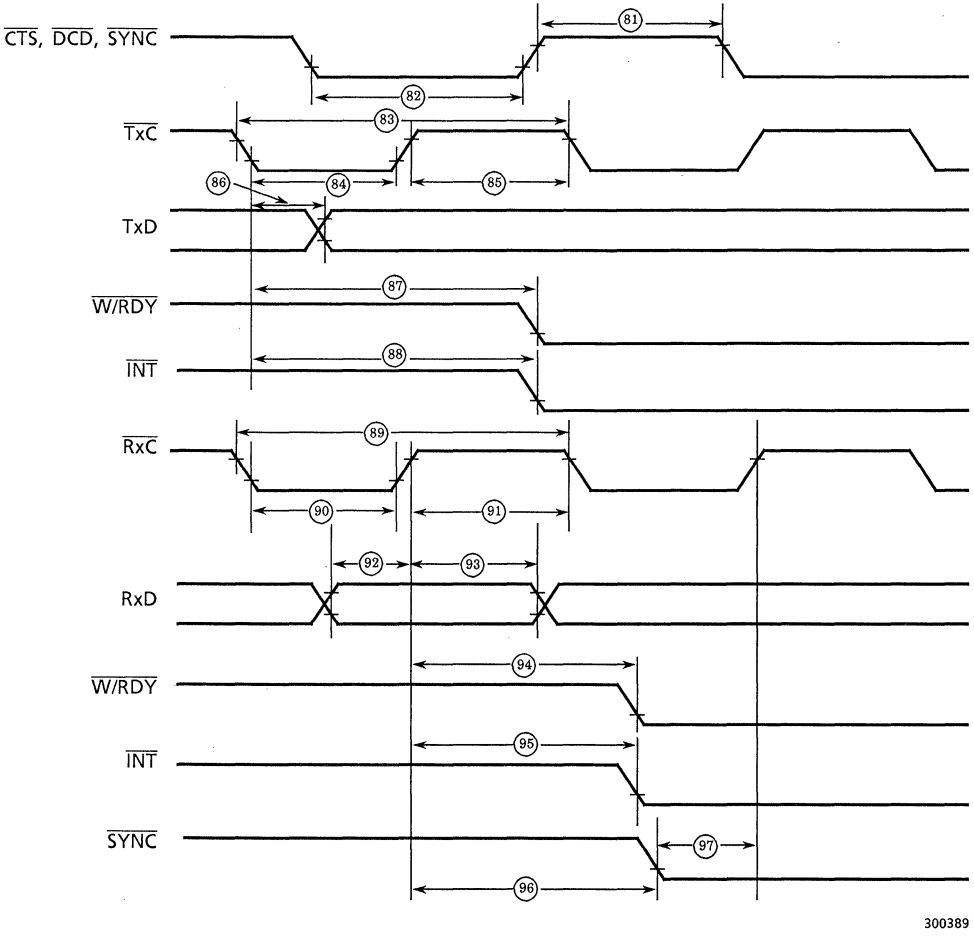


Figure 4.6.9 SIO timing diagram (b) (Inactive)

4.6.5 AC Timing Charts of WDT (in Inactive State)

(The mode setting and daisy chain interrupt setting registers on WDT)

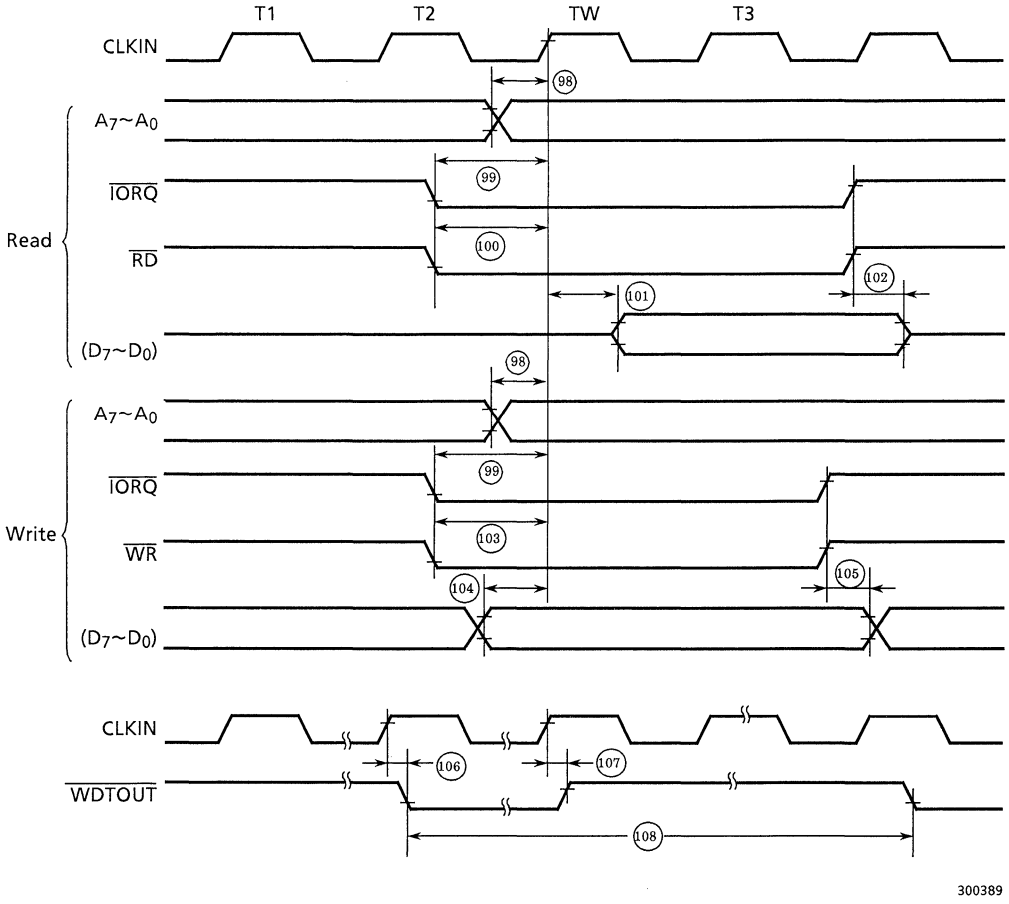


Figure 4.6.10 RD/WRITE, $\overline{\text{WDTOUT}}$ timing Diagram

4.7 PIN CAPACITANCE

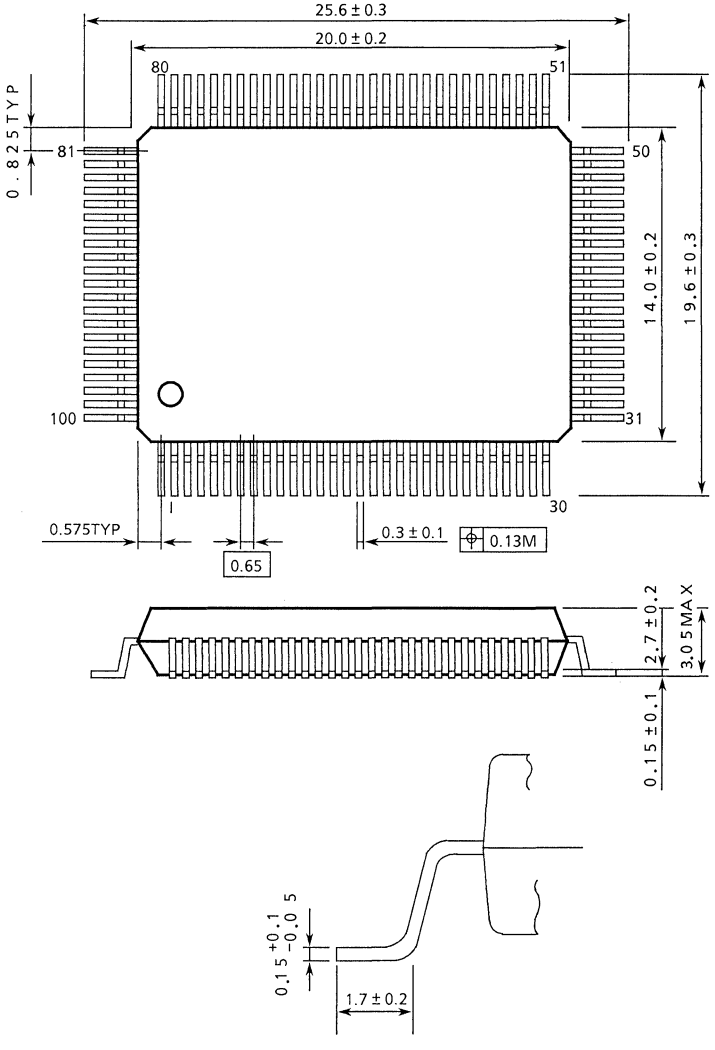
SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CLOCK	Clock Input Capacitance	F = 1MHz All terminals except that to be measured be earthed	–	–	8	PF
CIN	Input Capacitance		–	–	6	
COUT	Output Capacitance		–	–	10	

300389

5. EXETERNAL DIMENSIONS

QFP100-P-1420B

Unit : mm



290589

The Guide Of TOSHIBA TLCS-Z80 ASSP Development Tools And Development Methods

This guide shows how to develop the ASSP application systems with development tools.

1. GENERAL

Toshiba ASSP series are high-performance CMOS Z80 micro-processors, in which they include counter/timer circuit (CTC), clock generator/controller (CGC), input/output port (PIO, SIO), and watchdog timer. ASSP series are applied to the communication systems, controllers, and measurement instruments.

In developing, these processors work as the evaluation chips. Thus development tools for Z80 MPU (Toshiba's real time emulator : RTE-80 or others) can be used, and new tools for these processors are not required.

2. EVALUATION MODE OF ASSP

EV pin is provided to disconnect the MPU from the ASSP processor. Normally this EV pin is low level.

In developing, after one machine code has executed the MPU of the ASSP has disconnected from the target system by EV pin input high and $\overline{\text{BUSREQ}}$ pin input low.

Then an emulator executes programs, and operates peripherals in the ASSP processor.

Table 2.1 shows the pins of the MPU which are disconnected by EV and $\overline{\text{BUSREQ}}$.

Figure 2.1 shows the timing of disconnection.

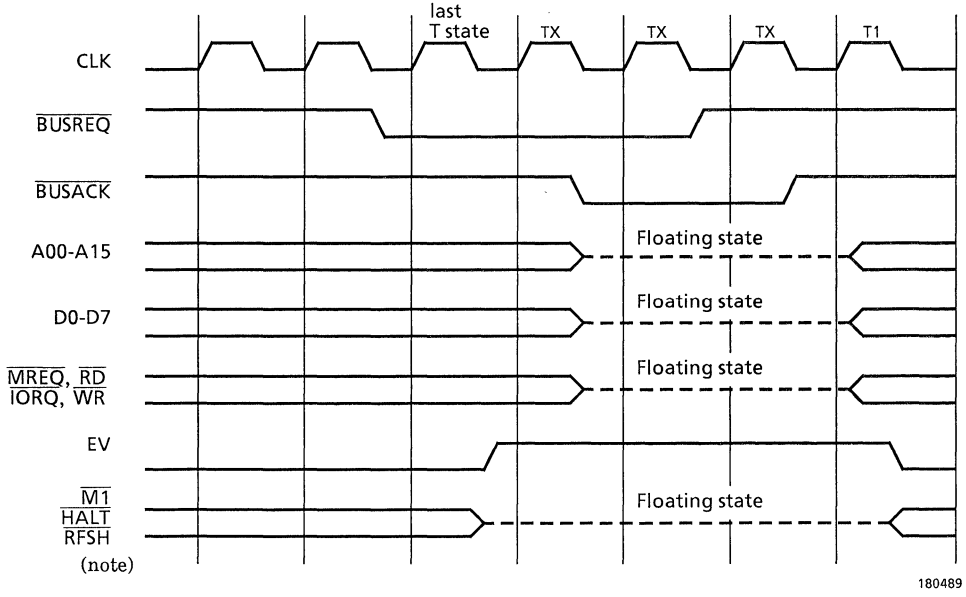
Table 2.1 Pins to be disconnected

mode set pin	pin to be disconnected
EV	$\overline{\text{M1}}$, $\overline{\text{HALT}}$, $\overline{\text{RFSH}}$ (note)
$\overline{\text{BUSREQ}}$	$\overline{\text{WR}}$, $\overline{\text{MREQ}}$, $\overline{\text{IORQ}}$ A00 – A15, D0 – D7, $\overline{\text{RD}}$

180489

Note : $\overline{\text{BUSACK}}$ is not disconnected

: For the TMPZ84C011A $\overline{\text{RFSH}}$ pin is not disconnected and goes "High".



180489

Note : Using TMPZ84C011A, $\overline{\text{RFSH}}$ pin is not placed in the floating state and outputs high level.

Figure 2.1 Timing of Disconnection

3. EMULATOR CONNECTION AND ADAPTER BOARDS

When developing the systems using ASSP processor, an evaluation board interfacing between the target board and real time emulator is required.

There are two ways to develop the ASSP systems (Level 1 and Level 2), and the evaluation boards for both ways are available.

3.1 LEVEL 1 (Evaluation boards : BM8024/25)

This level is adopted when the pattern for the ASSP is not completed on the target board or the target board is pre-production.

Remove ASSP chip from the target board, then connect the all signal lines of the ASSP to the evaluation board. An ASSP chip on the evaluation board is working in evaluation mode. (MPU in the chip is disconnected. Therefore emulator operates the peripherals in ASSP.)

On the evaluation board there is an universal area to compose target system.

Evaluation board BM8024 for TMPZ84C011A
BM8025 for TMPZ84C015A

3.2 LEVEL 2 (Adapter boards : BM8026/27)

In this case the ASSP chip on the target board works in evaluation mode. This level is adopted to debug final target board.

Adapter board BM8026 for TMPZ84C011A
BM8027 for TMPZ84C015A

4. DEVELOPMENT METHOD

4.1 LEVEL 1

Figure 4.1 shows the way to develop the target system with evaluation board and RTE80 (Toshiba's real time emulator).

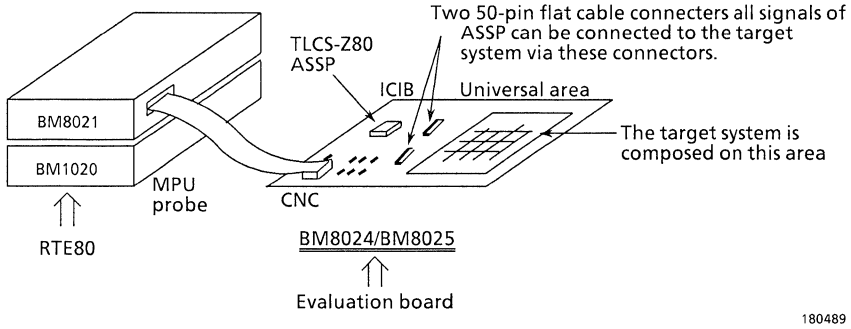


Figure 4.1 Example of connection (Level 1)
(For the detail, refer to instruction manuals)

4.2 LEVEL 2

Figure 4.2 shows the way to develop the target system with adapter board and RTE80.

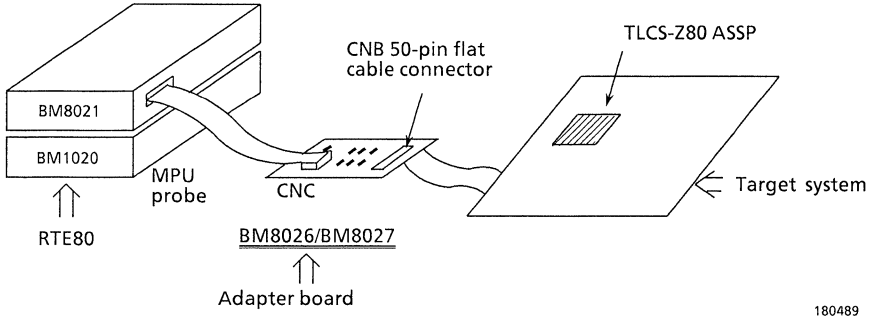
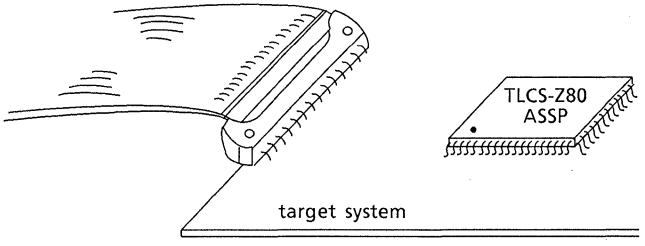


Figure 4.2 Example of connection (Level 2)
(For the detail, refer to instruction manuals)

4.3 CONNECTION OF BM8026/BM8027 AND TARGET SYSTEM

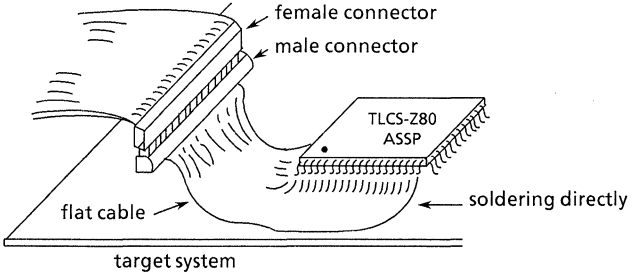
There are three ways to connect 50-pin connector of the adapter board and ASSP on the target system.

(1) Provide a 50-pin flat cable socket on the target system



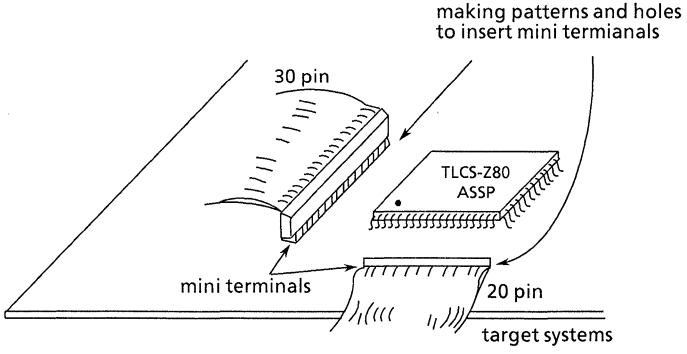
180489

(2) Solder the male connector of the flat cable directly



180489

(3) To provide patterns on the target system to insert mini terminals on a board



180489

The TMPZ84C011A and TMPZ84C015A are housed in the 100-pin mini flat package. When connecting the adapter board (BM8026/BM8027) for them by using any of the ways, take care on the following points.

(1) To use ASSP chip on the target system in the evaluation mode by connecting to the emulator, all bus signals of the MPU in ASSP should be placed in floating state or opened for emulator.

The adapter board can be disconnected from the emulator by controlling the interface circuit inside the emulator, and it can be operated independently.

(2) Tables 4.1 (a) and (b) show the signal names of the connector.

CNA : Connect to the emulator (RTE80 or others)

40-pin DIP socket

pin Name	pin Number		pin Name
A11	1	40	A10
A12	2	39	A9
A13	3	38	A8
A14	4	37	A7
A15	5	36	A6
CLK	6	35	A5
D4	7	34	A4
D3	8	33	A3
D5	9	32	A2
D6	10	31	A1
VCC	11	30	A0
D2	12	29	GND
D7	13	28	$\overline{RFSH}$
D0	14	27	$\overline{M1}$
D1	15	26	$\overline{RESET}$
$\overline{INT}$	16	25	$\overline{BUSREQ}$
$\overline{NMI}$	17	24	$\overline{WAIT}$
$\overline{HALT}$	18	23	$\overline{BUSACK}$
$\overline{MREQ}$	19	22	$\overline{WR}$
$\overline{IORQ}$	20	21	$\overline{RD}$

180489

Table 4.1 (a) Pin Connection Table

Connect to target system
50-pin flat cable socket

CNB : BM8026

pin Name	pin Number		pin Name
D7	1	2	D6
D5	3	4	D4
D3	5	6	D2
D1	7	8	D0
$\overline{INT}$	9	10	EV
$\overline{HALT}$	11	12	$\overline{MREQ}$
$\overline{IORQ}$	13	14	$\overline{WR}$
$\overline{U.RD}$	15	16	$\overline{RD}$
$\overline{U.BUSACK}$	17	18	$\overline{BUSACK}$
$\overline{WAIT}$	19	20	NC
$\overline{U.BUSREQ}$	21	22	$\overline{BUSREQ}$
$\overline{RESET}$	23	24	$\overline{M1}$
$\overline{U.RFSH}$	25	26	$\overline{RFSH}$
VCC	27	28	GND
A0	29	30	A1
A2	31	32	A3
A4	33	34	A5
A6	35	36	A7
A8	37	38	A9
A10	39	40	A11
A12	41	42	A13
A14	43	44	A15
$\overline{NMI}$	45	46	VCC
GND	47	48	CLK
NC	49	50	NC

CNB : BM8027

pin Name	pin Number		pin Name
NC	1	2	D7
D6	3	4	D5
D4	5	6	D3
D2	7	8	D1
D0	9	10	VCC
A15	11	12	A14
A13	13	14	A12
A11	15	16	A10
A9	17	18	A8
A7	19	20	A6
A5	21	22	A4
A3	23	24	A2
A1	25	26	A0
NC	27	28	$\overline{RFSH}$
$\overline{M1}$	29	30	$\overline{RESET}$
$\overline{U.BUSREQ}$	31	32	$\overline{BUSREQ}$
NC	33	34	$\overline{WAIT}$
$\overline{U.BUSACK}$	35	36	$\overline{BUSACK}$
NC	37	38	$\overline{WR}$
$\overline{U.RD}$	39	40	$\overline{RD}$
$\overline{IORQ}$	41	42	GND
$\overline{MREQ}$	43	44	$\overline{HALT}$
$\overline{INT}$	45	46	NC
CLKOUT	47	48	EV
GND	49	50	$\overline{NMI}$

$\overline{U.RD}$, $\overline{U.BUSACK}$, $\overline{U.BUSREQ}$, $\overline{U.RFSH}$ are target system's signals.

180489

Table 4.1 (b) Pin Connection Table

(3) A 50-pin flat cable connector is provided on the adapter board to connect ASSP chip on the target board.

(4) When connecting the adapter board and the target system :

a) Connect all the signals of the TMPZ84C011A or TMPZ84C015A to the target system directly or via the connector

b) EV pin is driven by adapter board, so pull down this pin by a resistor. (10kΩ – 100kΩ)
(Don't connect directly to Vss)

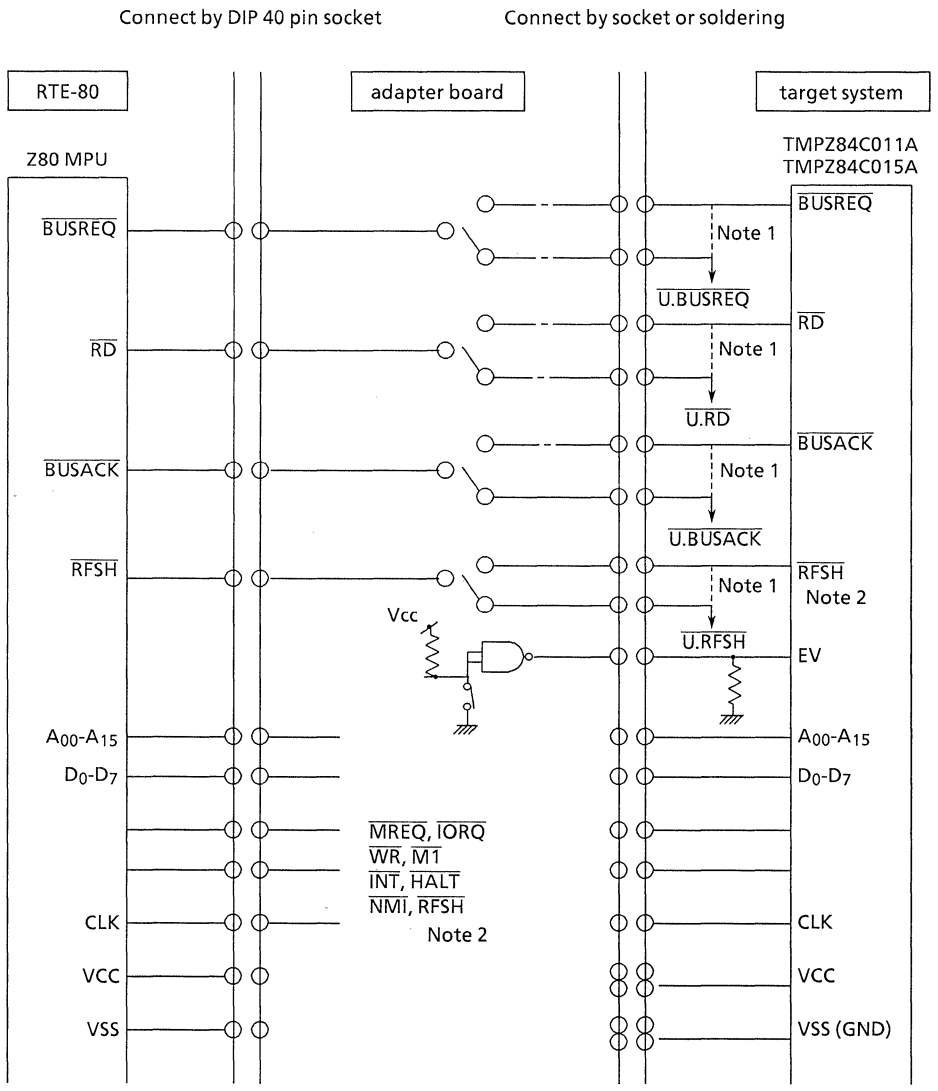
(c) It is not necessary to connect $\overline{\text{BUSREQ}}$ and $\overline{\text{U.BUSREQ}}$, $\overline{\text{BUSACK}}$ and $\overline{\text{U.BUSACK}}$, $\overline{\text{RD}}$ and $\overline{\text{U.RD}}$ on the target system.
(These signals are connected on the adapter board respectively)
Note : When the adapter board is removed, these signals should be connected on the target system, respectively.

(d) When the target system does not use $\overline{\text{BUSREQ}}$ pin and $\overline{\text{BUSACK}}$ pin, $\overline{\text{U.BASACK}}$ pin can be used in open state, and input to $\overline{\text{U.BUSREQ}}$ must be CMOS high level.

Note 1: The following signals in adapter board are pulled up by resistor.
$\overline{\text{HALT}}$, $\overline{\text{RD}}$, $\overline{\text{U.RD}}$, $\overline{\text{M1}}$
$\overline{\text{RFSH}}$ (applied for BM8027 only), and $\overline{\text{BUSACK}}$.

Note 2 : Power Supply of the adapter board must be supplied by the target system.

Figure 4.3 shows the connection of adapter board, target system, and emulator.



Note 1 : Connect these signals when adapter board is disconnected.
Note 2 : For the BM8026, $\overline{\text{RFSH}}$ pin and $\overline{\text{U.RFSH}}$ pin are provided in place of $\overline{\text{RFSH}}$ pin.

180489

Figure 4.3 Interface Circuit

For more details, refer to the following manuals.

1. TLCS-Z80 DEVELOPMENT SYSTEM MANUAL
2. TMPZ84C011A Experimental Board (Evaluation Board : BM8024)
   BM8024 Instruction Manual
3. TMPZ84C015A Experimental Board (Evaluation Board : BM8025)
   BM8025 Instruction Manual
4. TMPZ84C011A Adapter Board (BM8026)
   BM8026 Instruction Manual
5. TMPZ84C015A Adapter Board (BM8027)
   BM8027 Instruction Manual

TOSHIBA

PART 2 8-BIT MICROPROCESSOR

TLCS-85 FAMILY

TOSHIBA CORPORATION

CONTENTS

TMP8085AP-2/TMP8085AHP-2
8-BIT MICROPROCESSOR

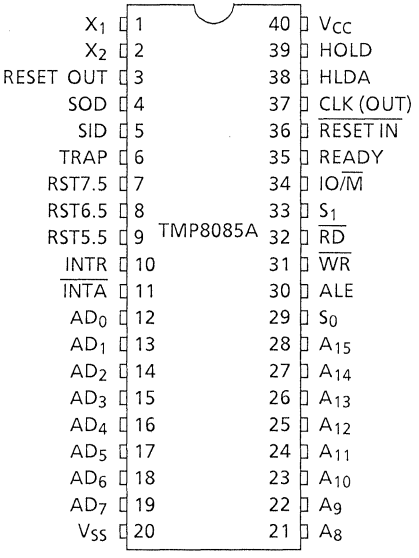
1. GENERAL DESCRIPTION

The TMP8085AP-2/TMP8085AHP-2, hereafter on referred to as TMP8085A, is a 8 bit micro processing unit (MPU). TMP8085A uses a multiplexed data bus. The address is split between the 8 bit address bus and the 8 bit data bus. The on-chip address latches of TMP8155P-2/TMP8156P-2 allow a direct interface with TMP8085A.

2. FEATURES

- 0.8μSec Instruction Cycle (CLK Cycle Period @200nSec)
- Single +5V Power Supply
  (TMP8085AP-2: 5V±5%, TMP8085AHP-2: 5V±10%)
- On-Chip Clock Generator (with External Crystal or RC Network)
- On-Chip System Controller; Cycle status information available for Large System Control
- 4 Vectored Interrupts (One is Non-Maskable)
- Decimal, Binary and Double Precision Arithmetic
- Serial In/Serial Out Port
- Direct Addressing Capability up to 64K Byte Memory Space
- Compatible with Intel's 8085A
- Low Power Consumption (TMP8085AHP-2: Icc max=135mA)

3. PIN CONNECTION (TOP VIEW)



050489

Figure 3.1

4. BLOCK DIAGRAM

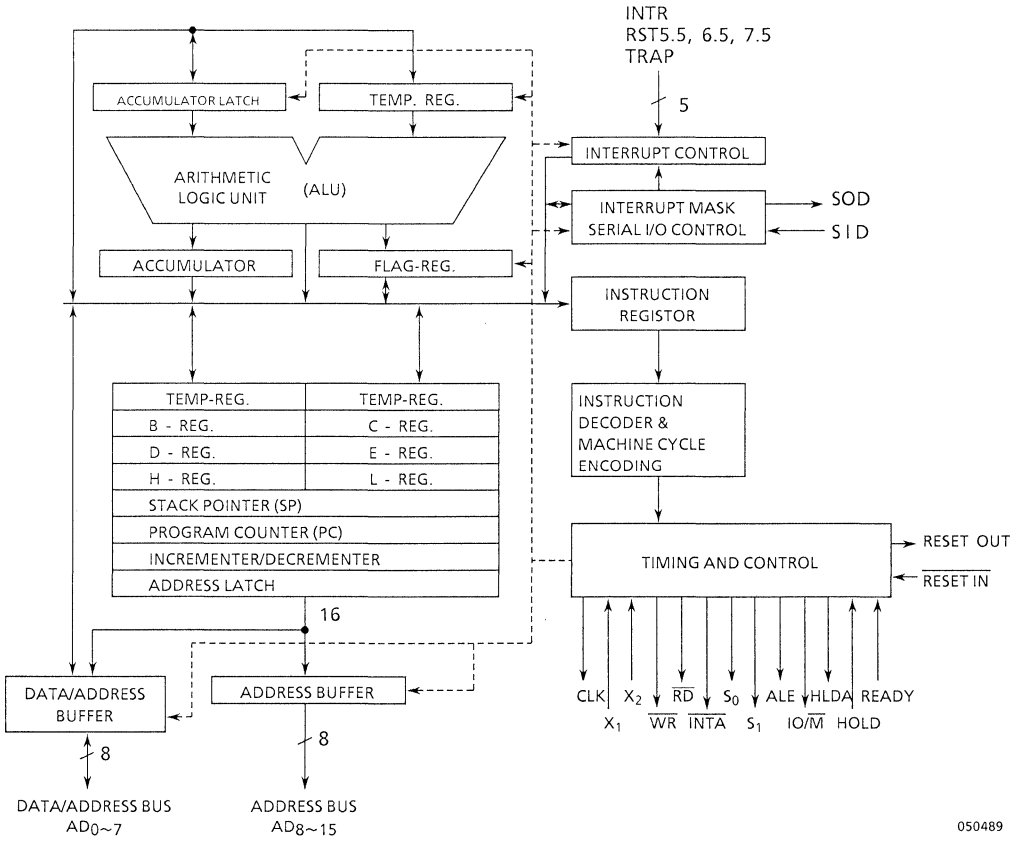


Figure 4.1

5. PIN NAME AND PIN DESCRIPTION

- X_1, X_2 (Input)

  Crystal, LC, or RC network are connected to X_1 and X_2 to drive the internal clock generator. X_1 and X_2 can also be driven by an external clock source. The input frequency is devided by 2 to give the processor's internal operating frequency.

- CLK(Output)

  Clock Output for use as a system clock. The period of CLK is twice the X_1, X_2 input period.

- $\overline{\text{RESETIN}}$ (Input)

  The $\overline{\text{RESETIN}}$ Input initializes the processor by clearing the program counter, instruction register, SOD latch, Interrupt Enable flip-flop and HLDA flip-flop. The address and data buses and the control lines are 3-stated during RESET and because of the asynchronous nature of RESET, the processor's internal registers and flags may be altered by RESET with unpredictable results. $\overline{\text{RESETIN}}$ is a Schmitttriggered input, allowing connection to an RC network for power on RESET delay. The TMP8085A is held in the reset condition as long as $\overline{\text{RESETIN}}$ is applied.

- RESET OUT (Output)

  The RESET OUT signal indicates that the TMP8085A is being reset. It can be used as a system reset. It is synchronized to the processor clock and lasts an integral number of clock periods.

- SOD (Output)

  Serial output data line. The output SOD is set or reset as specified by the SIM instruction.

- SID (Input)

  Serial input data line. The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed.

- INTR (Input)

  INTERRUPT REQUEST signal provides a mechanism for external devices to modify the instruction flow of the program in progress. It is sampled only during the next to the last clock cycle of an instruction and during Hold and Halt states. If it is recognized, the processor will complete the execution of the current instruction, and then the Program Counter (PC) will be inhibited from incrementing and an $\overline{\text{INTA}}$ will be issued. During this cycle a RESTART or CALL instruction can be inserted to jump to the interrupt service routine. The INTR is enabled and disabled by software. It is disabled by RESET and immediately after an interrupt is accepted.

- $\overline{\text{INTA}}$ (Output)

  INTERRUPT ACKNOWLEDGE: Occurs in response to an Interrupt input and indicates that the processor will be ready for an interrupt instruction on the data bus. It is used instead of (and has the same timing as) $\overline{\text{RD}}$ during the instruction cycle after an INTR is accepted.

RST 5.5
RST 6.5 } (Inputs)
RST 7.5

RESTART INTERRUPTS: These three inputs have the same timing as INTR except they cause an internal RESTART to be automatically inserted. These interrupts have a higher priority than INTR. The priority of these interrupts is ordered as shown Table 1. These interrupts are maskable using the SIM instruction.

- TRAP (Input)

  Trap interrupt is a nonmaskable RESTART interrupt. It is sampled at the same timing as INTR or RST 5.5-7.5. It is unaffected by SIM or Enable/Disable Interrupt instruction. It has the highest priority of any interrupt.

- AD_0-AD_7 (Input/Output, 3-state)

  Lower 8 bits of the memory address (or I/O address) appear on the bus during the first clock cycle (T_1 state) of a machine cycle. It then becomes the data bus during the second and third clock cycles.

- A_8-A_{15} (Output, 3-state)

  Upper 8 bits of the memory address or the 8 bits of the I/O address, 3-stated during Hold and Halt modes and during RESET.

- S_0, S_1, and IO/$\overline{M}$ (Output)

  Machine cycle status:

IO/$\overline{M}$	S_1	S_0	Status
0	1	1	Opcode fetch
0	1	0	Memory read
0	0	1	Memory write
1	1	0	I/O read
1	0	1	I/O write
1	1	1	Interrupt Acknowledge
TS	0	0	Halt
TS	X	X	Hold
TS	X	X	Reset

  Note : TS = 3-state (high impedance)
  X = unspecified

- ALE (Output)

  Address Latch Enable: It occurs during the first clock state of a machine cycle and enables the address to get latched into the external latch or the on-chip latch of peripherals. The falling edge of ALE can be used to strobe the status information. ALE never goes 3-stated.

- $\overline{WR}$ (Output, 3-state)

  WRITE control: A low level on $\overline{WR}$ indicates the data on the Data Bus is to be written into the selected memory or I/O location. Data is set up at the trailing edge of $\overline{WR}$. It is 3-stated during Hold and Halt modes and during RESET.

- $\overline{RD}$ (Output, 3-state)

  READ control: A low level on $\overline{RD}$ indicates the selected memory or I/O device to be read and that the Data Bus is available for the data transfer, 3-stated during Hold and Halt modes and during RESET.

- READY (Input)

  When READY is inactive (low), indicating the external operation has not been complete yet, the processor will enter the Wait state. It will wait for an integral number of clock cycles until READY goes high before completing the read or write cycle.

- HOLD (Input)

  The HOLD input cause TMP8085A to release the control over the address bus and the data bus. When HOLD goes active, the processor completes its current operation, activates the HLDA output, and puts the Address, Data, $\overline{RD}$, $\overline{WR}$, and IO/$\overline{M}$ lines into high-impedance state. Internal processing can continue. The Holding device can then utilize the address and data buses instead of TMP8085A. TMP8085A can regain the bus only after the HOLD goes inactive (low).

- HLDA (Output)

  The Hold Acknowledge output, HLDA signal is a response to a HOLD input. It indicates that TMP8085A has received the HOLD request and HOLD will release the bus control in the next cycle. HLDA goes low after the HOLD goes inactive (low). TMP8085A takes over the bus control one half clock after HDLA goes low.

- V_{CC}

  +5 volt supply

- V_{SS}

  Ground Reference

6. FUNCTIONAL DESCRIPTION

TMP8085A is a 8-bit central processor.

TMP8085A is provided with internal 8-bit registers and 16-bit registers. TMP8085A has eight addressable 8-bit registers. Six of them can be used either as 8-bit registers or as 16-bit register pairs. In addition to the register pairs, the TMP8085A contains two additional 16-bit registers. TMP8085A register set is as follows:

- The acccumulator (A Register) is the focus of all of the accumulator instructions, which include arithmetic, logic, load and store, and I/O instructions.
- The program counter (PC) always points to the memory location of the next instruction to be executed.
- General-purpose registers BC, DE, and HL may be used as 8-bit registers or as three 16-bit registers, interchangeably, depending on the instruction being performed.
- The stack pointer (SP) is a special data pointer that always points to the stack top (next available stack address).
- The flag register contains five one-bit flags, each of which records processor status information and may also control processor operation.

The five flags in TMP8085A are shown below:

(MSB)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
S	Z		AC		P		C

- The carry flag (C) is set and reset by arithmetic operations. An additional operation that resutls in an overflow out of the high-order bit of the accumulator sets the carry flag. The carry flag also acts as a "borrow" flag for subtract instruction.
- The auxiliary carry flag (AC) indicates overflow out of bit 3 of the accumulator in the same way that C flag indicates overflow out of bit 7. This flag is commonly used in BCD arithmetic.
- The sign flag (S) is set to the condition of the most significant (MSB) bit of the accumulator following the execution of arithmetic or logic instructions.
- The zero flag (Z) is set if the result generated by certain instructions is zero. The zero flag is cleared if the result is not zero.
- The parity flag (P) is set to 1 if the parity (number of 1-bits) of the accumulator is even. If odd, it is cleared. TMP8085A offers the functions of clock generation,

system bus control, and interrupt priority selection as well as execution of the instruction set. TMP8085A uses a multiplexed Data Bus. The address is split between the higher 8-bit Address Bus and the lower 8-bit Address/Data Bus. During the first T state (T_1 clock cycle) of a machine cycle the lower order address is sent out on the Address/Data Bus. These lower 8 bits may be latched externally by the Address Latch Enable signal (ALE). During the rest of the machine cycle, the data bus is used for memory or I/O data transferring.

7. INTERRUPT AND SERIAL I/O

TMP8085A has 5 interrupt inputs: INTR, RST 5.5, RST 6.5, RST 7.5, and TRAP. INTR is identical in function to 8080A INT. Each of three RESTART inputs 5.5, 6.5, 7.5, has a programmable mask. TRAP is also one of RESTART interrupts but it is nonmaskable.

The three RESTART interrupts cause the internal execution of RESTART if the interrupts are enabled and if the corresponding interrupt mask is not set. The nonmaskable TRAP always causes internal interrupt execution whether INTR, RST 5.5, RST 6.5 and RST 7.5 interrupts are enable or not.

There are two different types of inputs in the restart interrupts. RST 5.5 and RST 6.5 are high-level sensitive like INTR and are recognized with the same timing as INTR. RST 7.5 is rising edge-sensitive.

For RST 7.5, only a pulse is required to set an internal flip-flop which generates the interrupt request. The RST 7.5 request flip-flop remains set until the request is serviced. Then it is reset automatically. This flip-flop may also be reset by using the SIM instruction or by issuing a $\overline{\text{RESETIN}}$ to TMP8085A. The RST 7.5 internal flip-flop will be set by a pulse on the RST 7.5 pin even when the RST 7.5 interrupt is masked.

The interrupts are arranged in a fixed priority that determines which interrupt is to be recognized if more than one is pending: TRAP-highest priority. RST 7.5, RST 6.5, RST 5.5, INTR-lowest priority. This priority scheme does not take into account the priority of a routine that was started by a higher priority interrupt. RST 5.5 can interrupt a RST 7.5 routine if the interrupts were reenabled before the end of the RST 7.5 routine.

The TRAP interrupt is useful for catastrophic errors such as power failure or bus error. It is not affected by any flag or mask. The TRAP input is both edge and level sensitive. The TRAP input must go high and remain high until it is acknowledged. It will not be recognized again until it goes low, then high again. This avoids any false triggering due to noise or logic glitches.

Table 7.1 Interrupt Priority, Restart Address, and Sensitivity

Name	Priority	Address branched to When Interrupt Occurs	Type Trigger
TRAP	1	24 (Hex.)	Rising edge and high level until sampled.
RST 7.5	2	3C (Hex.)	Rising edge (latched) .
RST 6.5	3	34 (Hex.)	High level until sampled.
RST 5.5	4	2C (Hex.)	High level until sampled.
INTR	5	See Note (2)	High level until sampled.

050489

Notes : (1) The processor pushes the PC on the stack before branching to the indicated address.
(2) The address branched to depends on the instruction provided to TMP8085A when the interrupt is acknowledged.

The TRAP interrupt is special in that it disables all other interrupts, but preserves the previous interrupt enable status. Performing the first RIM instruction following a TRAP interrupt allows you to determine whether interrupts were enabled or disabled prior to the TRAP. All subsequent RIM instruction provide current interrupt enable status. Performing a RIM instruction following INTR, or RST 5.5-7.5 will provide current interrupt enable status, revealing that interrupts are disabled.

The serial I/O system is also controlled by the RIM and SIM instructions. SID is read by RIM, and SIM sets the SOD date.

8. BASIC TIMING

The execution of each instruction by the TMP8085A consists of a sequence of from one to five machine cycles, and cach machine cycle consists of a minimum of from three to six clock cycles. Most machine cycles consist of three T states, (cycles of the CLK output) with the exception of opecode fetch, which normally has either four or six T states (unless WAIT or HOLD states are forced by the receipt of READY or HOLD inputs). Any T state must be one of ten possible states, shown in Table 3.

At the beginning of every machine cycle, TMP8085A sends out three status signals ($IO/\overline{M}$, S_1, S_0) that define what type of machine cycle is about to take place. TMP8085A also sends out a 16-bit address at the beginning of every machine cycle to identify the particular memory location or I/O port that the machine cycle applies to.

The special timing signal, ADDRESS LATCH ENABLE (ALE), is used a strobe to sample the lower 8-bits of address on the AD_0-AD_7 lines. ALE is present during T_1 of every machine cycle. Control lines $\overline{RD}$ ($\overline{INTA}$) and $\overline{WR}$ become active later, at the time when the transfer of data is to take plece.

Figure 3 shows an instruction fetch, memory read and I/O write cycle (as would occur during processing of the OUT instruction).

Table 8.1 TMP8085 Machine Cycle Chart

MACHINE CYCLE	$IO/\overline{M}$	S_1	S_0	$\overline{RD}$	$\overline{WR}$	$\overline{INTA}$
OPCODE FETCH	0	1	1	0	1	1
MEMORY READ	0	1	0	0	1	1
MEMORY WRITE	0	0	1	1	0	1
I/O READ	1	1	0	0	1	1
I/O WRITE	1	0	1	1	0	1
ACKNOWLEDGE OF INTR	1	1	1	1	1	0
BUS IDLE: DAD	0	1	0	1	1	1
ACK. OF RST, TRAP	1	1	1	1	1	1
HALT	TS	0	0	TS	TS	1

050489

Notes : 0 = Logic "0", 1 = Logic "1", TS = High Impedance

Table 8.2 TMP8085 Machine State Chart

MACHINE STATE	S_1, S_0	IO/$\overline{M}$	A_8-A_{15}	AD_0-AD_7	$\overline{RD}$, $\overline{WR}$	$\overline{INTA}$	ALE
T_1	X	X	X	X	1	1	1°
T_2	X	X	X	X	X	X	0
T_{WAIT}	X	X	X	X	X	X	0
T_3	X	X	X	X	X	X	0
T_4	1	0 +	X	TS	1	1	0
T_5	1	0 +	X	TS	1	1	0
T_6	1	0 +	X	TS	1	1	0
T_{RESET}	X	TS	TS	TS	TS	1	0
T_{HALT}	0	TS	TS	TS	TS	1	0
T_{HOLD}	X	TS	TS	TS	TS	1	0

050489

Notes : (1) 0 = LOGIC "0", 1 = Logic "1", TS = High Impedance, X = Unspecified
(2) °ALE not generated during 2nd and 3rd machine cycles of DAD instruction
(3) +IO/$\overline{M}$ = 1 during T_4-T_6 of INA maachine cycle

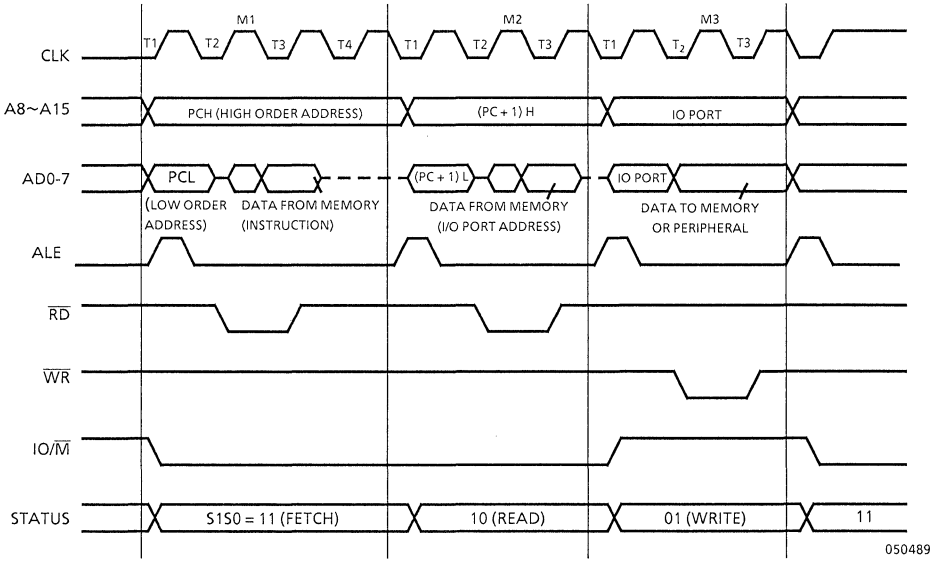


Figure 8.1 TMP8085A Basic System Timing

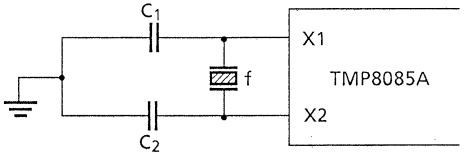
9. DRIVING THE X1 AND X2 INPUTS

The clock inputs of TMP8085A may be driven by a crystal oscilator, an LC tuned circuit, an RC network or an external clock source. The Minimum driving frequency must be 1 MHz, and must be twice as much as the desired internal clock frequency.

(1) Quartz Crystal Clock Driver

If a crystal used, it must have the following characteristics.

- Parallel resonance at twice the clock frequency desired
- C_S (shunt capacitance) $\leqq$ 7 pF
- R_S (equivalent shunt resistance) $\leqq$ 75 Ohms



050489

Figure 9.1

Note : A value of the external capacitors C_1 and C_2 between X1, X2 and ground.
The following values are recommended

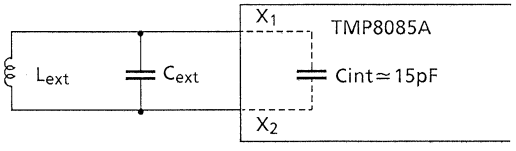
$1MHz \leqq f < 4MHz$: $C_1 = 20pF, C_2 = 20pF$
$4MHz \leqq f \leqq 8MHz$: $C_1 = 10pF, C_2 = 10pF$
$8MHz < f \leqq 10MHz$: $C_1 = 0, C_2 = 0$

(2) LC Turned Circuit Clock Driver

A parallel-resonant LC circuit may be used as the frequency-determining network for TMP8085A, providing that its frequency tolerance of approximately 10% is acceptable. The components are chosen from the formula.

$$f = \frac{1}{2\pi\sqrt{L\,(C_{ext} + C_{int})}}$$

The use of an LC circuit is not recommended for frequencies higher than approximately 5 MHz.



050489

Figure 9.2

(3) RC Circuit Clock Driver

An RC circuit may be used in the case without precise clock frequency stability. Variations in the on-chip timing generation can cause a wide variation in frequency when using RC circuit. The driving frequency generated by the circuit shown is approximately 3 MHz. It is not recommended that frequencies greatly higher or lower than this be attempted.

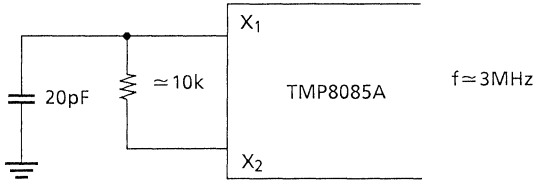


Figure 9.3

(4) External clock Driver Circuit

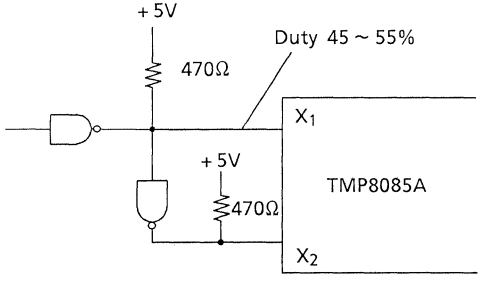


Figure 9.4

10. POWER-ON-RESET

The TMP8085A is not guaranteed to work until 10 ms after V_{CC} reaches 4.75V. It is suggested that $\overline{\text{RESETIN}}$ be kept low during this period. Note that the 10 ms period does not include the time it takes for the power supply to reach its 4.75V level.

For the TMP8085AHP-2, Power-on-reset can be performed when the supply voltage reaches 4.50V.

11. INSTRUCION SET

The code and function of machine instructions are shown as following table. In this table the symbols and abbreviated symbols are used for describing the instructions. However, the symbols required for special attention are explained on the page using them.

Explanation of Symbols

Division	Symbol	Description	
Register	r, g	Register	B, C, D, E, H, L, A
	t	Register pair	BC, DE, HL
		Stack pointer	SP
	q	Register pair	BC, DE, HL
		Program status word	PSW
Memory	mn	Memory address or immediate data indicated by 16 bits m expresses higher 8 bits and n expresses lower 8 bits	
	(mn)	Contents of memory address indicated by mn	
	(HL)	Contents of memory address indicated by register pair	
		Contents of memory address indicated by register pair BC, DE and SP	
Flag status	0	Be reset to 0	
	1	Be set to 1	
	-	No change	
	*	Be affected by operation (be set to 0 or 1.)	
Operation symbol	←	Transfer	
	↔	Exchange	
	+	Addition	
	−	Subtraction	
	∧	Logical AND for every bit	
	∨	Logical OR for every bit	
	∀	Exclusive OR for every bit	
Other	IM	Interrupt mask register	
	CY	Carry flag	

050489

TLCS-85 List of Machine Instruction (1/2)

CLASS / ITEM	ASSEMBLER MNEMONIC	OBJECT CODE BIN (76 543 210)	OBJECT CODE HEX	FUNCTION	FLAGS CY	Z	S	P	AC	CYCL	STAT
8bit load	MOV g, r	01 ggg rrr	40+g×8+r	g←r	–	–	–	–	–	1	4
	MOV g, M	01 ggg 110	46+g×8	g←(HL)	–	–	–	–	–	2	7
	MOV M, r	01 110 rrr	70+r	(HL)←r	–	–	–	–	–	2	7
	MVI g, n	00 ggg 110 nn nnn nnn	06+g×8 n	g←n	–	–	–	–	–	2	7
	MVI M, n	00 110 110 nn nnn nnn	36 n	(HL)←n	–	–	–	–	–	3	10
	LDAX B	00 001 010	0A	A←(BC)	–	–	–	–	–	2	7
	LDAX D	00 011 010	1A	A←(DE)	–	–	–	–	–	2	7
	LDA mn	00 111 010 nn nnn nnn mmmmmmmm	3A n m	A←(mn)	–	–	–	–	–	4	13
	STAX B	00 000 010	02	(BC)←A	–	–	–	–	–	2	7
	STAX D	00 010 010	12	(DE)←A	–	–	–	–	–	2	7
	STA mn	00 110 010 nn nnn nnn mmmmmmmm	32 n m	(mn)←A	–	–	–	–	–	4	13
16bit load	LXI t, mn	00 tt0 001 nn nnn nnn mmmmmmmm	01+t×16 n m	t←mn	–	–	–	–	–	3	10
	LHLD mn	00 101 010 nn nnn nnn mmmmmmmm	2A n m	H←(mn+1) L←(mn)	–	–	–	–	–	5	16
	SHLD mn	00 100 010 nn nnn nnn mmmmmmmm	22 n m	(mn+1)←H (mn)←L	–	–	–	–	–	5	16
	SPHL	11 111 001	F9	SP←HL	–	–	–	–	–	1	6
	PUSH q	11 qq0 101	C5+q×6	(SP−1)(SP−2)←q SP←SP−2	–	–	–	–	–	3	12
	POP q	11 qq0 001	C1+q×6	q ←(SP+1)(SP) SP ←SP+2	–	–	–	–	–	3	10
(a)	XCHG	11 101 011	EB	DE⇔HL	–	–	–	–	–	1	4
	XTHL	11 100 011	E3	H⇔(SP+1),L⇔(SP)	–	–	–	–	–	5	16
8bit arithmetic and logical	ADD r	10 000 rrr	80+r	A ←A+r	*	*	*	*	*	1	4
	ADD M	10 000 110	86	A ←A+(HL)	*	*	*	*	*	2	7
	ADI n	11 000 110 nn nnn nnn	C6 n	A ←A+n	*	*	*	*	*	2	7
	ADC r	10 001 rrr	88+r	A ←A+r+CY	*	*	*	*	*	1	4
	ADC M	10 001 110	8E	A ←A+(HL)+CY	*	*	*	*	*	2	7
	ACI n	11 001 110 nn nnn nnn	CE n	A ←A+n+CY	*	*	*	*	*	2	7
	SUB r	10 010 rrr	90+r	A ←A−r	*	*	*	*	*	1	4
	SUB M	10 010 110	96	A ←A−(HL)	*	*	*	*	*	2	7
	SUI n	11 010 110 nn nnn nnn	D6 n	A ←A−n	*	*	*	*	*	2	7
	SBB r	10 011 rrr	98+r	A ←A−r−CY	*	*	*	*	*	1	4
	SBB M	10 011 110	9E	A ←A−(HL)−CY	*	*	*	*	*	2	7
	SBI n	11 011 110 nn nnn nnn	DE n	A ←A−n−CY	*	*	*	*	*	2	7
	ANA r	10 100 rrr	A0+r	A ←A∧r	0	*	*	*	1	1	4
	ANA M	10 100 110	A6	A ←A∧(HL)	0	*	*	*	1	2	7
	ANI n	11 100 110 nn nnn nnn	E6 n	A ←A∧n	0	*	*	*	1	2	7
	ORA r	10 110 rrr	B0+r	A ←A∨r	0	*	*	*	0	1	4
	ORA M	10 110 110	B6	A ←A∨(HL)	0	*	*	*	0	2	7
	ORI n	11 110 110 nn nnn nnn	F6 n	A ←A∨n	0	*	*	*	0	2	7
	XRA r	10 101 rrr	A8+r	A ←A∀r	0	*	*	*	0	1	4
	XRA M	10 101 110	AE	A ←A∀(HL)	0	*	*	*	0	2	7

rrr / ggg	Register
000	B
001	C
010	D
011	E
100	H
101	L
111	A

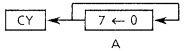
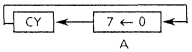
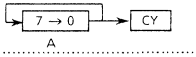
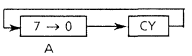
tt	Register
00	BC
01	DE
10	HL
11	SP

qq	Register
00	BC
01	DE
10	HL
11	PSW

PSW only

050489

TLCS-85 List of Machine Instruction (2/2)

CLASS / ITEM	ASSEMBLER MNEMONIC	OBJECT CODE BIN (76 543 210)	OBJECT CODE HEX	FUNCTION	FLAGS CY	Z	S	P	AC	CYCL	STAT
8bit arithmetic and logical	XRI n	11 101 110 nn nnn nnn	EE n	A←A∀n	0	*	*	*	0	2	7
	CMP r	10 111 rrr	B8+r	A-r	*	*	*	*	*	1	4
	CPM M	10 111 110	BE	A-(HL)	*	*	*	*	*	2	7
	CPI n	11 111 110 nn nnn nnn	FE n	A-n	*	*	*	*	*	2	7
	INR g	00 ggg 100	04+g×8	g ←g+1	–	*	*	*	*	1	4
	INR M	00 110 100	34	(HL)←(HL)+1	–	*	*	*	*	3	10
	DCR g	00 ggg 101	05+g×8	g ←g-1	–	*	*	*	*	1	4
	DCR M	00 110 101	35	(HL)←(HL)-1	–	*	*	*	*	3	10
(b)	DAA	00 100 111	27	Decimal adjust accumulator	*	*	*	*	*	1	4
	CMA	00 101 111	2F	A←NOT A	–	–	–	–	–	1	4
	CMC	00 111 111	3F	CY←NOT CY	*	–	–	–	–	1	4
	STC	00 110 111	37	CY←1	1	–	–	–	–	1	4
	NOP	00 000 000	00	no operation	–	–	–	–	–	1	4
	HLT	01 110 110	76	MPU halt	–	–	–	–	–	1	5
	DI	11 110 011	F3	INTE F/F Reset	–	–	–	–	–	1	4
	EI	11 111 011	FB	INTE F/F Set	–	–	–	–	–	1	4
(c)	DAD t	00 tt1 001	09+t×10	HL←HL+t	*	–	–	–	–	3	10
	INX t	00 tt0 011	03+t×10	t ←t+1	–	–	–	–	–	1	6
	DCX t	00 tt1 011	0B+t×10	t ←t+1	–	–	–	–	–	1	6
Rotate and sift	RLC	00 000 111	07		*	–	–	–	–	1	4
	RAL	00 010 111	17		*	–	–	–	–	1	4
	RRC	00 001 111	0F		*	–	–	–	–	1	4
	RAR	00 011 111	1F		*	–	–	–	–	1	4
Jump and call and return	JMP mn	11 000 011 nn nnn nnn mmmmmmmm	C3 n m	PC ←mn	–	–	–	–	–	3	10
	Jc mn	11 ccc 010 nn nnn nnn mmmmmmmm	C2+C×8 n m	IF cc is true, PC←mn otherwise, PC←PC+3	–	–	–	–	–	3 2	10 7
	PCHL	11 101 001	E9	PC ←HL	–	–	–	–	–	1	6
	CALL mn	11 001 101 nn nnn nnn mmmmmmmm	CD n m	(SP-1)(SP-2)←PC, SP ←SP-2, PC ←mn	–	–	–	–	–	5	18
	Cc mn	11 ccc 100 nn nnn nnn mmmmmmmm	C4+c×8 n m	IF c is ture, (SP-1)(SP-2) ←PC, SP←SP-2, PC←mn otherwise, PC←PC+3	–	–	–	–	–	5 2	18 9
	RET	11 001 001	C9	PC←(SP+1)(SP), SP←SP+2	–	–	–	–	–	3	10
	Rc	11 ccc 000	C0+C×8	IF c is true, PC←(SP+1)(SP), SP←SP+2 otherwise, PC←PC+1	–	–	–	–	–	3 1	12 6
	RST j	11 kkk 111	C7+k×8	(SP-1)(SP-2)←PC, SP←SP-2, PC←kkk×8	–	–	–	–	–	3	12
(d)	IN n	11 011 011 nn nnn nnn	DB n	A ←(n)	–	–	–	–	–	3	10
	OUT n	11 010 011 nn nnn nnn	D3 n	(n) ←A	–	–	–	–	–	3	10
(e)	RIM	00 100 000	20	A ←IM	–	–	–	–	–	1	4
	SIM	00 110 000	30	IM ←A	–	–	–	–	–	1	4

ccc	c	Condition
000	NZ	no zero
001	Z	zero
010	NC	no carry
011	C	carry
100	PO	parity odd
101	PE	parity even
110	P	sign positive
111	M	sign negative

kkk	j
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

050489

Note : "POP PSW" instruction changes state of flags.

(a) Exechange (b) General purpose arithmetic and MPU control
(c) 16bit arithmetic (d) Input and output (e) Interrupt

12. ELECTRICAL CARACTERISTICS

12.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item	Ratings	Units
VCC	VCC Supply voltage	− 0.5 to + 7.0	V
PD	Power Dissipation	1.5	W
Tsolder	Soldering Temperature	260 (10 sec)	°C
Tstg	Storage Temperature	− 55 to 150	°C
Topr	Operating Temperature	0 to 70	°C

050489

12.2 DC CHARACTERISTICS

TA = 0°C to 70°C, VCC = 5V ± 5% : TMP8085AP-2
VCC = 5V ± 10% : TMP8085AHP-2

Symbol	Parameter	Test Conditions	Min.	Max.	Units
VIL	Input Low Voltage		− 0.5	0.8	V
VIH	Input High Voltage		2.0	VCC + 0.5	V
VOL	Output Low Voltage	IOL = 2mA		0.45	V
VOH	Output High Voltage	IOH = − 400μA	2.4		V
ICC	Power Supply Current	TMP8085AP/AP-2		170	mA
		TMP8085AHP/AHP-2		135	
IIL	Input Leakage	0 ≦ VIN ≦ VCC		± 10	μA
ILO	Output Leakage	0.45 ≦ VOUT ≦ VCC		± 10	μA
VILR	Input Low Level (RESET)		− 0.5	0.8	V
VIHR	Input High Level (RESET)		2.4	VCC + 0.5	V
VHY	Hysteresis (RESET)		0.25		V

050489

12.3 AC CHARACTERISTICS

TA = 0°C to 70°C, V_{CC} = 5V ± 5% : TMP8085AP-2
VCC = 5V ± 10% : TMP8085AHP-2
Vss = 0V, unless otherwise noted.

Symbol	Parameter	Min.	Max.	Units
tCYC	CLK Cycle Period	200	2000	ns
tL	CLK Low Time	40		ns
		50*		ns
tH	CLK High Time	70		ns
		80*		ns
tr, tf	CLK Rise and Fall Time		30	ns
tXKR	X1 Rising to CLK Rising	30	100	ns
tXKF	X1 Rising to CLK Falling	30	110	ns
tAC	A8-15 Valid to Leading Edge of Control [1]	115		ns
tACL	A0-7 Valid to Leading of Control	115		ns
tAD	A0-15 Valid to Valid Data In		350	ns
tAFR	Address Float after Leading Edge of ALE READ (INT)		0	ns
tAL	A8-15 Valid before Trailing Edge of ALE [1]	50		ns
tALL	A0-7 Valid before Trailing Edge of ALE	50		ns
tARY	READY Valid from Address Valid		100	ns
tCA	Address (A8-15) Valid after Control	60		ns
tCC	Width of Control Low (RD, WR, INTA) Edge of ALE	230		ns
tCL	Trailing Edge of Control to Leading Edge of ALE	25		ns
tDW	Data Valid to Trailing Edge of WRITE	230		ns
tHABE	HLDA to Bus Enable		150	ns
tHABF	Bus Float after HLDA		150	ns
tHACK	HLDA Valid to Trailing Edge of CLK	40		ns
tHDH	HOLD Hold Time	0		ns
tHDS	HOLD Setup Time to Trailing Edge of CLK	120		ns
tINH	INTR Hold Time	0		ns
tINS	INTR, RST and TRAP Setup Time to Falling	150		ns
tLA	Edge of CLK Address Hold Time after ALE	50		ns
tLC	Trailing Edge of ALE to Leading Edge of Control	60		ns

050489

Symbol	Parameter	Min.	Max.	Units
tLCK	ALE Low during CLK High	50		ns
tLDR	ALE to Valid Data during Read		270	ns
tLDW	ALE to Valid Data during Write		120	ns
tLL	ALE Width	80		ns
tLRY	ALE to READY Stable		30	ns
tRAE	Trailing Edge of READ to Re-Enabling of Address	90		ns
tRD	READ (or INTA) to Valid Data		150	ns
tRV	Control Trailing Edge of Leading Edge of Next Control	220		ns
tRDH	Data Hold Time After READ INTA	0		ns
tRYH	READY Hold Time	0		ns
tRYS	READY Setup Time to Leading Edge of CLK	100		ns
tWD	Data Valid After Trailing Edge of WRITE	60		ns
tWDL	LEADING Edge of WRITE to Data Valid		20	ns

050489

Test conditions CL = 150pF (*: CL = 50pF + 1TTL)
tcyc = 200ns

Notes : 1. A8-15 address specs apply to IO/M, S0 and S1 expect A8-15 are undifiend during T4-T6 of Cycle whereas IO/M, S0, and S1 are stable.
2. Timing defining signal voltage are;
Output High level = 2.0V, Low level = 0.8V
3. To calculate timing specifications at other value of tCYC use Table 12.1.

12.4 TIMING WAVEFORMS

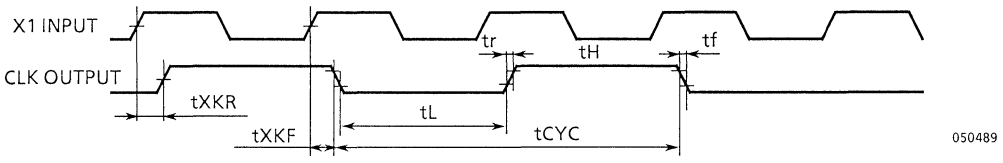


Figure 12.1 Clock Timing Waveform

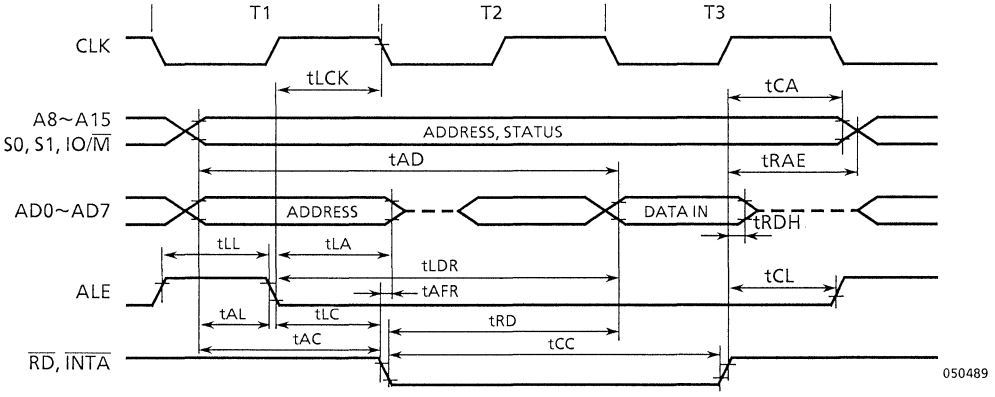


Figure 12.2 Read Operation

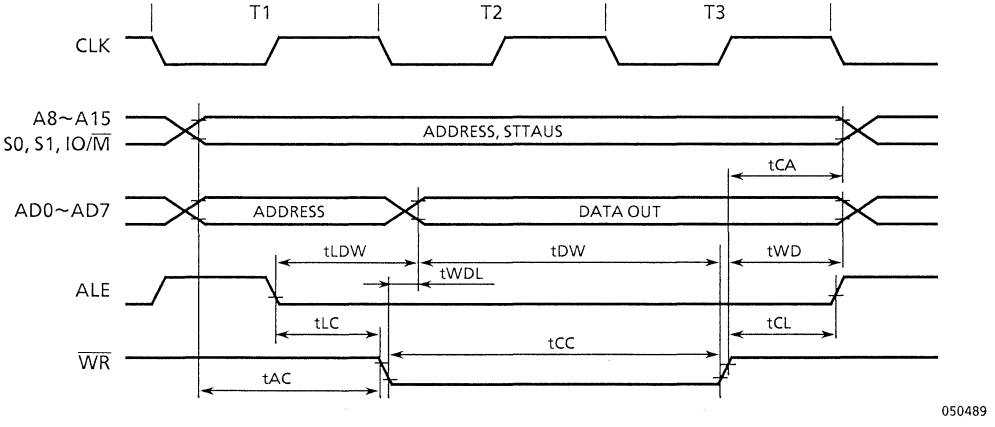


Figure 12.3 Write Operation

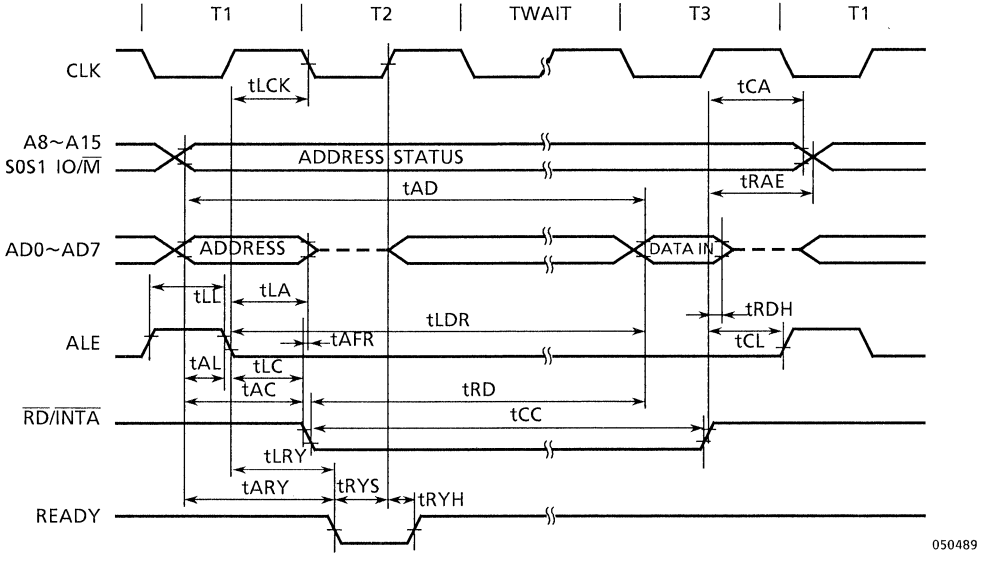


Figure 12.4 Read Operation with Wait Cycle (Typical)
- Same Ready Timing Applied to Write Operation

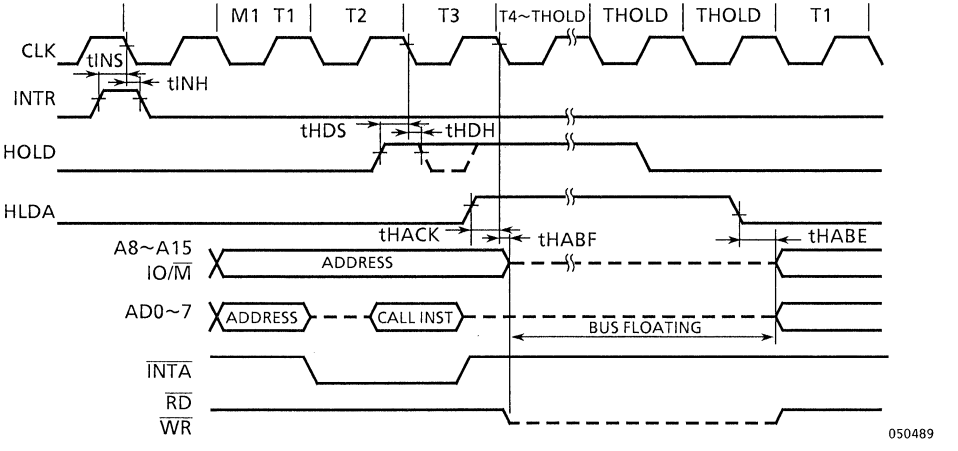
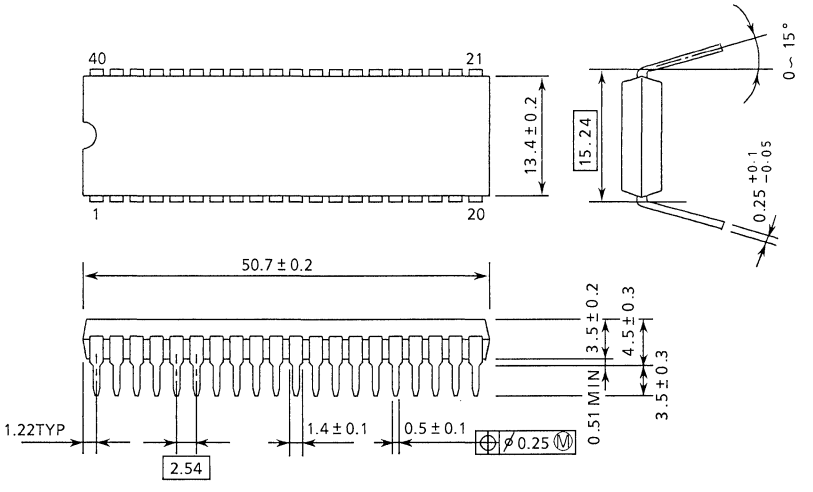


Figure 12.5 Interrupt and Hold Timing

13. OUTLINE DRAWING (40PINS PLASTIC PACKAGE)

DIP40-P-600

Unit : mm



270289

Note: Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoritical positions with respect to No.1 and No.40 leads.

TMP8155P-2/TMP8156P-2

256 BYTE STATIC RAM WITH I/O PORTS AND TIMER

1. GENERAL DESCRIPTION

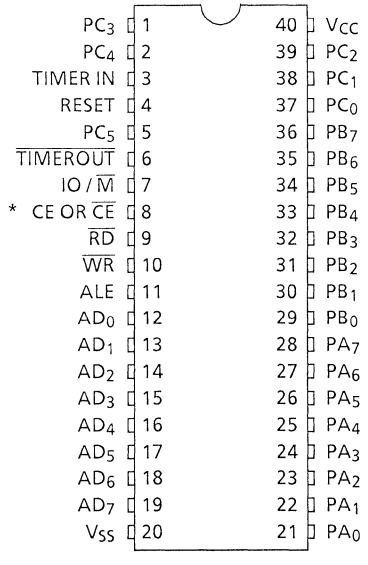
The TMP8155/8156P-2 are RAM including I/O ports and counter/timer on one chip for using in the TLCS-85A microcomputer system. The RAM portion is designed with 2K bit static memory cells organized as 256 x 8. The 14 bit programmable counter/timer is the down counter. It provides either a square wave terminal count pulse for the MPU system depending on timer mode.

The I/O portion consists of 2 programmable 8 bit I/O ports and 1 programmable 6 bit I/O port. The programmble I/O ports can be operated by BASIC MODE and STROBE MODE.

2. FEATURES

- Single +5V Power supply
- Access Time: 330 ns
- Internal Address Latch
- 2 Programmable 8 bit I/O Ports and 1 Programmable 6 bit I/O Port
- 256 x 8 bit RAM
- Programmable 14 bit Binary Counter/Timer
- Multiplexed Address/Data Bus
- Chip Enable Active High (TMP8156P-2) or Low (TMP8155P-2)
- 40 pin DIP

3. PIN CONNECTION (TOP VIEW)

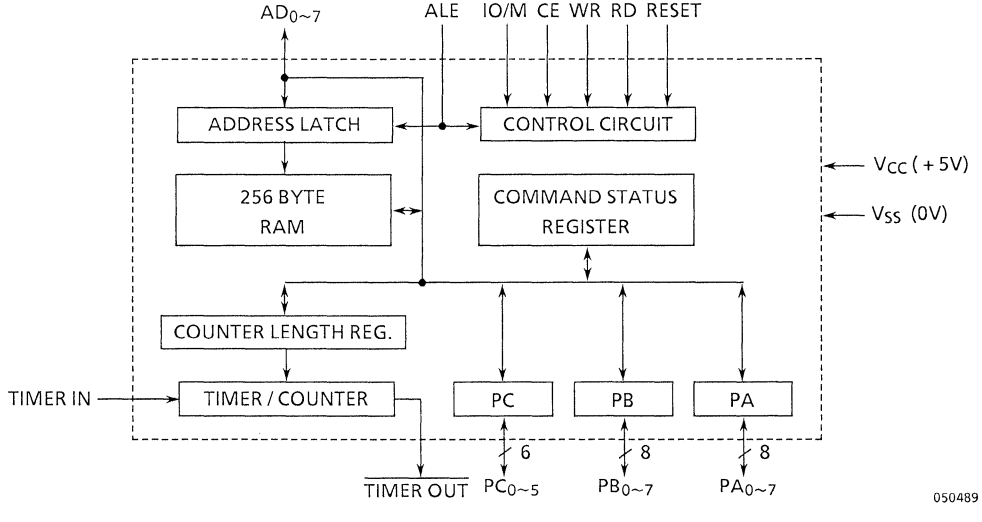


050489

Figure 3.1

Note : * TMP8155 = $\overline{CE}$
TMP8156 = CE

4. BLOCK DIAGRAM



050489

Figure 4.1

5. PIN NAMES AND PIN DESCRIPTION

- RESET (INPUT)

  The Reset signal is a pulse provided by TMP8085A to initialize the system. Input high on this line resets the chip and initializes the internal three I/O ports to input mode.

  The width of RESET pulse should typically be two clock cycles of TMP8085A.

- $AD_{0\sim7}$ (INPUT/OUTPUT, 3-STATE)

  These are 3-state Address/Data lines that interface with the MPU lower 8-bit Address/Data Bus. The 8-bit address can be applied to the memory section or the I/O section depending on the polarity of the IO/$\overline{M}$ input signal. 8-bit data is either written into or read from the TMP8155/8156 depending on the status of $\overline{WR}$ or $\overline{RD}$ input signal.

- CE OR $\overline{CE}$ (INPUT)

  Chip Enable : On the TMP8155P-2, this pin is $\overline{CE}$ and is Active Low.

  On The TMP8156P-2, this pin is CE and is Active High.

- $\overline{RD}$ (INPUT)

  Input low on this line with the Chip Enable active enables the $AD_{0\sim7}$ buffers. If IO/$\overline{M}$ pin is low, the RAM content will be read out from the $AD_{0\sim7}$ bus. Otherwise the content of the selected I/O port or command/status register will be read out from the $AD_{0\sim7}$ bus.

- $\overline{WR}$ (INPUT)

  Input low on this line with the active CE/$\overline{CE}$ causes the data on the $AD_{0\sim7}$ lines to be written to the RAM or I/O ports and command/status register depending on the polarity of IO/$\overline{M}$.

- ALE (INPUT)

  Address Latch Enable : This control signal latches both the data on the $AD_{0\sim7}$ lines and the state of the Chip Enable and IO/$\overline{M}$ into the chip at the falling edge of ALE.

- IO/$\overline{M}$ (INPUT)

  IO/Memory Select : This line selects the memory if low and selects the I/O or command/status register if high.

- $PA_{0\sim7}$ (INPUT/OUTPUT, 3-STATE)

  These 8 pins are general purpose I/O pins. The in/out direction is selected by programming the Command Register.

- PB $_{0\sim7}$ (INPUT/OUTPUT, 3-STATE)
  These 8 pins are general purpose I/O pins. The in/out direction is selected by programming the Command Register.

- PC $_{0\sim5}$ (INPUT/OUTPUT, 3-STATE)
  These 6 pins can function as either input port, output port, or as control signal for PA and PB. Programming is done through the Command Register.
  When PC $_{0\sim5}$ are used as control signals, they are defined as the following :

  PC0 - A INTR (Port A Interrupt)
  PC1 - A BF (Port A Buffer Full)
  PC2 - A $\overline{\text{STB}}$ (Port A Strobe)
  PC3 - B INTR (Port B Interrupt)
  PC4 - B BF (Port B Buffer Full)
  PC5 - B $\overline{\text{STB}}$ (Port B Strobe)

- TIMER IN (INPUT)
  This is the input to the counter-timer.

- $\overline{\text{TIMEROUT}}$ (OUTPUT)
  This pin is the timer output. This output can be either a square wave or a pulse depending on the timer mode.

- VCC (Power)
  +5 volt supply

- VSS (Power)
  Ground Reference

6. FUNCTIONAL DESCRIPTION

6.1 PROGRAMMING THE COMMAND REGISTER

The command register consists of either latches. Four bits (0-3) define the mode of the ports, two bits (4-5) enable or disable the interrupt from port C when is acts as control port, and the last two bits (6-7) are for the timer.

The command register contents can be altered at any time by using the I/O address XXXX000 during a WRITE operation. The functions of each bit of the command register is shown in Figure 6.1

Note that the command register is a write-only register and can not be read.

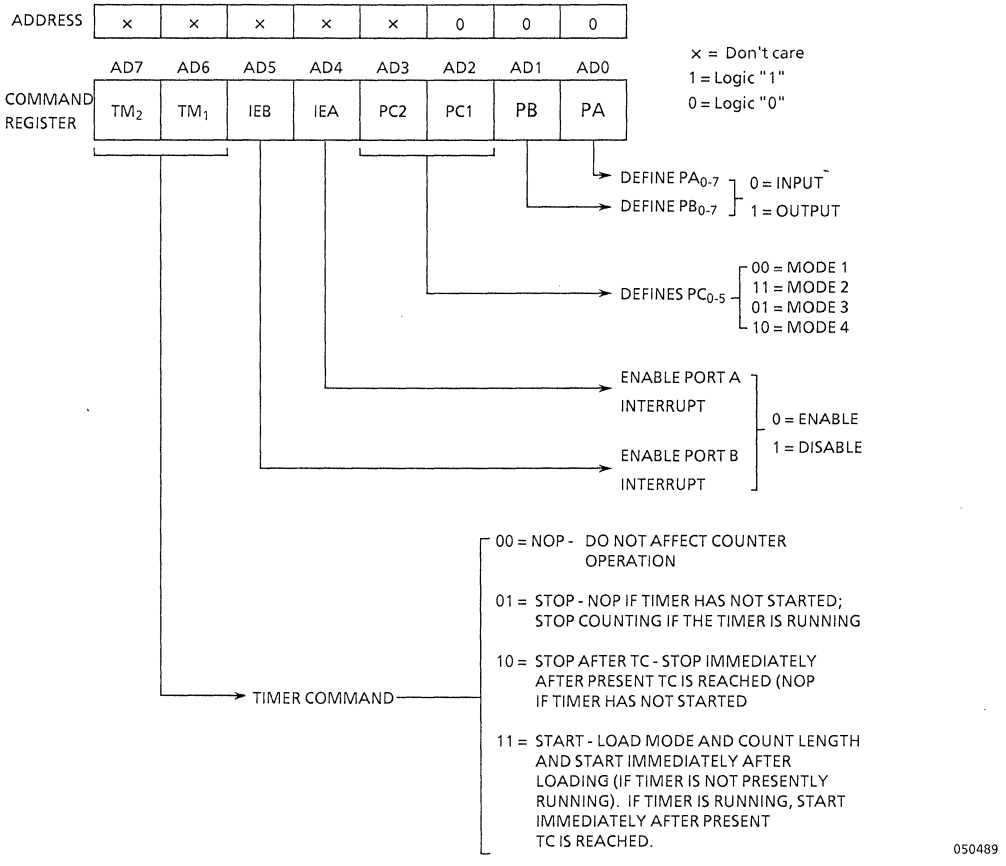


Figure 6.1 Command Register Bit Assignment

6.2 READING THE STATUS REGISTER

The status register consists of seven latches, one for each bit ; six (0-5) for the status of the ports and one (6) for the status of the timer.

The status of the timer and the I/O section can be polled by reading the Status Register (Address XXXX000). Status word format is shown in Figure 6.2.

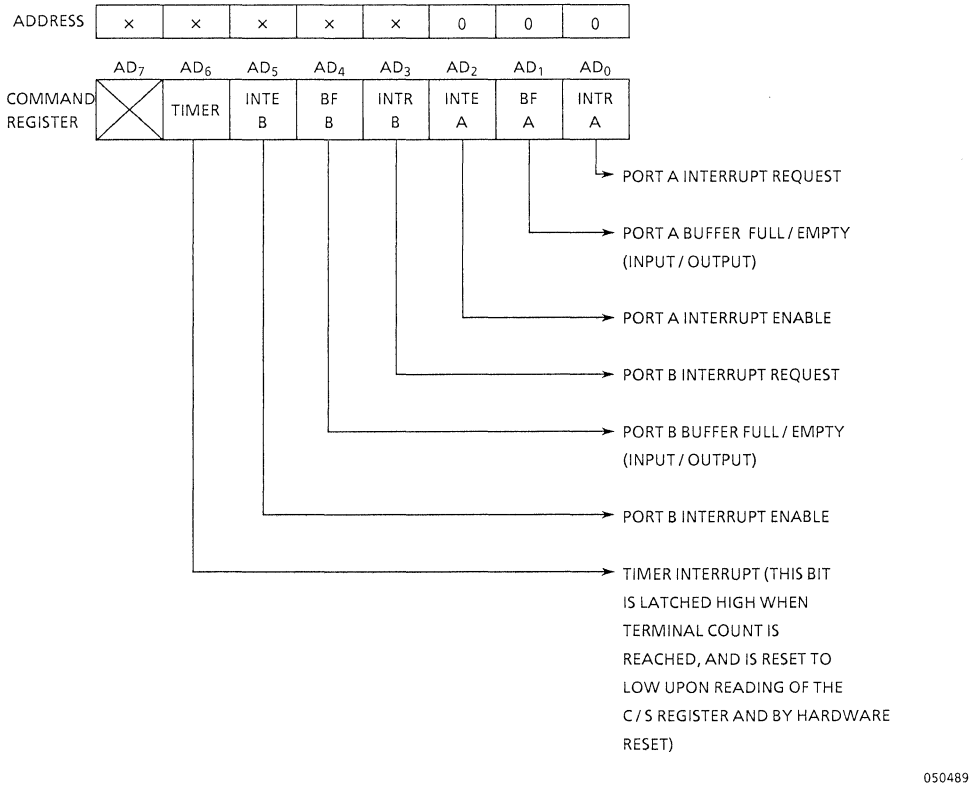


Figure 6.2 Status Register Bit Assignment

6.3 I/O SECTION

- COMMAND/STATUS REGISTER (C/S) – Both registers have the same address XXXXX000. When the C/S registers are selected during WRITE operation, a command is written into the C/S Register. The contents of this register are not accessible through the pins. When the C/S is selected during a READ operation, the status information of the I/O ports and the timer becomes available on the $AD_{0\sim7}$ lines.

- PA Register – This register can be programmed to be either input or output ports depending on the status of the contents of the C/S Register.
  Also depending on the command, this port can operate in either the basic mode or the strobed mode (See timing diagram). The I/O pins assigned in relation to this register are $PA_{0\text{-}7}$. The address of this register is XXXXX001.

- PB Register – This register functions the same as PA Register. The I/O pins assigned are $PB_{0\text{-}7}$. The address of this register is XXXXX010.

- PC Register – This register has the address XXXXX011 and contains only 6-bits. The 6-bits can be programmed to be either input ports, output ports or as control signals for PA and PB by properly programming the AD_2 and AD_3 bits of the C/S Register.
  When $PC_{0\text{-}5}$ is used as a control port, 3-bits are assigned for Port A and 3 for Port B. The first bit is an interrupt that the TMP8155/8156 issues. The second is an output signal indicating whether the buffer is full or empty, and the third is an input pin to accept a strobe for the strobed input mode. See Table 6.2.
  When the port C is programmed to either MODE 3 or MODE 4, the control signals for PA and PB are initialized as follows :

CONTROL MODE	BF	INTR	$\overline{STB}$
INPUT MODE	Low	Low	Input Control
OUTPUT MODE	Low	High	Input Control

050489

To summarize, the register's address assignments are shown in Table 6.1.

Table 6.1 I/O Port Addressing Scheme

I / O ADDRESS									SELECTION	NO. OF BITS
A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0			
X	X	X	X	X	0	0	0	Internal	Command / Status Register	8
X	X	X	X	X	0	0	1	PA_{0-7}	General Purpose I / O Port A	8
X	X	X	X	X	0	1	0	PB_{0-7}	General Purpose I / O Port B	8
X	X	X	X	X	0	1	1	PC_{0-7}	General Purpose I / O Port or Control	6
X	X	X	X	X	1	0	0		Low-Order 8 bits of Timer Count	
X	X	X	X	X	1	0	1		High 6 bits / 2 bits of Timer Count	

050489

Table 6.2 Port Control Assignment

Pin	MODE 1	MODE 2	MODE 3	MODE 4
PC0	Input Port	Output Port	A INTR (Port A Interrupt)	A INTR (Port A Interrupt)
PC1	Input Port	Output Port	A BF (Port A Buffer Full)	A BF (Port A Buffer Full)
PC2	Input Port	Output Port	A $\overline{STB}$ (Port A strobe)	A $\overline{STB}$ (Port A strobe)
PC3	Input Port	Output Port	Output Port	B INTR (Port B Interrupt)
PC4	Input Port	Output Port	Output Port	B BF (Port B Buffer Full)
PC5	Input Port	Output Port	Output Port	B $\overline{STB}$ (Port B strobe)

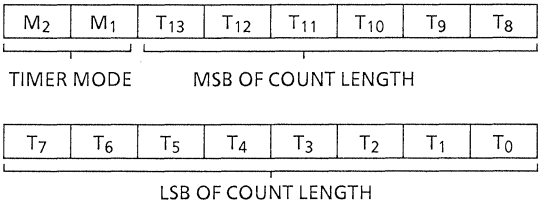
050489

6.4 TIMER SECTION

The timer is a 14-bit down-counter that counts the 'timer input' pulses and provides either a square wave or pulse when terminal count (TC) is reached.

The timer has I/O address XXXXX100 for the low order byte of the register and the I/O address XXXXX101 for the high order of the register.

To program the timer, the COUNT LENGTH REGISTER is loaded first, one byte at a time, by selecting the timer addresses. Bits 0-13 will specify the length of the next count and bits 14-15 will specify the timer output mode. The value loaded into the count length register can have any value from 2H through 3FFFH in bits 0-13.



050489

Figure 6.3 Timer Format

There are four timer modes which are defined by M2 and M1.

M2 M1

0 0 ——— Put out low during second half of count.

0 1 ——— Continuous square wave; The period of the square-wave equals the count length programmed with automatic reload at terminal count.

1 0 ——— Single pulse upon TC being reached.

1 1 ——— Continuous pulses.

Note : In case of an odd-numbered count, the first half-cycle of the square-wave output, which is high, is one count longer than the second (low) half-cycle as shown in Figure 6.3.

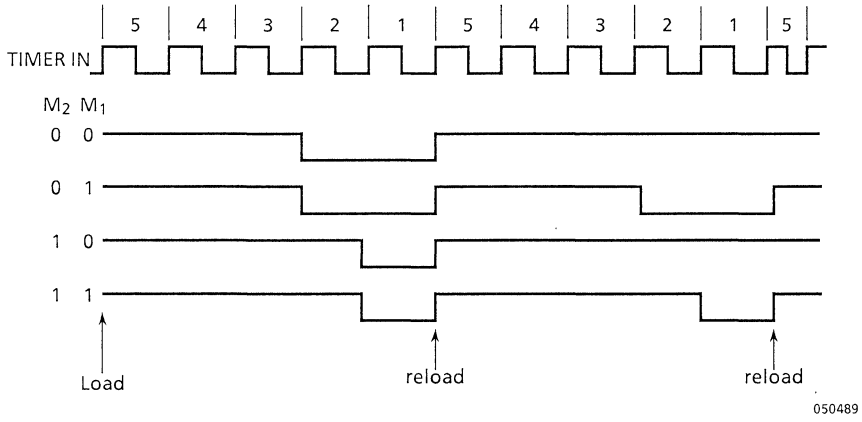


Figure 6.3 Asymmetrical Square-Wave Output resulting from Count 5

Bits 6-7 (TM2 and TM1) of command register contents are used to start and stop the counter. There are four commands to choose from;

TM2	TM1	
0	0	NOP : Do not affect counter operation.
0	1	STOP : NOP if timer has not started; stop counting if the timer is running.
1	0	STOP AFTER TC : Stop immediately after present TC is reached. (NOP if timer has not started)
1	1	START : Load mode and count length and start immediately after loading (if timer is not presently running). If timer is running, start the new mode and count length immediately after present TC is reached.

Note that while the counter is counting, a new count and mode can be loaded into the count length registers. Before the new count and mode will be used by the counter, a START command should be issued to the counter. This applies even though only the change of the count is required in the previous (same) mode.

The counter in the TMP8155/8156 is not initialized to any particular mode or count when hardware RESET occurs, but RESET does stop the counting.

Therefore a START command must be issued via the C/S register, because counting cannot begin following RESET.

Note that the timer circuit on the TMP8155/8156 chip is designed to be a square-wave timer, not an event counter. To achieve this, it counts down by two twice in completing one cycle. Thus, its registers do not contain values directly representing the number of TIMER IN pulses received. Counter value '1' can't be loaded as the initial value into the count register because the timer operates as its terminal count value is 10 (binary). After the timer has started counting down, the values residing in the count registers can be used to calculate the actual number of TIMER IN pulses required to complete the timer cycle if desired. To obtain the remaining count, the following operations should be done in order :

1. Stop the count.
2. Read in the 16-bit value from the count length registers.
3. Reset the upper two mode bits.
4. Reset the carry and rotate right one position all 16 bits through carry.
5. If carry is set, add 1/2 of the full original count (1/2 full count^{-1} if full count is odd.)

Note : When the initial count value is odd and the third count pulse has not come yet, it will be unknown whether one or two counts has occurred. Regardless of this, the TMP8155/8156 always counts out the right number of pulses in generating the $\overline{\text{TIMEROUT}}$ waveforms.

7. ELECTRICAL CHARACTERISTICS

7.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item	Rating	Units
V_{CC}	V_{CC} Supply Voltage with Respect to Vss	− 0.5 to + 7.0	V
P_D	Power Dissipation	1.5	W
T_{SOLDER}	Soldering Temperature (Soldering Time 10 sec.)	260	°C
T_{STG}	Storage Temperature	− 55 to + 150	°C
T_{OPR}	Operating Temperature	0 to + 70	°C

050489

7.2 D.C. CHARACTERISTICS

TA = 0°C to + 70°C, V_{CC} = + 5V ± 5%

Symbol	Parameter	Test Conditions	Min.	Max.	Units
V_{IL}	Input Low Voltage		− 0.5	0.8	V
V_{IH}	Input High Voltage		2.0	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2mA	−	0.45	V
V_{OH}	Output High Voltage	I_{OH} = − 400μA	2.4	−	V
I_{IL}	Input Leakage	V_{IN} = V_{CC} to 0V	−	± 10	μA
I_{LO}	Output Leakage Current	0.45V ≦ V_{OUT} ≦ V_{CC}	−	± 10	μA
I_{CC}	V_{CC} Supply Current		−	180	mA
I_{IL} (CE)	Chip Enable Leakage 8155 8156	V_{IN} = V_{CC} to 0V.	−	 + 100 − 100	 μA μA

050489

7.3 A.C. CHARACTERISTICS

$T_A = 0°C$ to $+70°C$, $V_{CC} = +5V \pm 5\%$

Symbol	Parameters	Min.	Max.	Units
t_{AL}	Address to Latch Setup Time	30	–	ns
t_{LA}	Address Hold Time after Latch	30	–	ns
t_{LC}	Latch to READ / WRITE Control	40	–	ns
t_{RD}	Valid Data out Delay from READ Control	–	140	ns
t_{AD}	Address Stable To Data Out Valid	–	330	ns
t_{LL}	Latch Enable Width	70	–	ns
t_{RDF}	Data Bus Float after READ	0	80	ns
t_{CL}	READ / WRITE Control Latch Enable	10	–	ns
t_{CC}	READ / WRITE Control Width	200	–	ns
t_{DW}	Data in to WRITE Setup Time	100	–	ns
t_{WD}	Data in Hold Time After WRITE	0	–	ns
t_{RV}	Recovery Time Between Controls	200	–	ns
t_{WP}	WRITE to Port Output	–	300	ns
t_{PR}	Port Input Setup Time	50	–	ns
t_{RP}	Port Input Hold Time	10	–	ns
t_{SBF}	Strobe to Buffer Full	–	300	ns
t_{SS}	Strobe Width	150	–	ns
t_{RBE}	READ to Buffer Empty	–	300	ns
t_{SI}	Strobe to INTR On	–	300	ns
t_{RDI}	Strobe to INTR Off	–	300	ns
t_{PSS}	Port Setup Time to Strobe	20	–	ns
t_{PHS}	Port Hold Time after Strobe	100	–	ns
t_{SBE}	Strobe to Buffer Empty	–	300	ns
t_{WBF}	WRITE to Buffer Full	–	300	ns
t_{WI}	WRITE to INTR Off	–	300	ns
t_{TL}	TIMER-IN to $\overline{\text{TIMEROUT}}$ Low	–	300	ns
t_{TH}	TIMER-IN to $\overline{\text{TIMEROUT}}$ High	–	300	ns
t_{RDE}	Data Bus Enable from READ Control	10	–	ns
t_L	TIMER-IN Low Time	40	–	ns
t_H	TIMER-IN High Time	70	–	ns
t_{CYC}	CLK Cycle Period	200	–	ns
t_r, t_f	CLK Rise and Fall Time	–	30	ns
t_{WT}	WRITE to TIMER-IN (for writes which start counting)	200	–	ns

230589

Notes: 1. Test conditions CL = 150pF
2. Timing defining signal voltage are ;
Output High level = 2V, Low level = 0.8V

8. TIMING WAVEFORMS

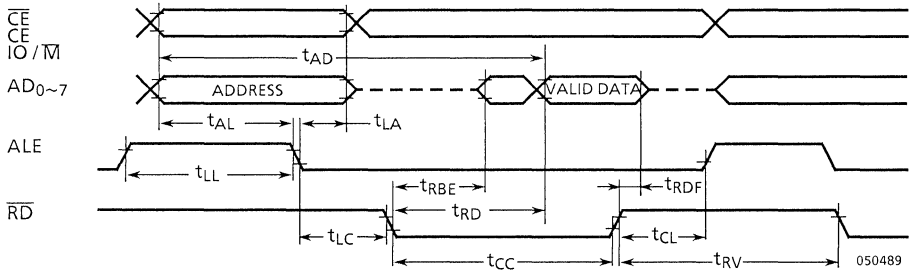


Figure 8.1 READ CYCLE

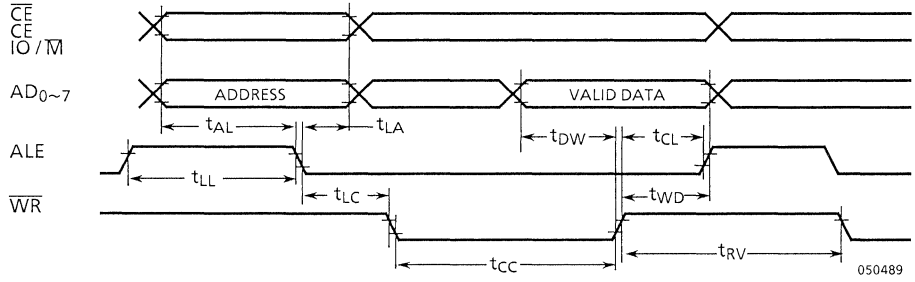


Figure 8.2 WRITE CYCLE

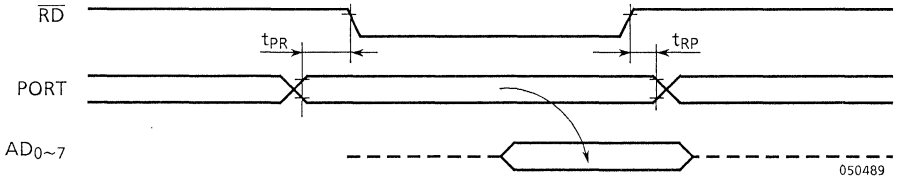


Figure 8.3 BASIC INPUT MODE

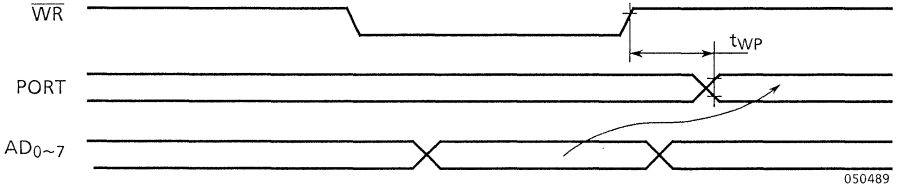


Figure 8.4 BASIC OUTPUT MODE

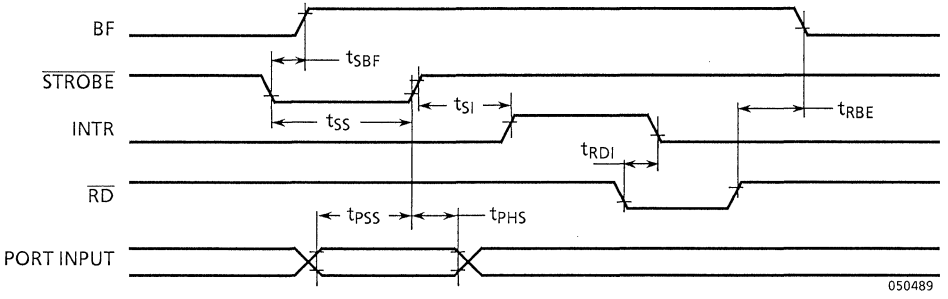


Figure 8.5 STROBED INPUT MODE

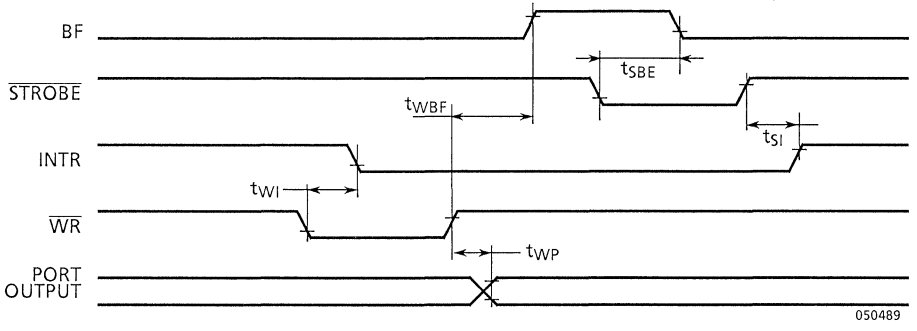


Figure 8.6 STROBED OUTPUT MODE

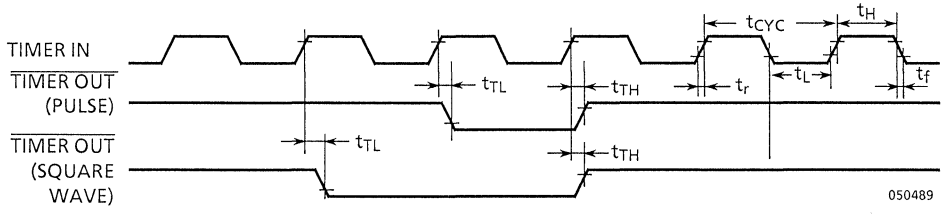
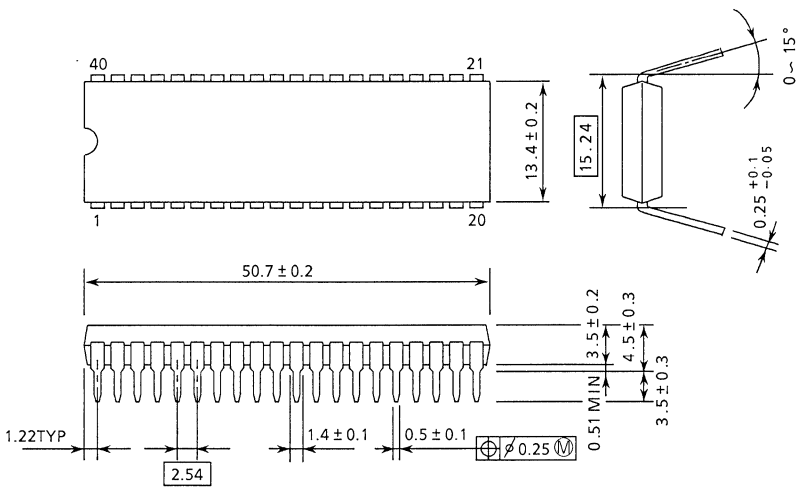


Figure 8.7 TIMER OUTPUT WAVEFORM

9. OUTLINE DRAWING (40Pins Plastic Package)

DIP40-P-600

Unit : mm



270289

Note : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoritical positions with respect to No. 1 and No. 40 leads.

TMP82C51AP-2 / TMP82C51AP-10
TMP82C51AM-2 / TMP82C51AM-10

PROGRAMMABLE COMMUNICATION INTERFACE

1. GENERAL DESCRIPTION

The TMP82C51A is the industry standard Universal Synchronous/Asynchronous Receiver/Transmitter (USART) that is fabricated using C-MOS silicon gate technology. The 82C51A is mainly used for 8-bit microcomputer extention system, which require serial data communications.

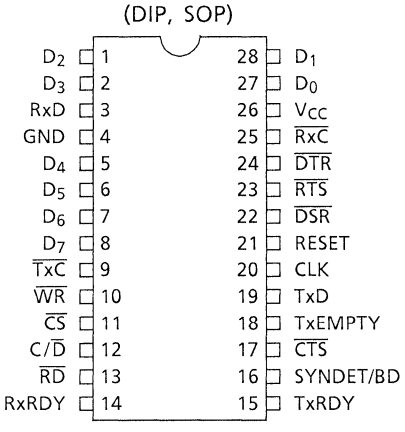
The TMP82C51AP-2/TMP82C51AP-10 is packaged in the 28 pin standard Dual Inline Package.

The TMP82C51AM-2/TMP82C51AM-10 is packaged in the 28 pin Small Out Line Package.

FEATURES

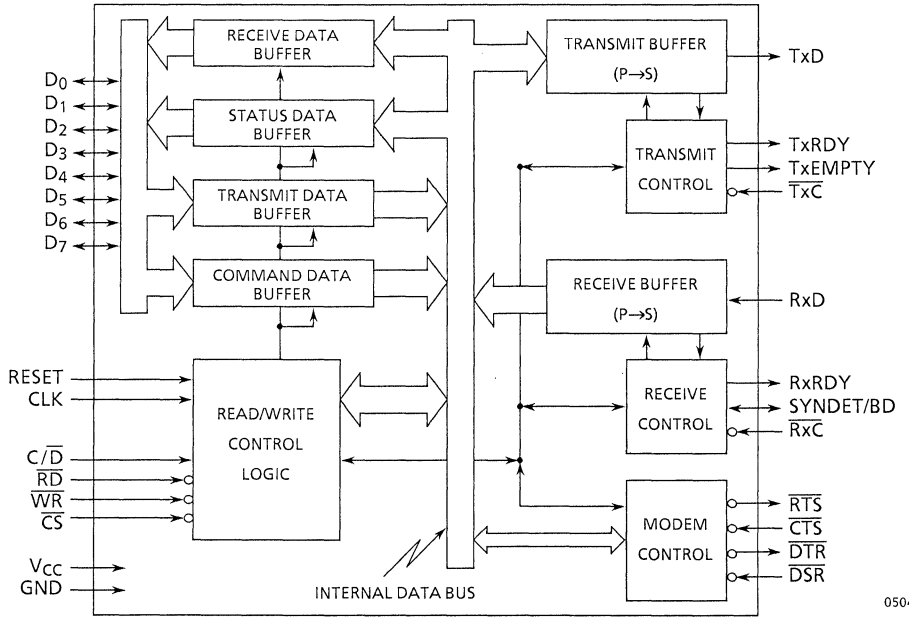
- Synchronous:
  5-8 Bit Characters
  Internal or External Character Synchronization
  Single or Double Character Synchronization (Inrternal)
  Automatic Sync Character(s) Insertion
- Asynchronous:
  5-8 Bit Characters
  Clock Rate -1, 16 or 64 Times Transfer Rate
  Break Character Generation
  1, 1.5 or 2 stop Bits
  False Start Bit Detection
  Automatic Break Detect and Handling
- Transfer Rate TMP82C51A-2 DC-104K bps TMP82C51A-10 DC-300K bps
- Full-Duplex, Double-Buffered, Transmitter and Receiver
- Error DetectionParity, Overrun and Framing.
- Single +5V Supply: 5V ±10%

2. PIN CONNECTIONS



050489

3. BLOCK DIAGRAM



050489

4. PIN NAMES AND PIN DESCRIPTIONS

4.1 INTERFACE SIGNALS TO CPU (MAIN SYSTEM)

- D0-D7 (Input/Output)

  This 3-state, bidirectional, 8-bit buffer is used to interface with the system Data Bus. Data is transmitted or received through the buffer upon execution of Input or Output Instructions of the CPU. Control Words, Command Words and Status Information are also transferred through the Data Bus Buffer.

- $\overline{WR}$ (Input)

  A "low" level signal on this input informs the 82C51A that the CPU is Writing Data or Control Words to the 82C51A.

- $\overline{RD}$ (Input)

  A "low" level signal on this input informs the 82C51A that the CPU is Reading Data or Status Information from the 82C51A.

- $\overline{CS}$ (Input)

  A "low" level signal on this input selects the 82C51A. No reading or writing operation will occur unless the device is selected. When $\overline{CS}$ is "high" the Data Bus is in the floating state and $\overline{RD}$ and $\overline{WR}$ have no effect on the chip.

- C/$\overline{D}$ (Input)

  This input signal, in conjuction with the $\overline{WR}$ and $\overline{RD}$ inputs, informs the 82C51A that the word on the Data Bus is either a Data Bus Character, Control Word or Status Information. A "high" level signal means Control or Status, a "low" level signal means Data.

C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$		
0	0	1	0	82C51A Receive DATA Buffer	→ DATA Bus
0	1	0	0	82C51A Transmit DATA Buffer	← DATA Bus
1	0	1	0	82C51A Status DATA Buffer	→ DATA Bus
X	1	1	0	DATA Bus is in floating state.	
X	X	X	1	DATA Bus is in floating state.	

- CLK (Input)

  The CLK input is used to generate internal device timing. No external input or output referenced to CLK, but the frequency of CLK must be greater than 30 times the Receiver or Transmitter Data Bit Rates ($\overline{RxC}$ or $\overline{TxC}$) in Synchronous Operation, and greater than 4.5 times the Receiver Data Bit Rates ($\overline{RxC}$ or Trransmiter, $\overline{TxC}$) in Asynchronous operation.

- RESET (Input)

  A "high" level signal on this input forces the 82C51A into an "Idle" mode. The device will remain "Idle" until a new set of Control Words is written into the 82C51A to program its functional definition. Minimum RESET pulse width is 6 tcy.

4.2 MODEM CONTROL SIGNALS

- $\overline{\text{DSR}}$ (Input)

  The $\overline{\text{DSR}}$ input signal is a general purpose, 1-bit inverting input port. Its condition can be tested by the CPU using a Status Read Operation. The $\overline{\text{DSR}}$ input is normally used to test MODEM conditions such as Data Set Ready signal.

- $\overline{\text{DTR}}$ (Output)

  The $\overline{\text{DTR}}$ output signal is a general purpose, 1-bit inverting output port. It can be set "low" by programming the appropriate bit in the Command Instruction Words. The $\overline{\text{DTR}}$ output signal is normally used for MODEM control such as Data Terminal Ready or Rate Select signal.

- $\overline{\text{RTS}}$ (Output)

  The $\overline{\text{RTS}}$ output signal is a general purpose, 1-bit inverting output port. It can be set "low" by pogramming the appropriate bit in the Command Instruction Word. The $\overline{\text{RTS}}$ output signal is normally used for MODEM control such as Request to Send signal.

- $\overline{\text{CTS}}$ (Input)

  A "low" level signal on this input enables the 82C51A to transmit serial data, if the Tx Enable Bit in the Command Byte is set to a "one" (TxEN=1). If either a Tx Enable off (TxEN=0) or $\overline{\text{CTS}}$ off (CTS=1) condition occurs while the Tx is in operaion, the Tx will transmit all the data in the USART, written prior to Tx Disable Command before shutting down.

4.3 TRANSMIT CONTROL SIGNALS

- $\overline{\text{TxC}}$ (Input)

  The transmitter Clock controls the rate at which the character is to be transmitted. In the Synchronous Transmission Mode, the Transfer Rate (1x) is equal to the $\overline{\text{TxC}}$ frequency. In Asynchronous Transmission Mode the transfer rate is a fraction of the actual $\overline{\text{TxC}}$ ferquency. A portion of the Mode Instruction selects this factor; it can be 1, 1/16 or 1/64 the $\overline{\text{TxC}}$.

For Example:
If Transfer Rate equals 110 bps,
$\overline{TxC}$ = 110 Hz (1x)
$\overline{TxC}$ = 1.76 kHz (16x)
$\overline{TxC}$ = 7.04 Hz (64x)
The falling edge of $\overline{TxC}$ shifts the serial data out of the 82C51A.

- TxD (Output)

This line is used to transmit serial data. Serial output data on TxD is changed from parallel data to serial data in accordance with the TxD line will be held in the marking state ('1' level) immediately on one of the followings.

- Master Reset
- Tx Disable (TxEN = 0)
- CTS signal is high ($\overline{CTS}$ = 1)
- TxEMPTY signal is high (TxEMPTY = 1)

- TxRDY (Output)

This output informs the CPU that the transmitter is ready to accept a Data Character. The TxRDY output pin can be used as an interrupt to the system, since it is masked by Tx Disable (TxEN = 0), or, for polled Operation, the CPU can check TxRDY using a Status Read Operation, TxRDY is automatically reset by the trailing edge of $\overline{WR}$ when a Data Character is loaded from the CPU. The TxRDY pin output status (TxRDY (pin)) is different from the TxRDY status bit status register (TxRDY (status bit)) as follows.

TxRDY (status bit) = (Transmit Data Buffer Empty)
TxRDY (pin) = (Transmit Data Buffer Empty) AND ($\overline{CTS}$ = 0) AND (TxEN = 1)

- TxEMPTY (Output)

The TxEMPTY output will go "high" when the 82C51A has no characters to send. It resets upon receiving a character from the CPU if the transmitter is enabled.

In Synchronous Mode, a "high" level signal on this output indicates that a Character has not been loaded and the SYNC Character or Characters are about to be or are being transmitted automatically as "fillers". TxEMPTY does not go "low" when the SYNC characters are being shifted out.

4.4 RECEIVE CONTROL SIGNALS

- $\overline{RxC}$ (Input)

  The Receiver Clock controls the rate at which the character is to be received. In Synchronous Mode, the Transfer Rate (1x) is equal to the actual frequency of $\overline{RxC}$. In Asynchronous Mode, the Transfer Rate is a fraction of the actual $\overline{RxC}$ frequency. A portion of the Mode Instruction selects this factor; 1, 1/16 or 1/64 the $\overline{RxC}$.

  For Example:

  If Transfere Rate equals 2400 bps,

  $\overline{RxC}$ = 2.4 kHz (1x)

  $\overline{RxC}$ = 38.4 kHz (16x)

  $\overline{RxC}$ = 153.6 kHz (64x)

  Data is sampled into the 82C51A on the rising edge of $\overline{RxC}$.

- RxD (Input)

  This line is used to receive the serial data. Serial input data on this line is changed to parallel data in accordance with the format specified by the Control Words, and then transfered to the Receive Data Buffer.

- RxRDY (Output)

  This output indicates that the 82C51A contains a Data Character that is ready to be input to the CPU. RxRDY can be connected to the interrupt structure of the CPU, or, for Polled Operation, the CPU can check the condition of RxRDY using Status Read Operation.

  Rx Enable off both masks and holds RxRDY in the Reset Condition.

- SYNDET/BD (Input/Output)

  This pin is used for SYNDET in Synchronous Mode and may used as either input or output, programmable through the Control Word. It is reset to output mode "low" upon RESET. When used as an Output (Internal Sync Mode), the SYNDET pin will go "high" to indicate that the 82C51A has located the SYNC Character in the Receive Mode. If the 82C51A is programmed to use Double Sync Characters then SYDET will go "high" in the middle of the last bit of the second SYNC Character. SYNDET is automatically reset upon a Status Read Operation. When used as an Input (External Sync Mode), a positive going signal will cause the 82C51A to start assembling Data Characters on the rising edge of the next $\overline{RxC}$.

In Asynchronous Mode this pin is used BD.

This output will go "high" whenever the receiver remains "low" through two consecutive Stop Bit Sequences (including the Start Bits, Data Bits, and parity bits). Break Detect may also be read as a Status Bit.

It is reset only upon a Master Chip Reset or Rx Data returning to a "one" state.

4.5 POWER SUPPLY

- VCC (Power)

  +5 Volt supply

- GND (Power)

  0 Volt supply

5. ELECTRICAL CHARACTERISTICS

5.1 MAXIMUM RATINGS

Symbol	Item	Rating
V_{CC}	Power Supply Voltage (with respect to GND)	– 0.5V to 7.0V
V_{IN}	Input Voltage (with respect to GND)	– 0.5V to V_{CC} + 0.5
V_{OUT}	Output Voltage (with respect to GND)	– 0.5V to V_{CC} + 0.5
P_D	Power Dissipation	250mW
T_{SOLDER}	Soldering Temperature (10 sec)	260°C
$T_{STG.}$	Storage Temperature	– 65°C to 150°C
$T_{OPR.}$	Operating Temperature	– 40°C to 85°C

050489

5.2 D.C CHARACTERS

Topr = – 40°C to + 85°C, VCC = + 5V ± 10%, GND = 0V, Unless otherwise noted.

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Units
V_{IL}	Input Low Voltage		– 0.5	–	0.8	V
V_{IH}	Input High Voltage		2.2	–	VCC + 0.5	V
V_{OL}	Output Low Voltage	IOL = 2.2mA	–	–	0.45	V
V_{OH1}	Output High Voltage	IOH = – 400μA	2.4	–	–	V
V_{OH2}	Output High Voltage	IOH = – 100μA	Vcc – 0.8	–	–	V
IOFL	Output Leak Current	0.45V ≦ VOUT ≦ VCC	–	–	± 10	μA
IIL	Input Leak Current	0.45V ≦ VOUT ≦ VCC	–	–	± 10	μA
ICC1	Power Supply Current (AP-2/AM-2, 5MHz)	tcyc = 200ns Vin = 4.8V/0.2V	–	1.2	5.0	mA
	Power Supply Current (AP-8/AM-8, 8MHz)	tcyc = 125ns Vin = 4.8V/0.2V	–	2.0	10.0	mA
	Power Supply Current (AP-10/AM-10, 10MHz)	tcyc = 100ns Vin = 4.8V/0.2V	–	2.5	15	mA
ICC2	Power Supply Current (Standby Mode)	STOP All Clocks VCC = 5V, $\overline{CS}$ = 1 Vin = 4.8V/0.2V	–	0.5	10.0	μA

050489

5.3 AC CHARACTERISTICS

Topr = −40°C to 85°C, VCC = 5V ± 10%, GND = 0V, Unless otherwise noted.

5.3.1 Bus Read Cycle Timing Note 1)

Symbol	Parameter	Test Conditions	AP-2 / AM-2		AP-10/AM-10		Units
			Min.	Max.	Min.	Max.	
t_{AR}	$\overline{CS}$, C/$\overline{D}$ Set-up Time for $\overline{RD}$		0	–	0	–	nS
t_{RA}	$\overline{CS}$, C/$\overline{D}$ Hold Time for $\overline{RD}$		0	–	0	–	nS
t_{RR}	$\overline{RD}$ Pulse Width		150	–	120	–	nS
t_{RD}	Data Delay Time for $\overline{RD}$ Note 2)	CL = 150pF Note 3)	–	140	–	100	nS
t_{DF}	Data Hold Time for $\overline{RD}$		10	80	10	50	nS

230589

5.3.2 Bus Write Cycle Timing (Note 1)

Symbol	Parameter	Test Conditions	AP-2 / AM-2		AP-10/AM-10		Units
			Min.	Max.	Min.	Max.	
t_{AW}	$\overline{CS}$, C/$\overline{D}$ Set-up Time for $\overline{WR}$		0	–	0	–	nS
t_{WA}	$\overline{CS}$, C/$\overline{D}$ Hold Time for $\overline{WR}$		0	–	0	–	nS
t_{WW}	$\overline{RD}$ Pulse Width		150	–	120	–	nS
t_{DW}	Data Set-up Time for $\overline{WR}$		100	–	70	–	nS
t_{WD}	Data Hold Time for $\overline{WR}$		0	–	0	–	nS
t_{RV}	Recovery Time Between Write	Note 4)	6	–	6	–	tcyc

230589

5.3.3 Other Timings

Symbol	Parameter	Test Conditions	AP-2 / AM-2		AP-10 / AM-10		Units
			Min.	Max.	Min.	Max.	
tcyc	Clock Period Note 5), 6)		200	–	100	–	nS
tH	Clock High Level Wdith		80	–	40	–	nS
tL	Clock Low Level Wdith		50	–	30	–	nS
tR, tF	Clock Rise, Fall Time		–	20	–	10	nS
tDTx	TxD Delay Time from Falling Edge of TxC		–	1	–	0.5	us
fTx	Transmitter Input Clock Frequency	1xBaud Rate	DC	104	DC	300	kHz
		16xBaud Rate	DC	528	DC	2000	
		64xBaud Rate	DC	832	DC	2000	
tTPH	Transmitter Input Clock Low Level Width	1xBaud Rate	12	–	12	–	tcyc
		16x, 64x, Baud Rate	1	–	1	–	
tTPL	Transmitter Input Clock High Level Width	1xBaud Rate	15	–	15	–	tcyc
		16x, 64x, Baud Rate	3	–	3	–	
fRx	Receiver Input Clock Frequency	1xBaud Rate	DC	104	DC	300	kHz
		16xBaud Rate	DC	528	DC	2000	
		64xBaud Rate	DC	832	DC	2000	
tRPH	Receiver Input Clock High Level Width	1xBaud Rate	12	–	12	–	tcyc
		16x, 64x, Baud Rate	1	–	1	–	
tRPH	Receiver Input Clock Low Level Width	1xBaud Rate	15	–	15	–	tcyc
		16x, 64x, Baud Rate	3	–	3	–	
tTxRDY	TxRDY Pin Delay Time from Center of Last Bit		–	14	–	14	tcyc
tTxRDY CLEAR	TxRDY Clear Delay Time from Leading Edge of $\overline{WR}$		–	400	–	150	ns
tRxRDY	TxRDY Pin Delay Time from Center of Last Bit		–	26	–	26	tcyc
tRxRDY CLEAR	TxRDY Clear Delay Time from Leading Edge of $\overline{WR}$		–	400	–	150	ns
tIS	Internal SYNDET Delay Time from Rising Edge of RxC		–	26	–	26	tcyc
tES	External SYNDET Set-up Time for Falling Edge of $\overline{RxC}$		–	18	–	18	tcyc
tTx EMPTY	TxEMPTY Delay Time from Center of Last Bit		–	20	–	20	tcyc
tWC	Control Delay Time from Rising Edge of $\overline{WR}$ (TxEN, $\overline{DTR}$ $\overline{RTS}$)		–	8	–	8	tcyc
tCR	$\overline{DSR}$, $\overline{CTS}$ Set-up Time for $\overline{RD}$		20	–	20	–	tcyc

230589

Notes:

1) AC Test Condition: Output measuring points VOH = 2.2V, VOL = 0.8V
Input supply level VIH = 2.4V, VIL = 0.45V

2) Assumes that Address is valid before the falling edge $\overline{RD}$.

3) CL means load capacitance.

4) This recovery time is defind only for Mode Intialization.
Write Data is allowed only when TxRDY = 1. Recovery Time between Writes for Asynchronous Mode is 8 tcyc and for Synchronous Mode is 16 tcyc.

5) The $\overline{TxC}$ and $\overline{RxC}$ frequencies have the following limitations with respect to CLK:
For 1x Transfer Rate, fTx or fRx < 1 (30 tcyc)
For 16x and 64x Transfer Rate, fTx or fRx ≦ 1 (4.5 tcyc)

6) Minimum Reset Pulse Width is 6 tcyc. System Clock must running during Reset.

7) Status up data can have a maximum delay of 28 clock periods from the event affecting the status.

6. TIMING WAVEFORMS

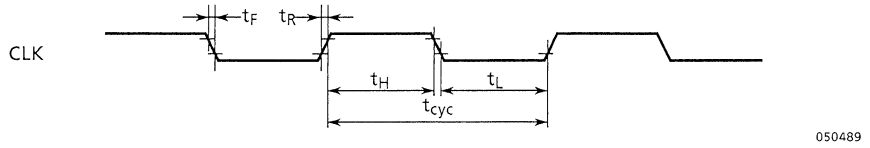


Figure 6.1 System Clock

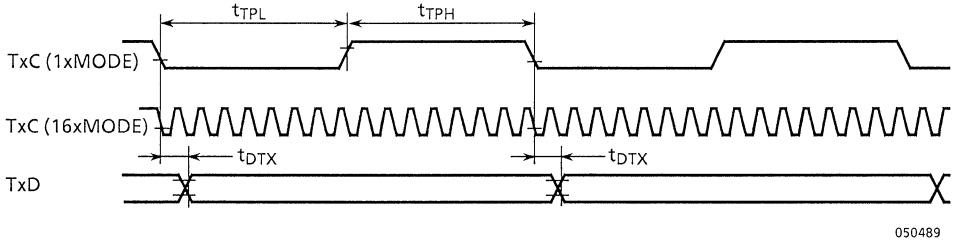


Figure 6.2 Transmitter Clock and Data

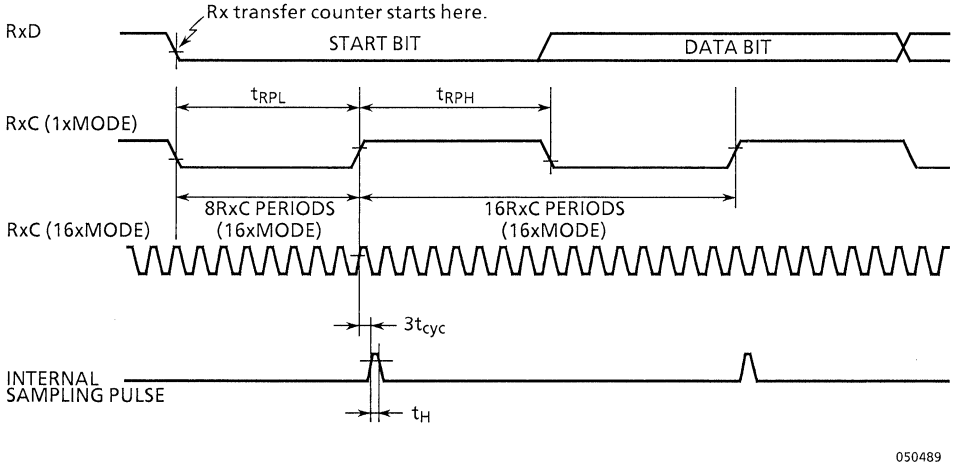


Figure 6.3 Receiver Clock and Data

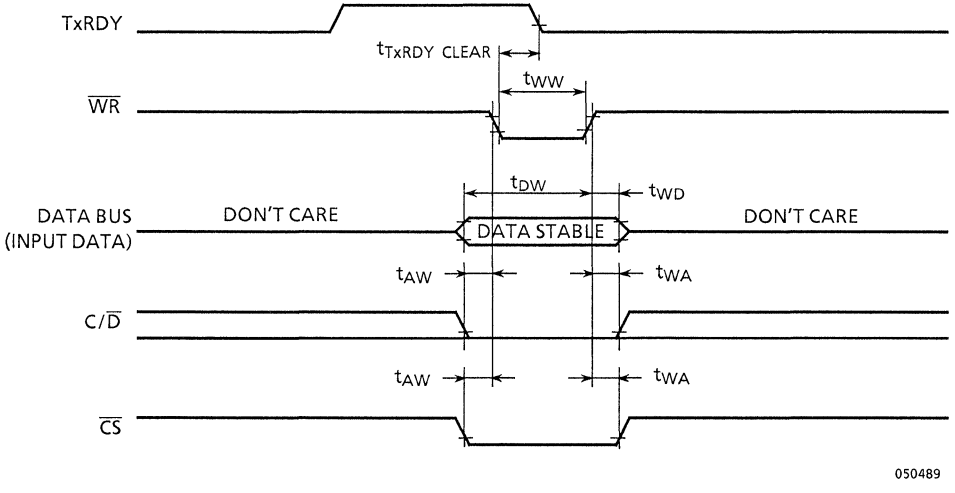


Figure 6.4 Write Data Cycle (MPU → 82C51A)

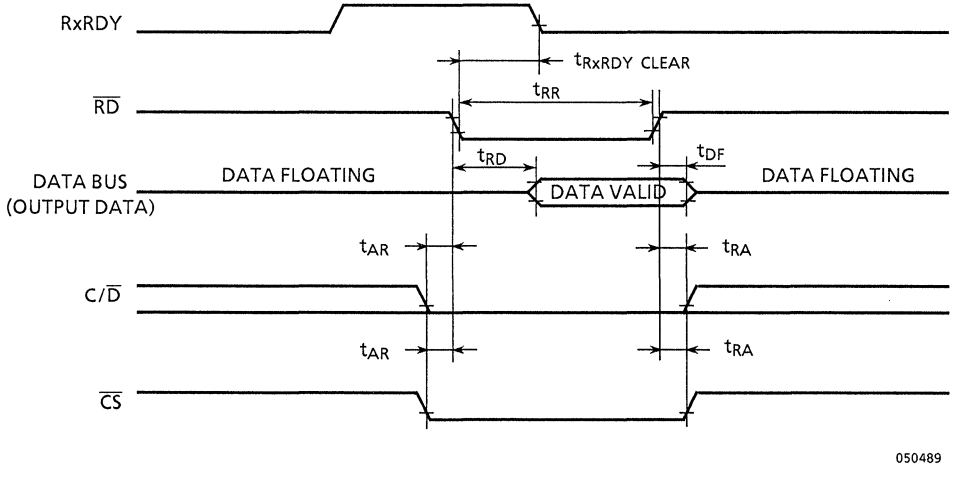


Figure 6.5 Read Data Cycle (82C51A → MPU)

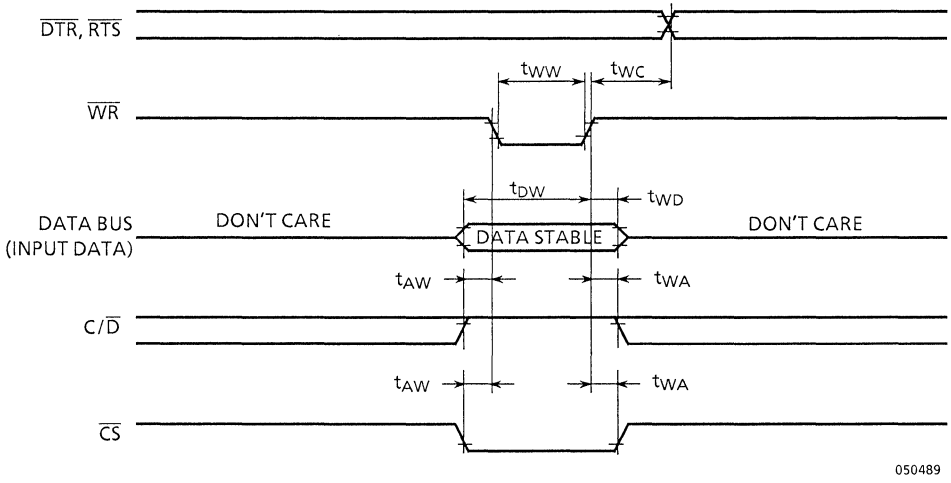


Figure 6.6 Write Control or Output Port Cycle (MPU → 82C51A)

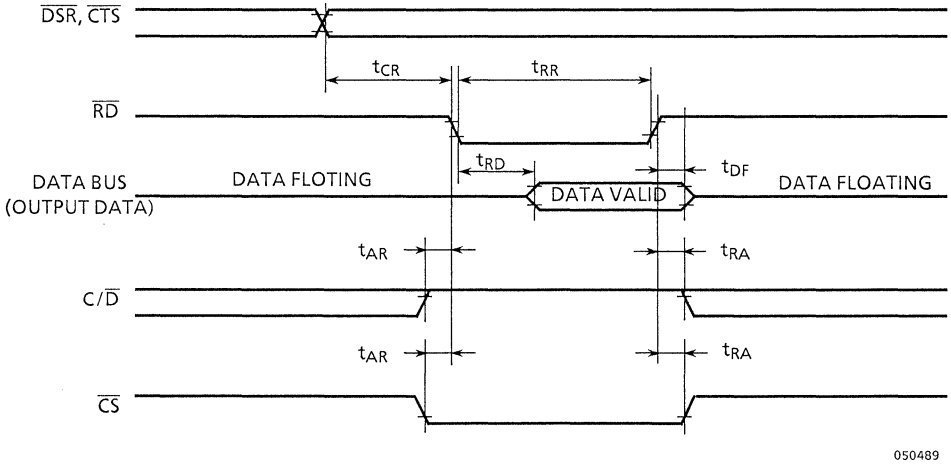
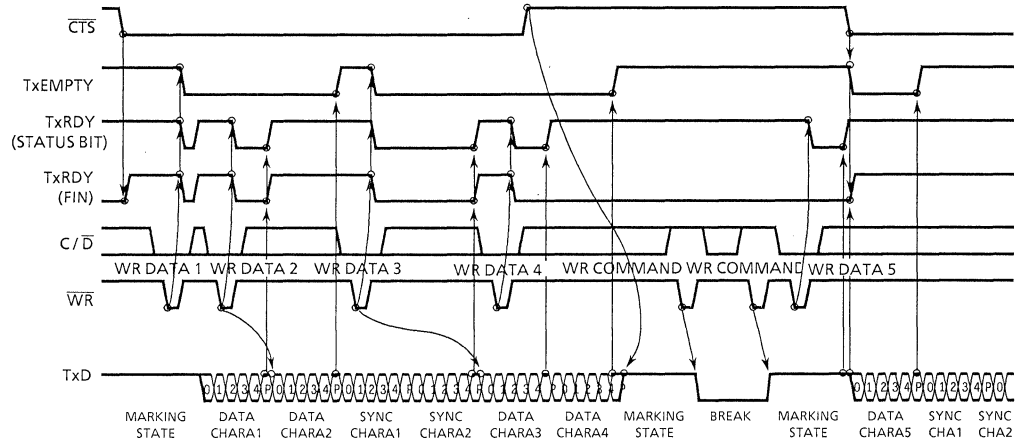
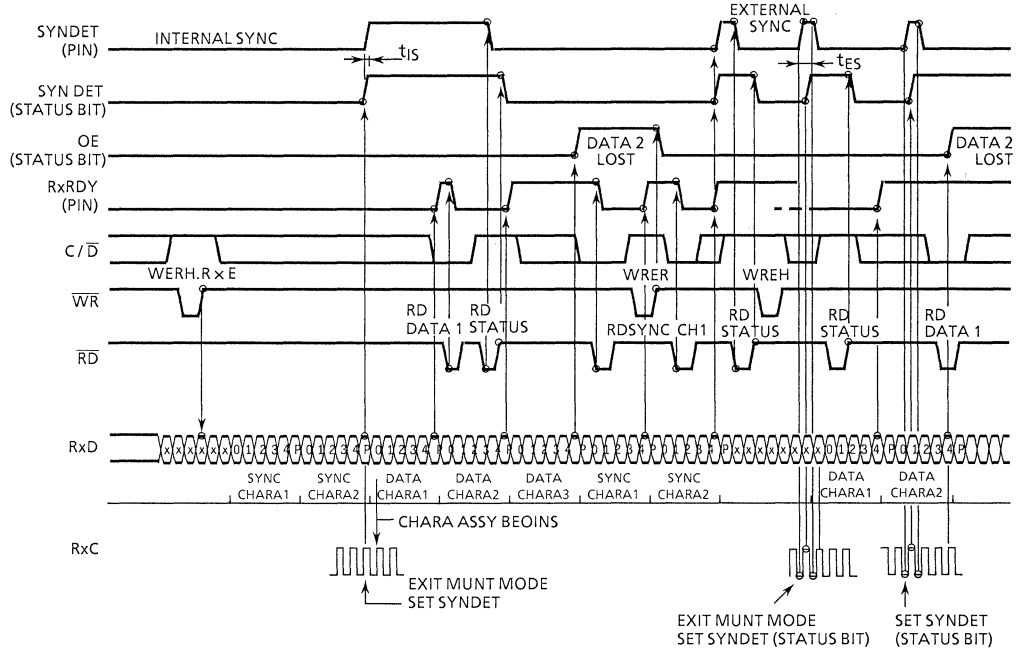


Figure 6.7 Read Control or Input Port Cycle (82C51A → MPU)



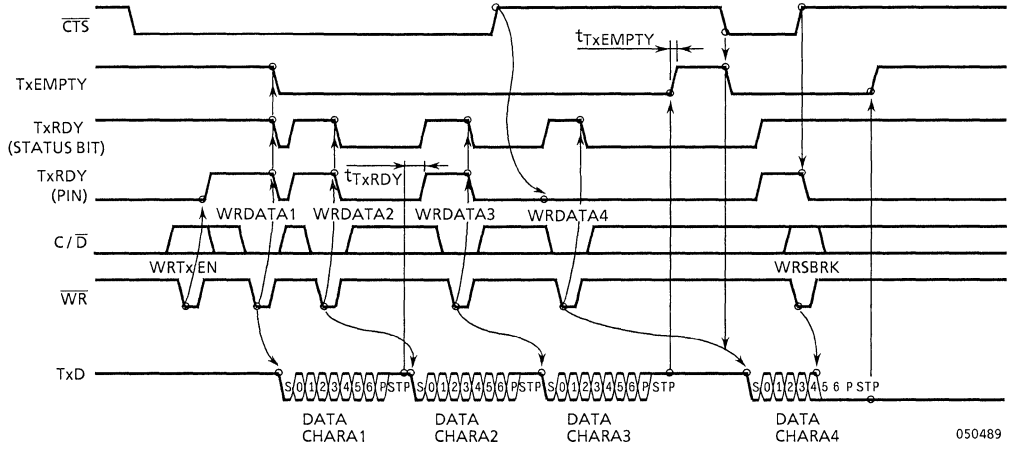
EXAMPLE FORMAT : 5 BIT CHARACTER WITH PARITY 2 SYNC CHARACTERS. 050489

Figure 6.8 Transmitter Control and Flag Timing (SYNC Mode)



EXAMPLE FORMAT : 5 BIT CHARACTERS WITH PARITY 2 SYNC CHARACTERS. 050489

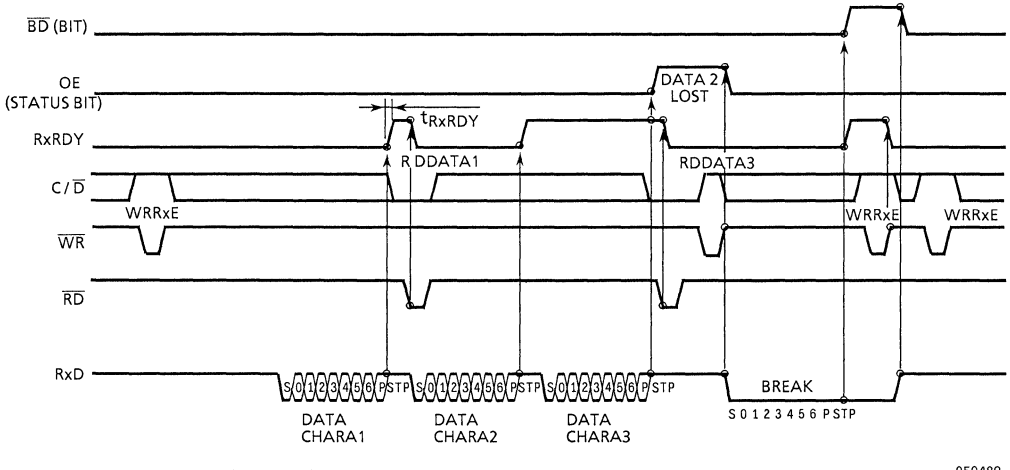
Figure 6.9 Receiver Control and Flag Timimg (SYNC Mode)



EXAMPLE FORMAT : 7 BIT CHARACTER & 2 STOP BITS.

NOTE: { TxRDY (PIN) = (Transmit Data Buffer is empty) . (TxEN = 1) . ($\overline{CTS}$ = 0)
TxRDY (STATUS BIT) = (Transmit Data Buffer is empty)

Figure 6.10 Transmitter Control and Flag Timimg (ASYNC Mode)



EXAMPLE FORMAT : 7 BIT CHARACTER WITH PARITY & 2 STOP BITS.

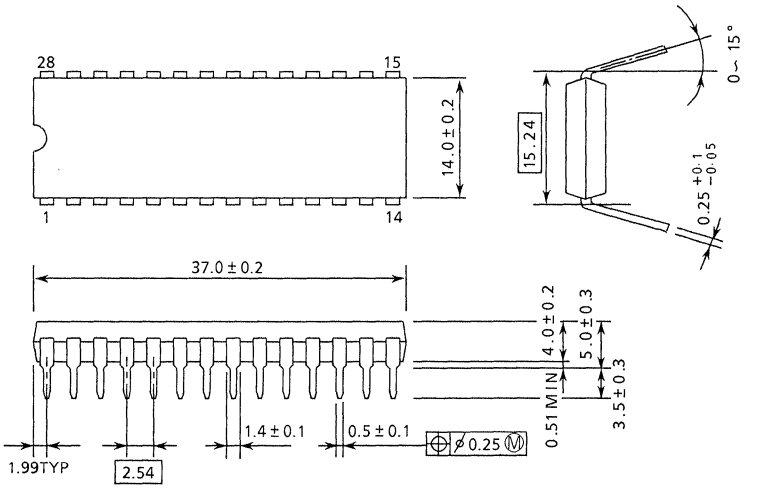
Figure 6.11 Receiver Control and Flag Timing (ASYNC Mode)

7. OUTLINE DRAWING

7.1 DIP

DIP28-P-600

Unit : mm



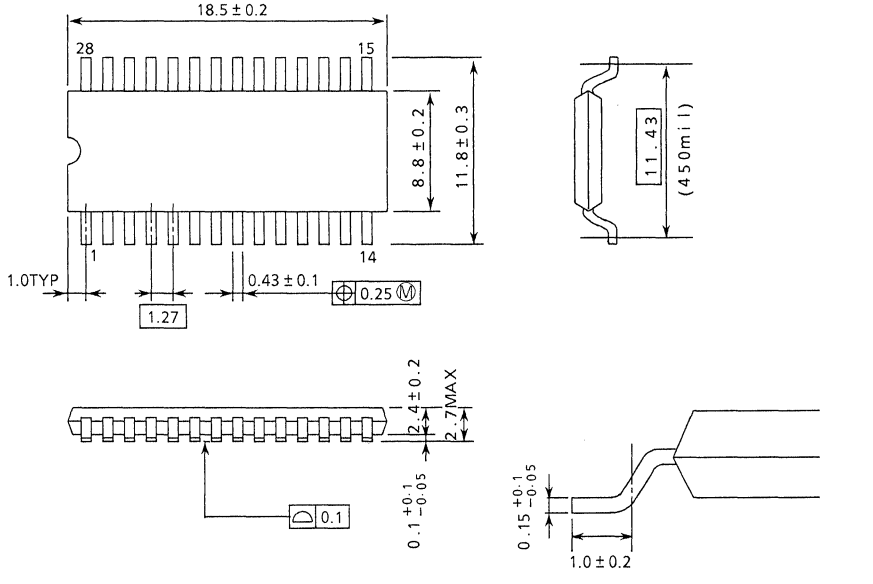
270289

Note : Lead pitch is 2.54mm and to lerance is ±0.25mm against theoretical center of each lead that is obtained on the basis of No.1 and No.28 leads.

7.2 SOP

SOP28-P-450

Unit : mm



270289

Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

PROGRAMMABLE COMMUNICATION INTERFACE

TMP8251AP

1. GENERAL DESCRIPTION

The TMP8251AP is the industry standard Universal Synchronous/Asynchoronous Receiver/Transmitter (USART) that is fabricated using N-channel silicon gate MOS technology.

The TMP8251A is mainly used for 8-bit microcomputer extension systems, which require serial data communications.

The TMP8251AP is packaged in the 28pin standard Dual Inline package.

FEATURES

- Synchoronous:

  5-8 Bit Characters
  Internal or External Character Synchronization
  Single or Double Character Synchronization (Internal)
  Automatic Sync Insertion

- Asynchronous:

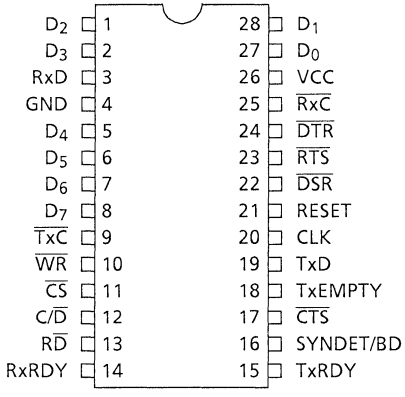
  5-8 Bit Characters
  Clock Rate - 1, 16 or 64 Times Transfer Rate
  Break Character Generation
  1, 1.5, or 2 Stop Bits
  False Start Bit Detection
  Automatic Break Detect and Handling

- Transfer Rate DC to 64K bps (Synchronous)

  DC to 19.6K bps (Asynchronous)

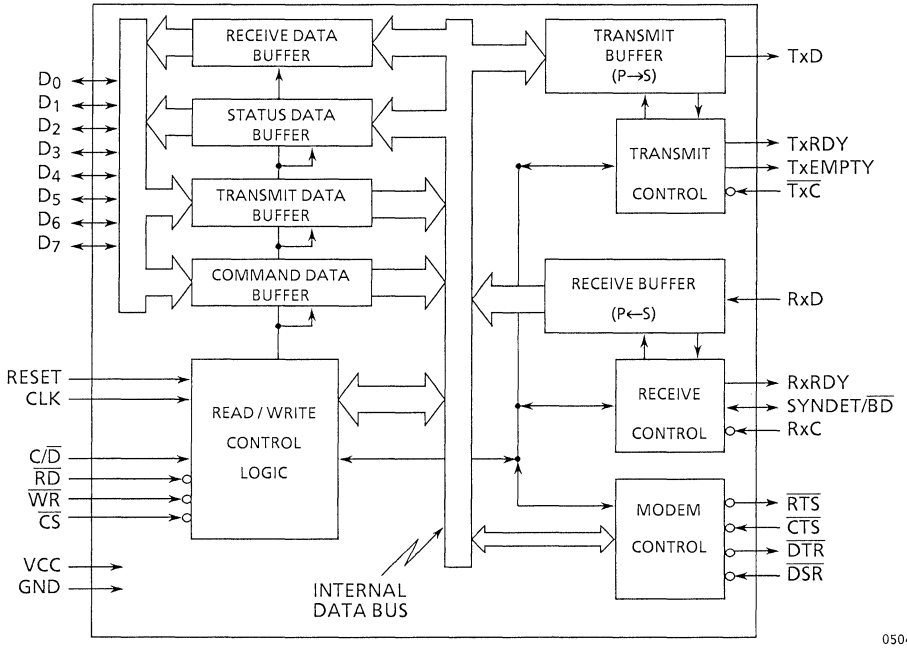
- Full-Duplex, Double-Buffered, Transmitter and Receiver
- Error DetectionParity, Overrun and Framing
- Single +5V Supply
- Compatible with Intel's 8251A/S2657

2. PIN CONNECTIONS (TOP VIEW)



050489

3. BLOCK DIAGRAM



050489

4. PIN NAMES AND PIN DESCRIPTIONS

4.1 INTERFACE SIGNALS TO MPU (MAIN SYSTEM)

- $D_0 \sim D_7$ (Input/Output)

  This 3-state bidirectional, 8-bit buffer is used to interface the 8251A to the system Data Bus. Data is transmitted or received through the buffer upon execution of Input or Output Instructions of the MPU. Control Words, Command Words and Status Information are also transffered through the Data Bus Buffer.

- $\overline{WR}$ (Input)

  A "low" level signal on this input informs the 8251A that the MPU is Writing Data or Control Words to the 8251A.

- $\overline{RD}$ (Input)

  A "low" level signal on this input informs the 8251A that the MPU is Reading Data or Status Information from the 8251A.

- $\overline{CS}$ (Input)

  A "low" level signal on this input selects the 8251A. No reading or writing operation will occur unless the device is selected. When $\overline{CS}$ is "high" the Data Bus is in the floating state and $\overline{RD}$ and $\overline{WR}$ have no effect on the chips.

- $C/\overline{D}$ (Input)

  This input signal, in conjunction with the $\overline{WR}$ and $\overline{RD}$ inputs, informs the 8251A that the word on the Data Bus is either a Data Character, Control Word or Status Information. A "high" level signal means Control or Status, a "low" level signal means Data.

$C/\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	
0	0	1	0	8251A Recieve DATA Buffer → Data Bus
0	1	0	0	8251A Transmit DATA Buffer ← Data Bus
1	0	1	0	8251A Status DATA Buffer → Data Bus
1	1	0	0	8251A Command DATA Buffer ← Data Bus
×	1	1	0	DATA Bus is in floating state.
×	×	×	1	

- CLK (Input)

  The CLK input is used to generate internal device timing. No external input or output is referenced to CLK, but the frequency of CLK must be greater than 30 times the Receiver or Transmitter Data Bit Rates ($\overline{RxC}$ or $\overline{TxC}$) in Synchronous Operation, and greater than 4.5 times the Receiver Data Bit Rated ($\overline{RxC}$) in Asynchronous Operation.

- RESET (Input)

  A "high" level signal on this input forces the 8251A into an "Idle" mode. The device will remain at "Idel" untill a new set of Control Words is written into the 8251A to program its functional definition. Minimum RESET pulse width is 6 tcy.

4.2 MODEM CONTROL SIGNALS

- $\overline{DSR}$ (Input)

  The $\overline{DSR}$ input signal is a general purpose, 1-bit inverting input port. Its condition can be tested by the MPU using a Status Read Operation. The $\overline{DSR}$ input is normlaly used to test MODEM conditions such as Data Set Ready signal.

- $\overline{DTR}$ (Output)

  The $\overline{DTR}$ output signal is a general purpose, 1-bit inverting output port. It can be set "low" by programming the appropriate bit in the Command Instsruction Word. The $\overline{DTR}$ output signal is normally used for MODEM control such as Data Terminal Ready or Rate Select signal.

- $\overline{RTS}$ (Output)

  The $\overline{RTS}$ output signal is a general purpose, 1-bit inverting output port. It can be set "low" by programming the appropriate bit in the Command Instruction Word. The $\overline{RTS}$ output signal is normally used for MODEM control such as Request to Send signal.

- $\overline{CTS}$ (Input)

  A "low" level signal on this input enables the 8251A to transmit serial data, if the Tx Enable Bit in the Command Byte is set to a "one" (TxEN = 1). If either a Tx Enable off (TxEN = 0) or CTS off ($\overline{CTS}$ = 1) condition occurs while the Tx is in operation, the Tx will transmit all the data in the USART, written prior to Tx Disable Command before shutting down.

4.3 TRANSMIT CONTROL SIGNALS

- $\overline{TxC}$ (Input)

  The Transmitter Clock Controls the rate at which the character is to be transmitted. In the Synchronous Transmission Mode, the transfer rate (1x) is equal to the $\overline{TxC}$ frequency. In Asynchronous Transmission Mode, the transfer rate is a fraction of the actual $\overline{TxC}$ frequency. A poriton of the Mode Instruction selects this factor; it can be 1, $^1/_{16}$ or $^1/_{64}$ the $\overline{TxC}$.

  For Example:
  If transfer rate equals 110 bps,
  $\overline{TxC}$ = 110 Hz (1x)
  $\overline{TxC}$ = 1.76 KHz (16x)
  $\overline{TxC}$ = 7.04 KHz (64x)

  The falling edge of TxC shifts the serial data out of the 8251A.

- TxD (Output)

  This line is used to transmit the serial data. Serial output data on TxD is changed from paralled data to serial data in accordancae with the format specified by the Control Words.

  TxD line will be held in the marking state ('1' level) immediately on one of the following.s
  - Master Reset
  - CTS signal is high ($\overline{CTS}$ = 1)
  - Tx Disable (TxEN = 0)
  - TxEMPTY signal is high (TxEMPTY = 1)

- TxRDY (Output)

  This output informs the MPU that the transmitter is ready to accept a Data Character. The TxRDY output pin can be used as an interrupt to the system, since it is masked by Tx Disable (TxEN = 0), or, for polled Operation, the MPU can check TxRDY using a Status Read Operaiton. TxRDY is automatically reset by the trailing edge of $\overline{WR}$ when a Data Character is loaded from the MPU. The Tx RDY pin output status (Tx RDY (pin)) is different from the TxRDY status bit status (TxRDY (status bit)) as follows.

  TXRDY (status bit) = (Transmit Data Buffer Empty)

  TxRDY (pin) = (Transmit Data Buffer Empty) · (CTS = 0) · (TxEN = 1)

- TxEMPTY (Output)

  The TxEMPTY output will go "high" when the 8251A has no characters to send. It resets upon receiving as character from the MPU if the transmitter is enabled.

In Synchronous Mode, a"high" level signal on this output indicates that a Character has not been loaded and the SYNC Character or Characters are about to be or are being transmitted automatically as "fillers". Tx EMPTY does not go "low" when the SYNC characters are being shifted out.

4.4 RECEIVE CONTROL SIGNLAS

- $\overline{\text{RxC}}$ (Input)

  The Receiver Clock controls the rate at which the character is to be received. In Synchronous Mode, the Transfer Rate (1x) is equal to the actual frequency of $\overline{\text{RxC}}$. In Asynchronous Mode, the Transfer Rate is a fraction of the actual $\overline{\text{RxC}}$ frequency. A portion of the Mode Instruciton selects this factor; 1, $^1/_{16}$ or $^1/_{64}$ the $\overline{\text{RxC}}$.

  For Example:
  if Transfer Rate equals 2400 bps,
  $\overline{\text{RxC}}$ = 2.4 KHz (1x)
  $\overline{\text{RxC}}$ = 38.4 KHz (16x)
  $\overline{\text{RxC}}$ = 153.6 KHz (64x)

  Data is sampled into the 8251A on the rising edge of $\overline{\text{RxC}}$.

- RxD (Input)

  This line is used to receive the serial data. Serial input data on this line is changed to parallel data in accordance with the format specified by the Control Words, and then transfered to the Receive Data Buffer.

- RxRDY (Output)

  This output indicates that the 8251A contains a Data Character that is ready to be input to the MPU. RxRDY can be connected to the interrupt structure of the MPU, or, for Polled Operation, the MPU can check the condition of RxRDY using a Status Ready Operaiton.

  Rx Enable off both masks and holds RxRDY in the Reset Condition.

- SYNDET/BD (Input/Output)

  This pin is used for SYNDET in Synchronous Mode and may be used as eigher input or ourput, programmable through the Contorl Word. It is reset to output mode "low" upon RESET. When used as an Output (Internal Sync Mode), the SYNDET pin will go "high" to indicate that the 8251A is programmed to use SYNC Character in the Receive Mode. If the 8251A is programmed to use Double Sync Characters then SYNDET will go "high" in the middle of the last bit of the second SYNC Character. SYNDET is automatically reset upon a Status Read Operation. When used as an Input (External Sync Mode), a positive going signal will cause the 8251A to start assembling Data Characters on the rising edge of the next $\overline{RxC}$.

  In Asynchronous Mode this pin is used for BD. This output will go "high" whenever the receiver remains "low" through two consecutive Stop Bit Sequences (including the Start Bits, Data Bits, and Parity Bits). Break Detect may also be read as a Status Bit. It is reset only upon a Master Chip Reset or Rx Data returning to a "one" state. But, if the Rx data returns to a "one" State during the last bit of the next character after the Break, Break detect does not always reset.

4.5 POWER SUPPLY

- V_{CC} (Power)
  +5 Volt supply
- GND (Power)
  0 Volt supply

5. ELECTRICAL CHARACTERISTICS

5.1 MAXIMUM RATINGS

SYMBOL	ITEM	RATING
V_{CC}	Power Supply Voltage (with respect to GND)	– 0.5V to 7.0V
V_{IN}	Input Voltage (with respect to GND)	– 0.5V to 7.0V
V_{OUT}	Output Voltage (with respect to GND)	– 0.5V to 7.0V
P_D	Power Dissipation (Ta = 70°C)	1W
T_{solder}	Soldering Temperature (10 sec)	260°C
$T_{stg.}$	Storage Temperature	– 55°C to 150°C
$T_{opr.}$	Operating Temperature	0°C to 70°C

050489

5.2 D.C. CHARACTERISTICS

T_{opr} = 0°C to 70°C, V_{CC} = 5V ± 5%, GND = 0V, Unless otherwise noted.

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		– 0.5	–	0.8	V
V_{IH}	Input High Voltage		2.2	–	V_{CC}	V
V_{OL}	Output Low Voltage	I_{OL} = 2.2mA	–	–	0.45	V
V_{OH}	Output High Voltage	I_{OH} = – 400μA	2.4	–	–	V
I_{OFL}	Output Leak Current	$0.45V \leqq V_{OUT} \leqq V_{CC}$	–	–	± 10	μA
I_{IL}	Input Leak Current	$0.45V \leqq V_{IN} \leqq V_{CC}$	–	–	± 10	μA
I_{CC}	Power Supply Current	All Outputs = "High"	–	–	100	mA

050489

5.3 A.C CHARACTERISTICS

T_{opr} = 0°C to 70°C, V_{CC} = 5V ± 5%, GND = 0V, Unless otherwise noted.

5.3.1 Bus Read Cycle Timing Note 1)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
t_{AR}	$\overline{CS}$, C/$\overline{D}$ Set-up Time for $\overline{RD}$		50	–	–	ns
t_{RA}	$\overline{CS}$, C/$\overline{D}$ Hold Time for $\overline{RD}$		50	–	–	ns
t_{RR}	$\overline{RD}$ Pulse Width		250	–	–	ns
t_{RD}	Data Delay Time for RD Note 2)	C_L = 150pF Note 3)	–	–	250	ns
t_{DF}	$\overline{CS}$, C/$\overline{D}$ Set-up Time for $\overline{RD}$		10	–	100	ns

050489

5.3.2 Bus Write Cycle Timing Note 1)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
t_{AW}	$\overline{CS}$, C/$\overline{D}$ Set-up Time for $\overline{WR}$		50	–	–	ns
t_{WA}	$\overline{CS}$, C/$\overline{D}$ Hold Time for $\overline{WR}$		50	–	–	ns
t_{WW}	$\overline{WR}$ Pulse Width		250	–	–	ns
t_{DW}	Data Set Up Time for $\overline{WR}$		150	–	–	ns
t_{WD}	Data Hold Time for $\overline{WR}$		50	–	–	ns
t_{RV}	Recovery Time between WRITES	Note 4)	6	–	–	t_{cyc}

050489

5.3.3 Other Timing

SYMBOL	PARAMETER		MIN.	TYP.	MAX.	UNIT	
t_{cyc}	Clock Period Note 5), 6)		320	–	1350	ns	
t_H	Clock High Level Width		140	–	$t_{cyc}-90$	ns	
t_L	Clock Low Level Width		90	–	–	ns	
t_R, t_F	Clock Rise and Fall Time		–	–	20	ns	
t_{DTx}	TxD Delay Time from Falling Edge of $\overline{TxC}$		–	–	1	μs	
f_{Tx}	Transmitter Input Clock Frequency	1x, 64x Baud Rate	DC	–	64	kHz	
		16x Baud Rate	DC	–	310		
		64x Baud Rate	DC	–	615		
t_{TPH}	Transmitter Input Clock High Level Width	1x Baud Rate	12	–	–	t_{cyc}	
		16, 64x Baud Rate	1	–	–		
t_{TPL}	Transmitter Input Clock Low Level Width	1x Baud Rate	15	–	–	t_{cyc}	
		64x Baud Rate	3	–	–		
f_{Rx}	Transmitter Input Clock Frequency	1x Baud Rate	DC	–	64	kHz	
		16, 64x Baud Rate	DC	–	310		
		64x Baud Rate	DC	–	615		
t_{RPH}	Transmitter Input Clock High Level Width	1x Baud Rate	12	–	–	t_{cyc}	
		64x Baud Rate	1	–	–		
t_{RPL}	Transmitter Input Clock Low Level Width	1x Baud Rate	15	–	–	t_{cyc}	
		64x Baud Rate	3	–	–		
t_{TxRDY}	TxRDY Pin Delay Time from Center of Last Bit		–	–	8	t_{cyc}	Note 7
$t_{RxRDY\ CLEAR}$	TxRDY Clear Delay Time from Trailing Edge of $\overline{WR}$		–	–	6	t_{cyc}	Note 7
t_{RxRDY}	RxRDY Pin Delay Time from Center of Last Bit		–	–	24	t_{cyc}	Note 7
$t_{RxRDY\ CLEAR}$	RxRDY Clear Delay Time from Leading Edge of $\overline{RD}$		–	–	6	t_{cyc}	Note 7
t_{IS}	Internal SYNDET Delaya Time from Rising Edge of $\overline{RxC}$		–	–	24	t_{cyc}	Note 7
t_{ES}	External SYNDET Set-Up Time fore Falling Edge of $\overline{RxC}$		16	–	–	t_{cyc}	Note 7
$t_{TxEMPTY}$	TxEMPTY Delay Time from Center of Last Bit		20	–	–	t_{cyc}	Note 7
t_{WC}	Control Delay Time from Rising Edge of $\overline{WR}$ (TxEN, $\overline{DTR}$, $\overline{RTS}$)		8	–	–	t_{cyc}	Note 7
t_{CR}	$\overline{DSR}$, $\overline{CTS}$ Set-Up Time for $\overline{RD}$		20	–	–	t_{cyc}	Note 7

050489

Note :

1) AC Test Conditions : Output measuring Point $V_{OH}=2.0V$, $V_{OL}=0.8V$
   Input supply level $V_{IH}=2.4V$, $V_{IL}=0.45V$
2) Assumes that Address is valid before the falling edge of $\overline{RD}$.
3) C_L means load capacitance.
4) This recovery time is defined only for Mode Intialization.
   Write Data is allowed only when TxRDY = 1. Recovery Time between Writes for Asynchronous Mode is 8 tcy and for Synchronous Mode is 16 tcy.
5) The TxC and RxC frequencies have the following limitations with respect to CLK:
   For 1x Transfer Rate, f_{Tx} or $f_{Rx} \leqq 1/(30tcy)$
   For 16x and 64x Transfer Rate, f_{Tx} or $f_{Rx} \leqq 1/(4.5tcy)$
6) Minimum Reset Pulse Width is 6 tcy. System Clock must be running during Reset.
7) Status up data can have a maximum delay of 28 clock periods from the event affecting the status.

6. TIMING WAVEFORMS

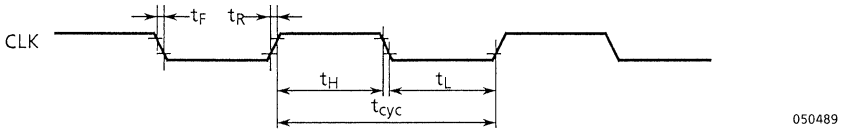


Figure 6.1 System Clock

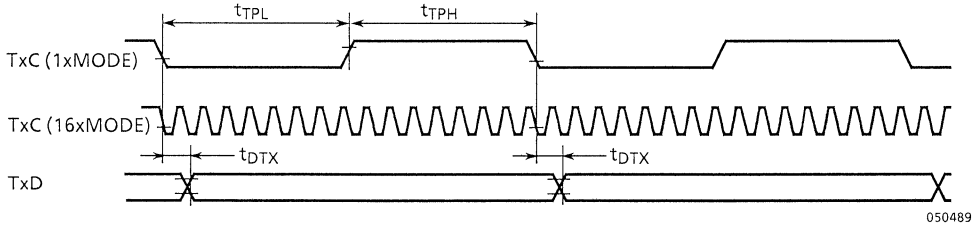


Figure 6.2 Transmitter Clock and Data

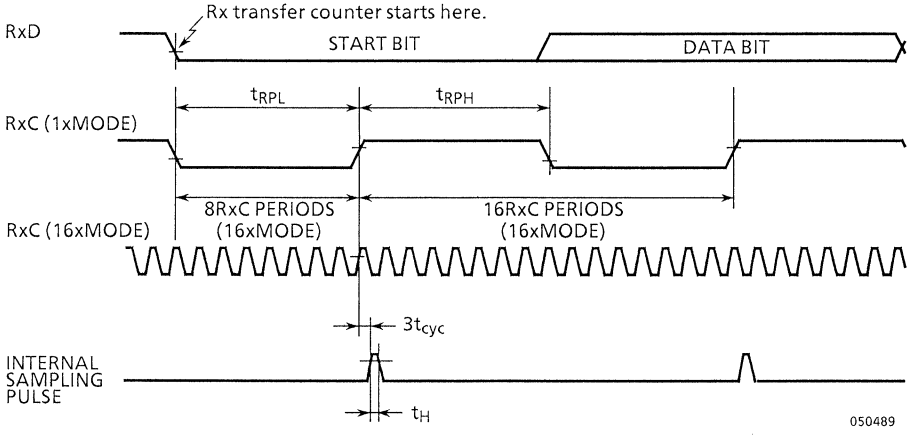


Figure 6.3 Receiver Clock and Data

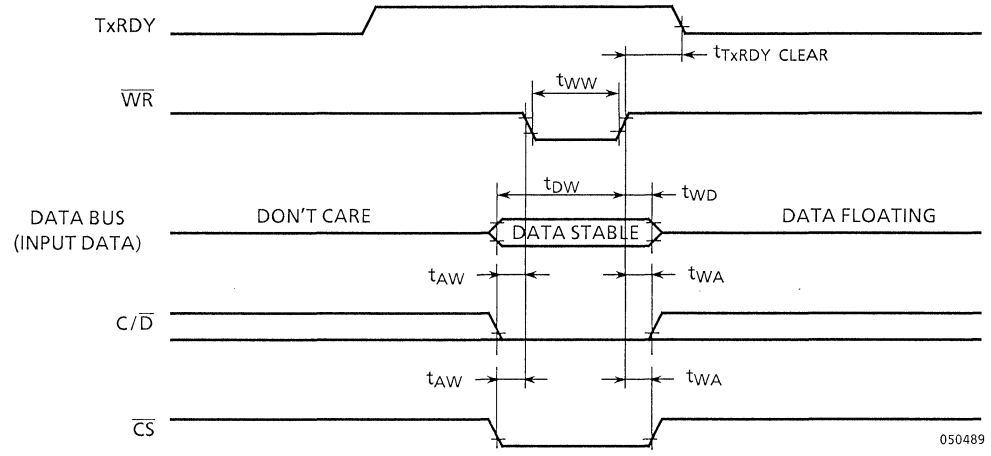


Figure 6.4 Write Data Cycle (MPU→8251A)

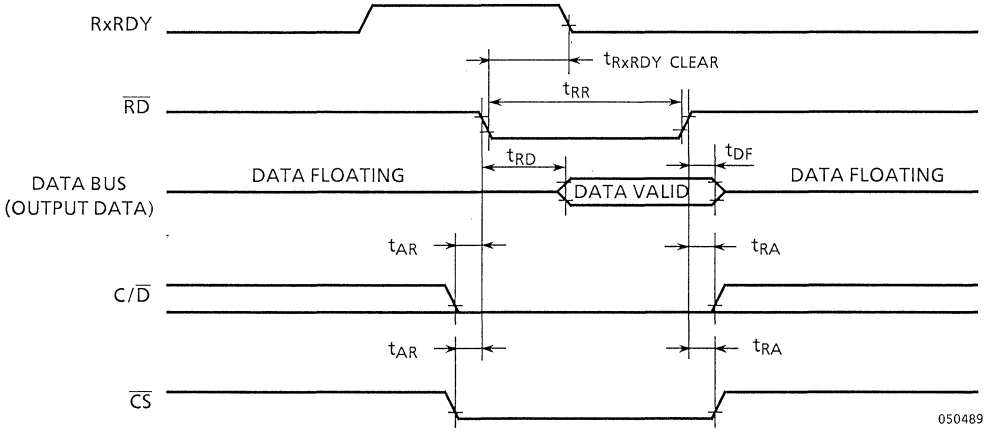


Figure 6.5 Read Data Cycle (8251A→MPU)

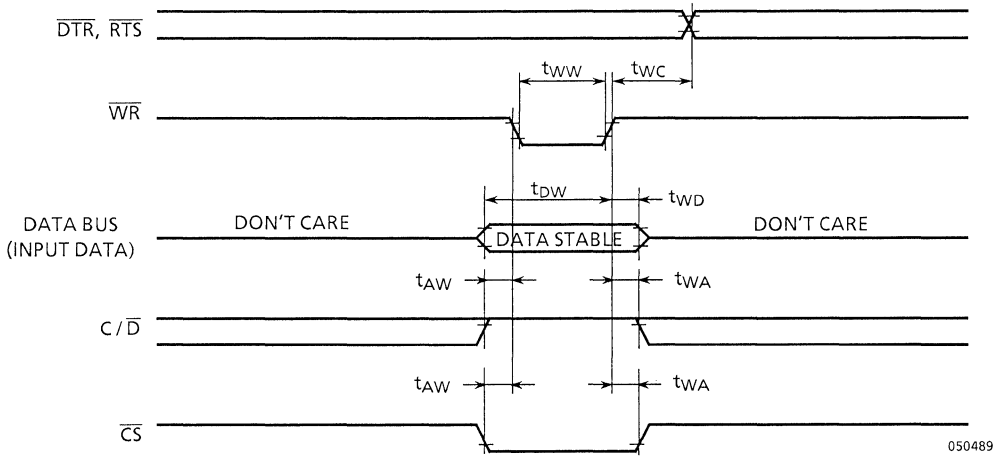


Figure 6.6 Write Contorl or Output Port Cycle (MPU→82251A)

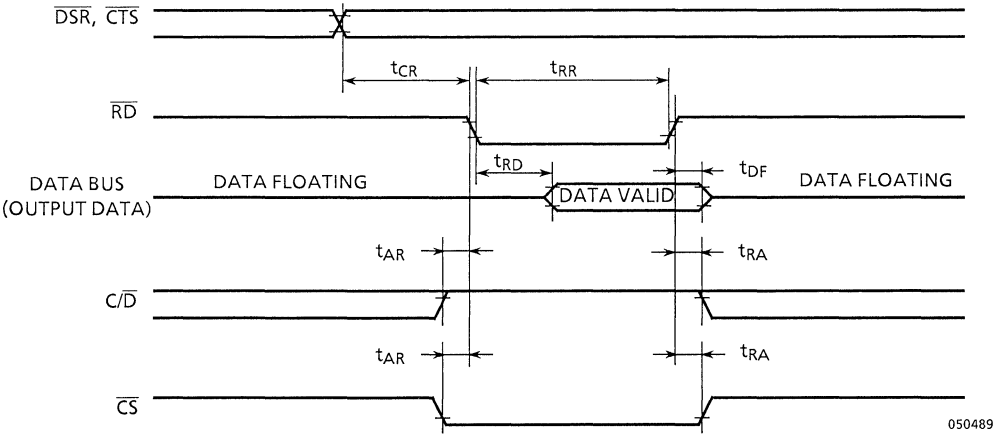
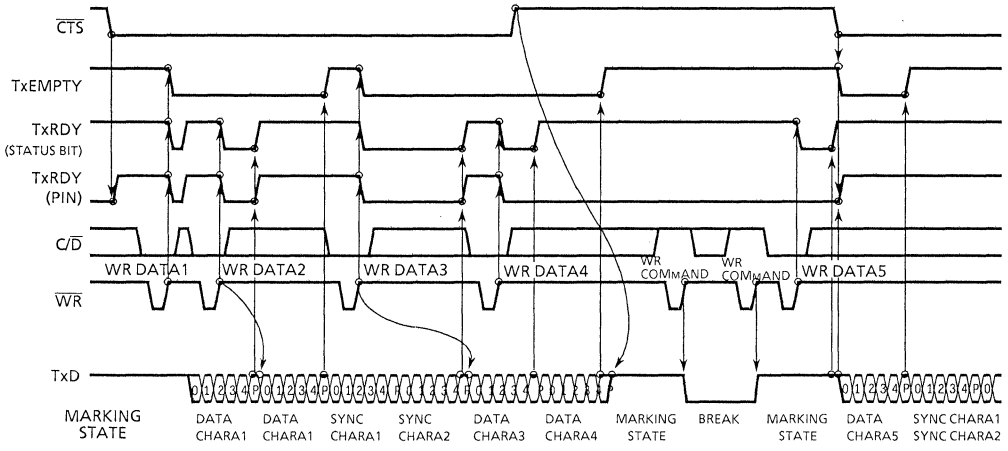


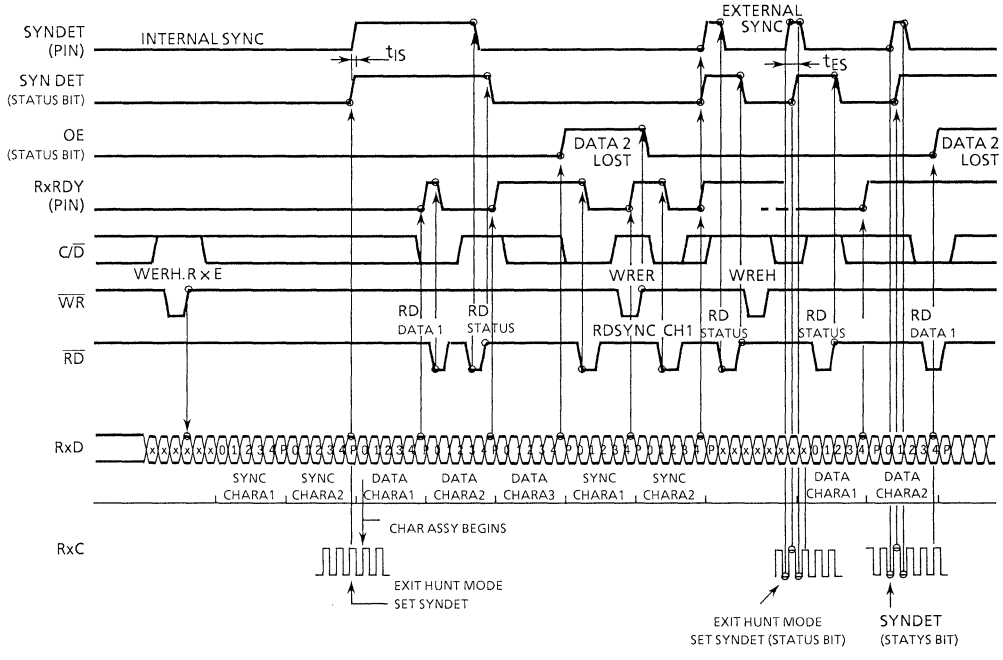
Figure 6.7 Read Control or Input Port Cycle (8251A→MPU)



EXAMPLE FORMAT : 5 BIT CHARACTER WITH PARITY 2 SYNC CHARACTERS.

050489

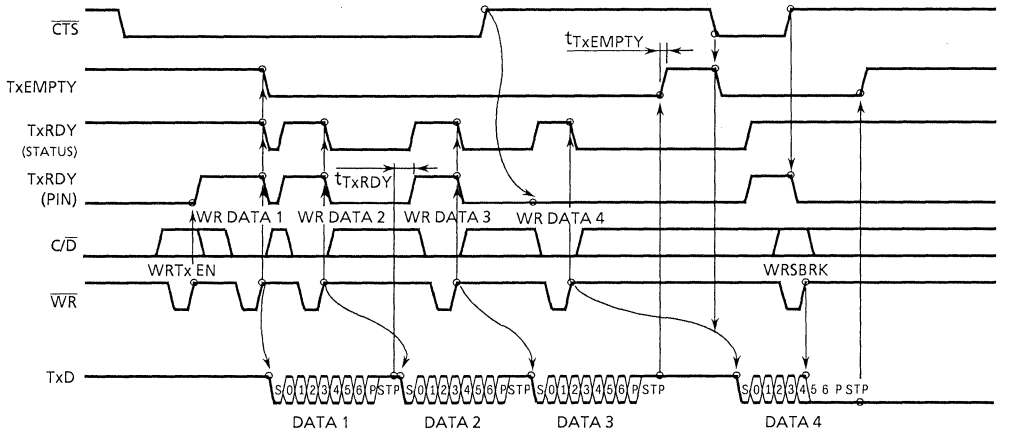
Figure 6.8 Transmitter Control and Flag Timing (SYNC Mode)



EXAMPLE FORMAT : 5 BIT CHARACTER WITH PARITY 2 SYNC CHARACTERS.

050489

Figure 6.9 Receiver Controla nd Flag Timing (SYNC Mode)

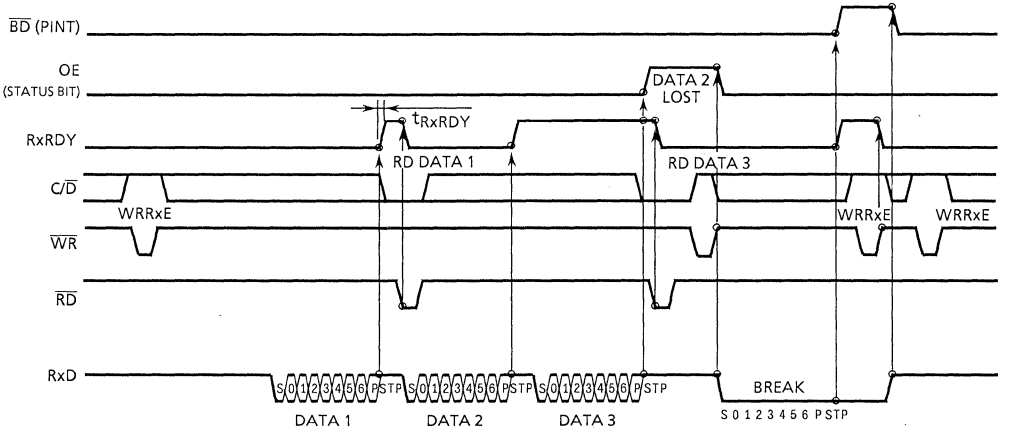


EXAMPLE FORMAT : 7 BIT CHARACTER & 2 STOP BITS.

050489

Note : TxRDY (PIN) = (Transmit Data Buffer is empty) · (TxEN = 1) · (CTS = 0)
TxRDY (STATUS BIT) = (Transmit Data Buffer is empty)

Figure 6.10 Transmitter Control and Flag Timing (SYNC Mode)



EXAMPLE FORMAT : 7 BIT CHARACTER & 2 STOP BITS.

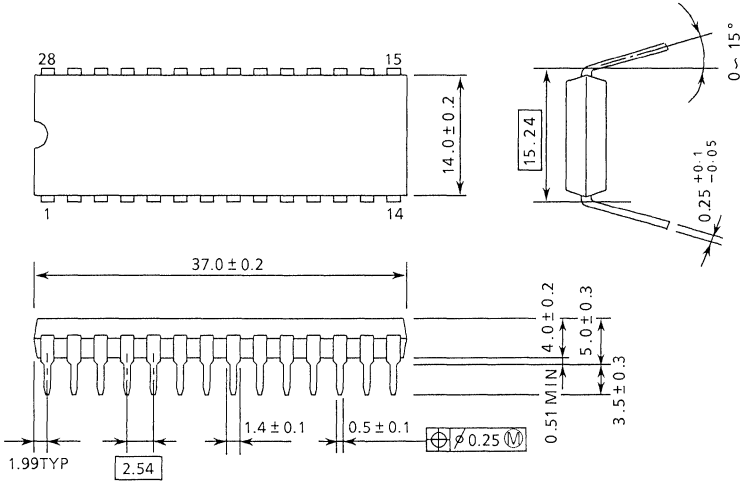
050489

Figure 6.11 Receiver Control and Flag Timing (ASYNC Mode)

7. OUTLINE DRAWING (Dual Inline Package)

DIP28-P-600

Unit : mm



270289

Note : Lead pitch is 2.54m and to lerance is ±25mm against theoretical center of each lead that is obtained on the basis of No.1 and No.28 leads.

PROGRAMMABLE INTERVAL TIMER

TMP82C53P-2

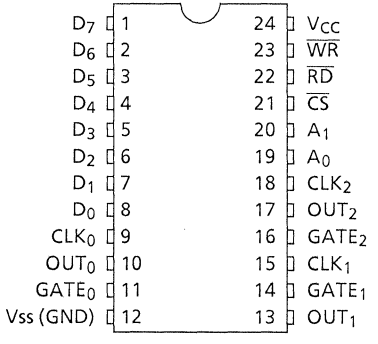
1. GENERAL DESCRIPTION

The TMP82C53P-2 (hereinafter referred to as TMP82C53) is a programmable counter/timer. It is organized as 3 independent 16 bit counters, each operates with a count rate of up to 5MHz. All modes of operation are software programmable.

2. FEATURES

- Count Binary or BCD
- 3 Independent 16 Bit Counters
- Single +5V Supply
- Count rate DC to 5MHz
- 6 programmable Counter Mode
- Low Power Consumption 5mA MAX. @5MHz
- Extended Operating Temperature −40°C to 85°C

3. PIN CONNECTIONS



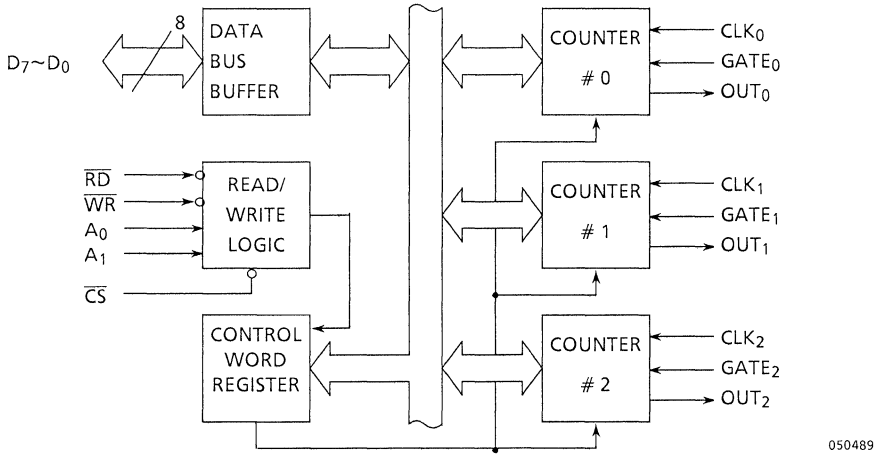
TMP82C53P-2

Table 3.1 PIN NAMES

D_7~D_0	Data Bus (8 bit)
CLK n	Counter Clock Input
GATE n	Counter Gate Input
OUT n	Counter Output
$\overline{RD}$	Read Counter
$\overline{WR}$	Write Counter
$\overline{CS}$	Chip Select
A_0~A_1	Counter Select
V_{CC}	+5V
V_{SS}	Ground(0v)

050489

4. BLOCK DIAGRAM



5. PIN NAMES AND PIN DESCRIPTION

- Vss (GND) (Power Supply)

  Ground.

- VCC (Power supply)

  +5V during operation.

- $\overline{CS}$ (Input)

  A low level input on this pin enables read and write communication between the MPU and the TMP82C53. The $\overline{CS}$ input has no effect upon the actual operation of the counters.

- A_0, A_1 (Input)

  These inputs acts in conjunction with the $\overline{CS}$, $\overline{WR}$, and $\overline{RD}$ pins. This pin is used to select one of the three counters to be operated on and to address the control word register for mode selection.

- $\overline{WR}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP82C53 to accept mode information or loading counters from the MPU.

- $\overline{RD}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP82C53 to output a counter value onto the data bus.

- D_0-D_7 (Input/Output)

  Bidirectional 8bit data bus. Mode information and the count values are transferred via this data bus.

- CLK_0-CLK_2 (Input)

  Clock inputs to counters. Falling edge on this pin enable the counter to count down.

- $GATE_0$-$GATE_2$ (Input)

  Gate inputs to counters. The function of this pin differs by the mode selection of counter operation.

- OUT_0-OUT_2 (Output)

  Outputs from the counters. The output signal from this pin differs by the mode selection of counter operation.

6. FUNCTIONAL DESCRIPTION

[Block Description]

Data Bus Buffer

This is 3-state, bidirectional, 8 bit buffer used for interfacing the TMP82C53 to the system data bus. The Data Bus Buffer has three functions as follows. Programming the MODEs of the TMP82C53, Loading the count registers, and Reading the count values.

Read/Write Logic

The Read/Write Logic accepts inputs from the system bus and in turn generates control signals for overall device operation.

Table6.1 Counter Addressing

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	
0	1	0	0	0	Load Counter #0
0	1	0	0	1	Load Counter #1
0	1	0	1	0	Load Counter #2
0	1	0	1	1	Write Mode Word
0	0	1	0	0	Read Counter #0
0	0	1	0	1	Read Counter #1
0	0	1	1	0	Read Counter #2
0	0	1	1	1	Data Bus is in High-impedance state
1	×	×	×	×	
0	1	1	×	×	

050489

Control Word Register

The Control Word Register is selected when A_0, A_1 are 11. It then accepts information from the data bus buffer and stores it in a register. The information stored in this register controls the operation mode of each counter, selection of binary or BCD counting and the loading of each count register. No reading of the contents of the control Word Register is available.

Counter #0, Counter #1, Counter #2

Since these three counters are identical, only a single counter will be described herafter. Each counter consists of a single, 16bit, presettable, down counter. The counter can operate in either binary or BCD and its input, gate and output are configured by the selection of modes (6 modes: MODE 0 to MODE 5) stored in the Control Word Register. Also the control word handles the loading of the count value so that software overhead can be minimized for these functions.

The reading of the contents of each counter is available to the programmer with simple read operations for event counting applications. Special commands and logic are built in the TMP82C53 so that the contents of each counter can be read "on-the-fly" without having to inhibit the clock input.

[MODE Definition]

Mode 0: Interrupt on Terminal Count.

The output will be initially low after the mode set operation. After the count is loaded into the selected count register, the output will remain low and the counter will count. When terminal count is reached, the output will go high and remain high until the selected count register is reloaded with the mode or a new count value is loaded. The counter continues to decrement after terminal count has been reached.

Rewriting a counter register during counting results in the following:

(1) Write 1st byte stops the current counting.
(2) Write 2nd byte starts the new count.

MODE1: Programmable One Shot.

The output will go low on the count following the rising edge of the GATE input. The output will go high on the terminal count. If a new count value is loaded while the output is low it will not affect the duration of the one-shot pulse until the succeeding trigger. The current count can be read at any time without affecting the one-shot pulse.

The one-shot is retriggerable, hence the output will remain low for the full count after any rising edge of the GATE input.

MODE2: Rate Generator

Divide by N counter. The output will be low for one period of the input clock. The period from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses the present period will not be affected, but the subsequent period will reflect the new value.
The GATE input, when low, will force the output high. When the GATE input goes high, the count will start from the initial count. Thus, the GATE input can be used to synchronize the counter.

When this mode is set, the output will remain high until after the count register is loaded. The output then can also synchronized by software.

MODE 3: Square Wave Rate Generator

Similar to MODE 2 except that the output will remain high until one half the count has been completed (for even numbers) and go low for the other half of the count.
This is accomplished by decrementing the counter by two on the falling edge of each clock pulse. When the counter reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated.

If the count is odd and the output is high, the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the clock by 2. After timeout, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decrements the counter by 3. Subsequent clock pulses decrement the count by 2 until timeout. Then the whole process is repeated. In this way, if the count is odd, the output will be high for $(N+1)/2$ counts and low for $(N-1)/2$ counts.

MODE 4: Software Triggered Strobe

After the mode is set, the output will be high. When the count is loaded, the counter will begin counting. On terminal count, the output will go low for one input clock period, then will go high again.
If the count register is reloaded between output pulses, counting will continue from the new value. The count will be inhibited while the GATE input is low. Reloading the counter register will restart counting beginning with the new number.

MODE 5: Hardware Triggered Strobe

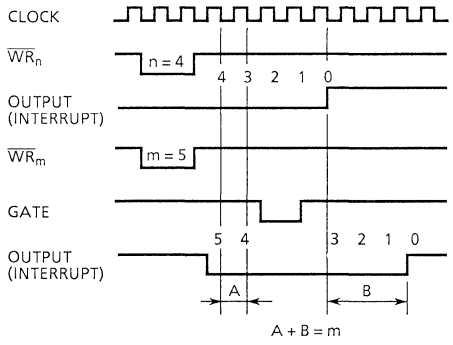
The counter will start counting after the rising edge of the trigger input and will go low for one clock period when the terminal count is reached. The counter is retriggerable. The output will not go low until the full count after the rising edge of any trigger.

Table 6.2 Gate Pin Operations

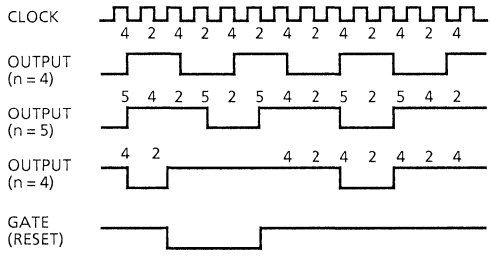
Status Modes	Low or Going Low	Rising	High
0	Disables counting	—	Enable counting
1	—	(1) Initiates counting (2) Resets output after next clock	
2	(1) Disables counting (2) Sets output immediately High	Initiates counting	Enable counting
3	(1) Disables counting (2) Sets output immediately High	Initiates counting	Enable counting
4	Disables counting	—	Enable counting
5	—	Initiates counting	—

050489

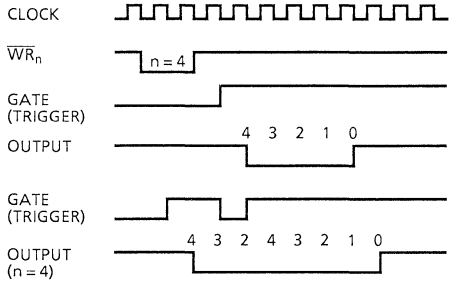
MODE0 : Interrupt on Terminal Count



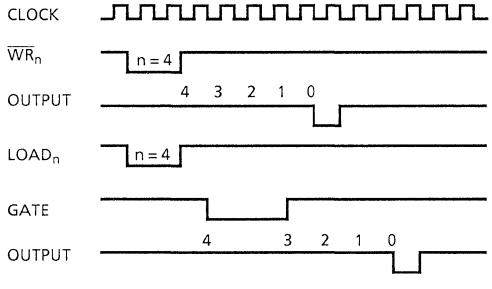
MODE3 : Square Wave Generator



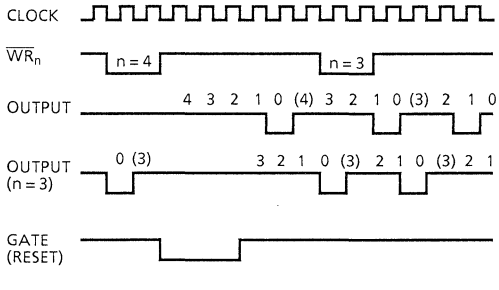
MODE1: Programmable One-Shot



MODE4 : Software-Triggered Strobe



MODE2 : Rate Generator



MODE5 : Hardware-Triggered Strobe

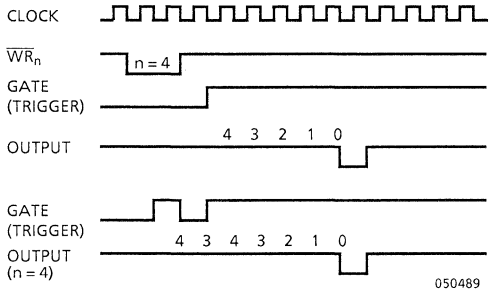
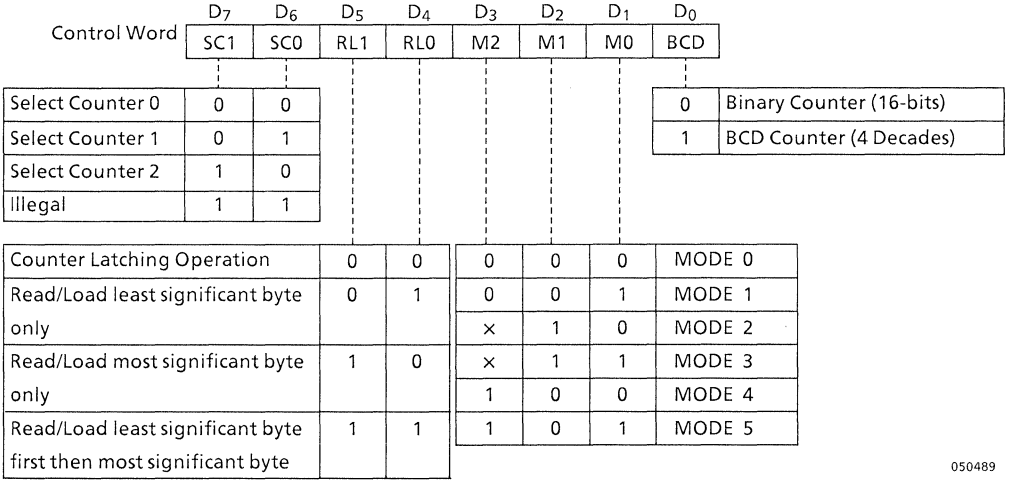


Figure 6.1 TMP82C53 Timing Diagrams

7. PROGRAMMING THE TMP82C53

All of the MODEs for each counter are programmed by the systems software by simple I/O operations.

Each counter of the TMP82C53 is individually programmed by writing a control word into the Control Word Register. ($\overline{CS}=0, A_0=A_1=1, \overline{WR}=0$)



Note. SC: Select Counter, RL: Read/Load, M: Mode, BCD: Binary Coded Decimal.

The programmer must write out to the TMP82C53 a Mode Control Word and the programmed number of count register bytes (1 or 2) prior to actually using the selected counter.

The actual order of the programming is quite flexible. Writing out of the Mode Control Word can be in any sequence of counter selection.

The loading of the Count Register with actual count value, however, must be done in exactly the sequence programmed in the Mode Control Word (RL0, RL1).

8. Counter Loading

The count register is not loaded until the count value is written (one or two bytes, depending on the mode selected by the RL bits), followed by a rising edge and a falling edge of the clock.

The count register must be loaded with the number of bytes programmed in the Mode Control Word. The one or two bytes to be loaded in the count register do not have to follow the associated Mode Control Word. They can be programmed at any time following the Mode Control Word loading as long as the correct number of bytes is loaded in order.

All counters are down counters. Loading all zeros will result in the maximum count (2^{16} for Binary or 10^4 for BCD). In MODE 0 and MODE 4, the new count will not restart until the load has been completed.

9. Read Operations

The TMP82C53 contains the circuit that will allow the programmer to easily read the contents of any of the three counters without distrubing the actual count in progress. There are two methods that the programmer can use to read the value of the counters. The first method involves the use of simple I/O read operations. By controlling the A_0, A_1 inputs to the TMP82C53, the programmer can select the counter to be read. The only requirement with this method is that in order to assure a stable count reading the actual this method is that in order to assure a stable count reading the actual operation of the selected counter must be inhibited either by controlling the GATE input or by external logic that inhibits the clock input.

The contents of the counter selected must be read in the sequence programmed in the Mode Control Word (RL0, RL1). When RL0, RL1 is 11. First I/O Read contains the least significant byte (LSB), second I/O Read contains the most significant byte (MSB), and the two bytes must be read before any loading WR command can be sent to the same counter.

The second method allows the programmer to read the contents of any counter without effecting or distrubing the count operation. When the programmer wishes to read the contents of a selected counter "On-the-fly", a special code should be written to the Mode register so that the counter holds an accurate, stable count value. The programmer then issues a normal read command to the selected counter. The contents of the latched register must be read in the sequence programmed in the Mode Control Word (RL0, RL1). This commands has no effect on the counters mode.

10. Program Example

```
Set up sequence      MVI   A, 00110000B    ... #0, LSB-MSB, MODE 0, Binary
for counter #0       OUT   CWAD            ... The address of Control Word Register
                     MVI   A, 53H          ... LSB for counter #0
                     OUT   CNT0            ... The address of counter #0
                     MVI   A, 82H          ... MSB for counter #0
                     OUT   CNT0            ... The address of counter #0
                           :
                           :
READ the contents    MVI   A, 0000XXXXB
of counter #0        OUT   CWAD            ... Latching count
                     IN    CNT0            ... Read LSB of counter #0
                     MOV   L, A
                     IN    CNT0            ... Read MSB of counter #0
                     MOV   H, A
                     :
                     :
RELOAD to            MVI   A, 28H
counter #0           OUT   CNT0            ... Load LSB for counter #0
                     MVI   A, 53H
                     OUT   CNT0            ... Load MSB for counter #0
```

11. ELECTRIC CHARACTERISTICS

11.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING	UNIT
VCC	Vcc Supply Voltage (with respect to Vss [GND])	− 0.5 to + 7.0	V
VIN	Input Voltage (with respect to Vss [GND])	− 0.5 to Vcc + 0.5	V
VOUT	Output Voltage (with respect to Vss [GND])	− 0.5 to Vcc + 0.5	V
PD	Power Dissipation	250	mW
Tsol	Soldering Temperature (Sodlering Time 10 sec)	260	°C
Tstg	Storage Temperature	− 65 to + 150	°C
Topr	Operating Temperature	− 40 to + 85	°C

050489

11.2 DC CHARACTERISTICS

Ta = − 40°C to + 85°C, Vcc = 5V ± 10%, Vss(GND) = 0V

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
VIL	Input Low Voltage		− 0.5		0.8	V
VIH	Input High Voltage		2.2		Vcc + 0.5	V
VOL	Output Low Voltage	IOL = 2.2mA			0.45	V
VOH1	Output High Voltage	IOH = − 400μA	2.4			V
VOH2	Output High Voltage	IOH = − 100μA	Vcc − 0.8			V
IIL	Inupt Leak Current	0V ≦ VIN ≦ Vcc			± 10	μA
IOFL	Output Leak Current	0.45V ≦ VOUT ≦ Vcc			± 10	μA
ICC1	Operating Supply Current	CLK = 5MHz VIH = Vcc − 0.2V VIL = 0.2V			5	mA
ICC2	Stand-by Supply Current	CLK = DC VIH = Vcc − 0.2V VIL = 0.2V			10	μA

050489

11.3 AC CHARACTERISTICS

Ta = − 40°C TO + 85°C, Vcc = 5V ± 10%, Vss(GND) = 0V

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
tAR	Address Set-up Time ($\overline{RD}$)		30			ns
tRA	Address Hold Time ($\overline{RD}$)		0			ns
tRR	$\overline{RD}$ Pulse Width		180			ns
tRD	Valid Data ($\overline{RD}$)	CL = 150pF			140	ns
tDF	Data Floating ($\overline{RD}$)		10		85	ns
tRV	Recovery Time		250			ns
tAW	Address Set-up Time ($\overline{WR}$)		0			ns
tWA	Address Hold Time ($\overline{WR}$)		30			ns
tWW	$\overline{WR}$ Pulse Width		150			ns
tDW	Data Set-up time ($\overline{WR}$)		100			ns
tWD	Data Hold Time ($\overline{WR}$)		30			ns
tCLK	Clock Period		200		DC	ns
tPWH	CLK High Pulse Width		80			ns
tPWL	CLK Low Pulse Width		65			ns
tGW	GATE Width High		50			ns
tGL	GATE Width Low		50			ns
tGS	GATE Set-up Time (CLK)		70			ns
tGH	GATE Hold Time (CLK)		50			ns
tOD	Output Delay From (CLK)	CL = 150pF			200	ns
tODG	Output Delay From (GATE)	CL = 150pF			200	ns

050489

Note : AC timings measurements are referenced to VIL = 0.45V, VIH = 2.4V,
VOL = 0.8V, VOH = 2.2V.

11.4 INPUT CAPACITANCE

Ta = 25°C, Vcc = Vss (GND) = 0V

SYMBOL	PPRAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
CI	Input Capacitance	fC = 1MHz Unmeasured pins, 0V			10	pF
CI/O	Input/Output Capacitance				20	

050489

12. TIMING DIAGRAM

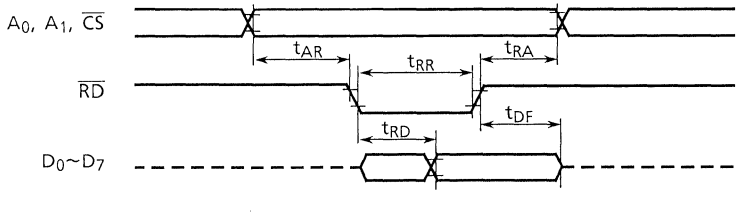


Figure 12.1 Read Timing

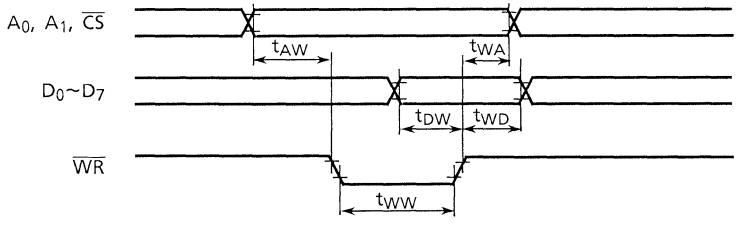


Figure 12.2 Write Timing

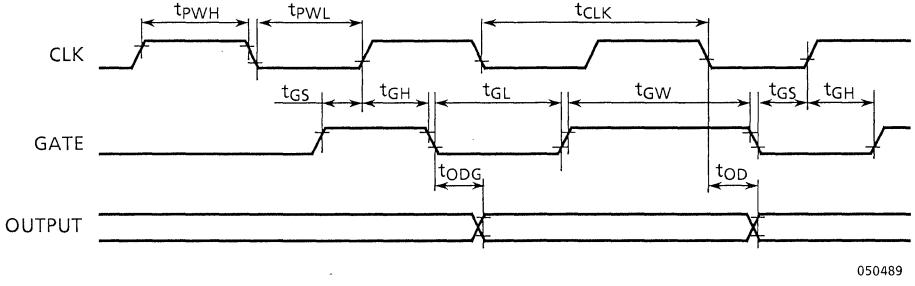


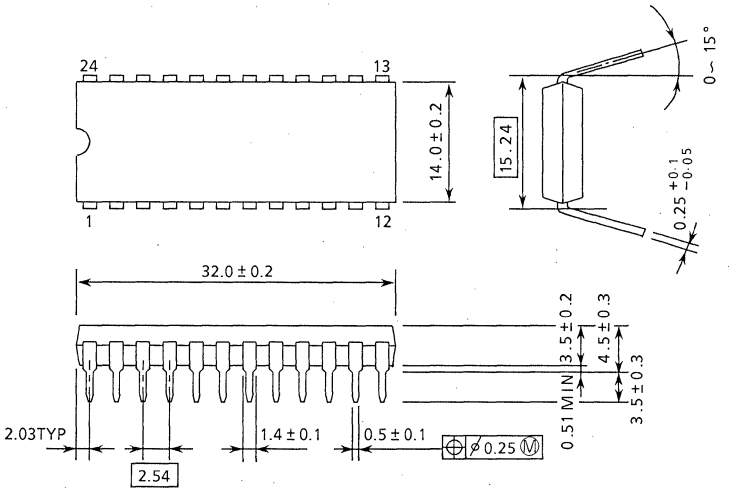
Figure 12.3 Clock & Gate Timing

13. EXTERNAL DIMENSION VIEW

13.1 24 Pin PLASTIC DIP

DIP24-P-600

Unit : mm



270289

Note : Lead pitch is 2.54mm and tolerance is ±0.25mm against theoretical center of each lead that is obtained on the basis of No.1 and No.24 leads.

PROGRAMMABLE INTERVAL TIMER

TMP82C54P-2 / TMP82C54M-2

The TMP82C54P-2 TMP82C54M-2 (hereafter called the TMP82C54) is a low power, CMOS general-purpose programmable timer/counter.

The TMP82C54 consists of 3 independent 16-bit counters to implement high-speed counting. It contributes to a greatly enhanced system throughput.

1. FEATURES

(1) Pin compatible with the TMP82C53P-2 (TMP82C54P-2)

(2) Three independent, 16-bit counters

(3) Count data read is available.

(4) Status read is available.

(5) Six programmable Counter modes

(6) Binary/decimal count selection

(7) High-speed operation
TMP82C54P-2/M-2 Clock input = 10MHz Max.

(8) Low power dissipation
In operation 30mA max.
In stand-by 10μA max.

(9) Extended operating temperature
−40°C to +85°C

(10) Power supply voltage
5V ± 10%

2. PIN CONNECTIONS (TOP VIEW)

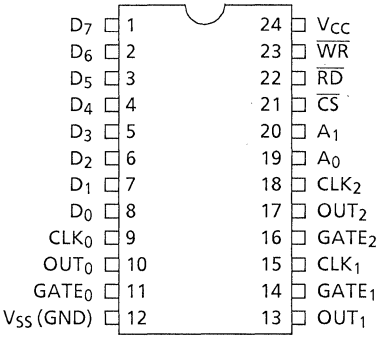


Figure 2.1 Pin Connections

3. BLOCK DIAGRAM

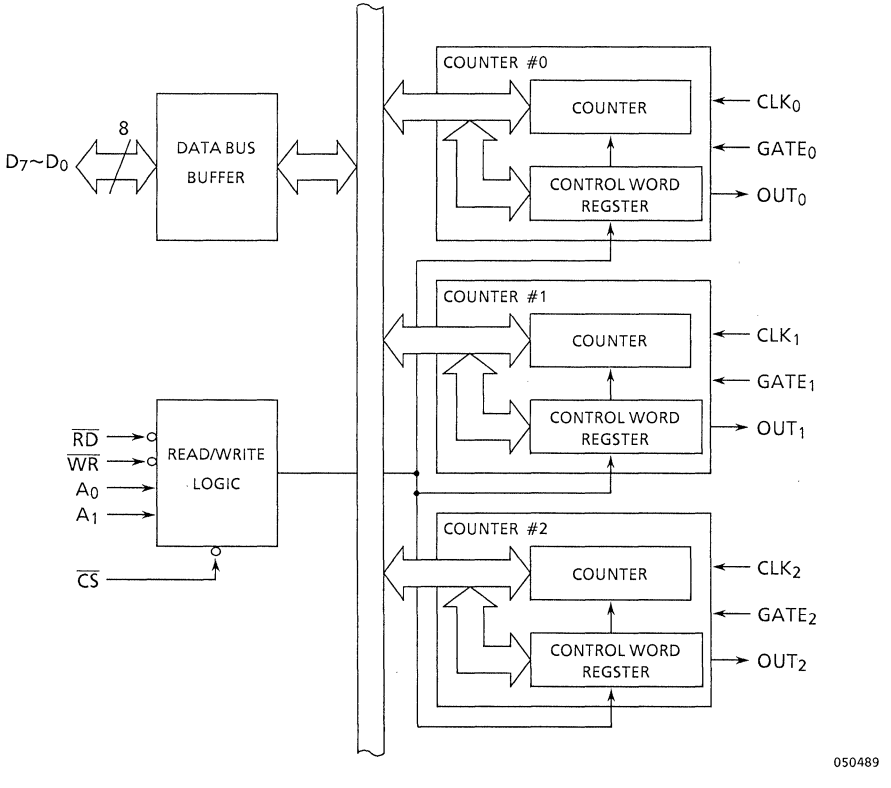


Figure 3.1 Block Diagram

4. I/O SIGNAL DESCRIPTION

- $\overline{CS}$ (Chip Select input)

  A low this input enables the TMP82C54. Unless $\overline{CS}$ is in the enabled state, the read/write operation is ignored. The $\overline{CS}$ input does not affect the actual counter operation.

- A_0, A_1 (Address inputs)

  These inputs are used to select one of the 3 internal counters or the control word register.

- $\overline{WR}$ (Write input)

  The low active write strobe signal for the TMP82C54 used, to set the mode or load data into the counters.

- $\overline{RD}$ (Read input)

  The low active read strobe signal for the TMP82C54 used, to read the value of the internal counter.

- $D_0 - D_7$ (Data bus I/O)

  The 8-bit bidirectional 3-state data bus pins connected to the microcomputer data bus.

- CLK $_{0\sim3}$ (Clock input)

  The clock input pin for the counter. The counter starts decrementing at the falling edge of this signal.

- GATE $_{0\sim3}$ (Gate input)

  The gate input pin for the counter. The function depends on the mode stored in the control word register.

- OUT $_{0\sim3}$ (Out output)

  The output pin form the counter. One of 6 different waveforms selected by the mode stored in the control word register is output.

5. OPERATIONAL DESCRIPTION

5.1 DATA BUS BUFFER

This is the bidirectional 3-state 8-bit data buffer to interface with the microcomputer system data bus. The transfer of the control word for mode set, data write to the counter, and data read from the counter are all performed through this buffer.

5.2 READ/WRITE LOGIC

This circuit receives the control signals ($\overline{RD}$, $\overline{WR}$) form the system bus to generate the control signals for the entire TMP82C54 system operations.

The chip select input ($\overline{CS}$) enables/desables these control signals. When the $\overline{CS}$ is in the disabled state, the control signal states do not affect the internal operations.

5.3 CONTROL WORD REGISTER

The control word register is selected when $A_0 = 1$ and $A_1 = 1$.

The information from the data bus buffer is stored in this register to select the counter operation mode, binary or decimal counting, and control the specification of the counter loading method. The control word can be read by using the read-back command.

5.4 COUNTERS #0-#2

These counters perform exactly the same operation independently.

The following description is made referring to only one counter for convenience. Figure 5.1 shows the block diagram of the counter.

The counter is the 16-bit presettable synchronous down-counter.

The status register, consisting of 8 bits, holds the current states of the control word register, output pin, and the null count flag information. (See "Read-back Command.")

The count latch, consisting of two 8 bits, normally follows the current value of the down-counter (non latch state). When the TMP82C54 receives an proper count latch command, the count latch holds the value of the down-counter at this time. When the latched data is read by the MPU, the latch is cleared and put in the non latch state again. After the count latch command, count value to be read is not the current value of the down-counter but the value of the count latch.

The count register consists of two 8 bits. When the count value is written to the counter, at first, the value is loaded into the counter register. Then the count value is loaded into the down-counter. Writing the count value to the count register is performed in units of 8 bits.

However, the count value is loaded into the down-counter in units of 16 bits.

When the counter value is written to the down-counter programmed in the read/load 1-byte mode, the other byte is automatically cleared to 00H.

The control logic is connected to the CLK input, GATE input, and OUT output signals. The operations of these signals specified by selecting one of modes 0-5 programmed in the control word register.

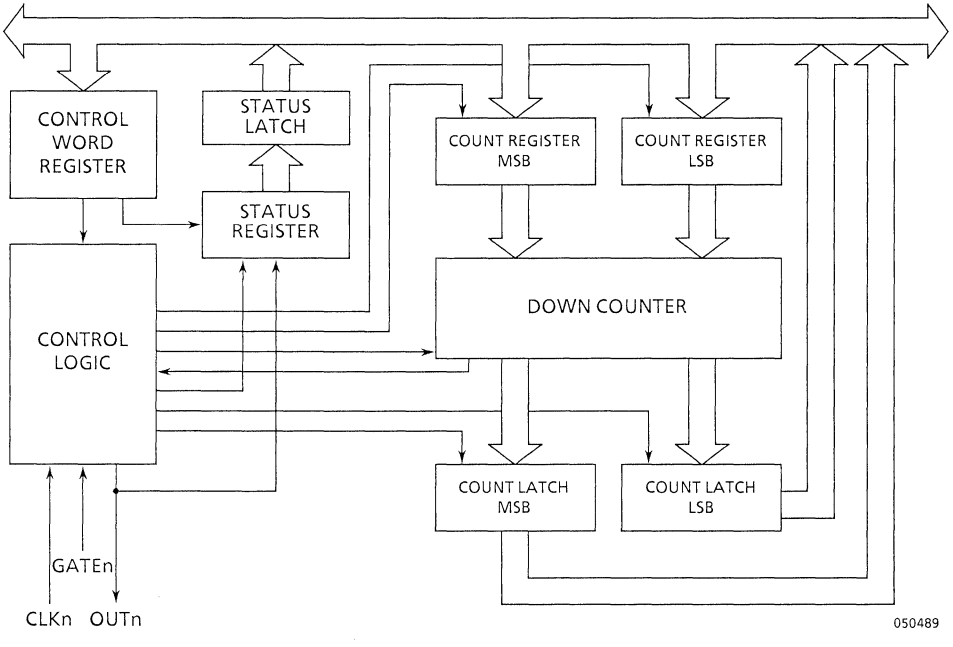


Figure 5.1 Block Diagram of Internal Counter

6. PROGRAMMING

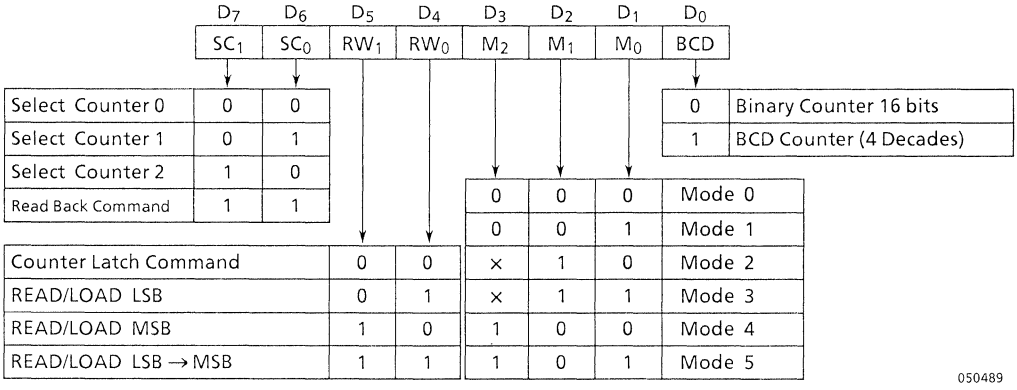
The OUT output pin state, the counter mode, and the count register values of the TMP82C54 are still undefined immediately after the power is turned on.

This requires to program the counters to be used. However, the counters not to be used need not be programmed.

6.1 MODE SETTING

The count modes of the TMP82C54 can be set with a simple I/O instruction. Each counter of TMP82C54 can be programmed separately by writing the control word to the control word register ($\overline{CS}=0$, $A_0=1$, $A_1=1$, $\overline{WR}=0$).

6.1.1 Control Word



Note : 1. Read/Load least significant byte first, then most significant byte
2. Read/Load is $\overline{RD}/\overline{WR}$
3. × : Don't care

Figure 6.1 Control Word

There is no specified order in which the counters are selected for mode setting. Mode setting can start from any counter.

6.2 COUNTER WRITING

Paying attention to the following two points makes it possible to program the TMP82C54 in any sequence.

(1) Writing of the control word to each counter must be performed before the count value is written.

(2) The count value must be written in the format specified in the control word (bits RW_0 and RW_1).

6.3 PROGRAM SEQUENCE EXAMPLES (IN THE LSB-MSB 2-BYTE MODE)

(1)
Counter 0: Control word
Counter 0: Lower 8 bits
Counter 0: Upper 8 bits
Counter 1: Control word
Counter 1: Lower 8 bits
Counter 1: Upper 8 bits
Counter 2: Control word
Counter 2: Lower 8 bits
Counter 2: Upper 8 bits

(2)
Counter 0: Control word
Counter 1: Control word
Counter 2: Control word
Counter 0: Lower 8 bits
Counter 1: Lower 8 bits
Counter 2: Lower 8 bits
Counter 0: Upper 8 bits
Counter 1: Upper 8 bits
Counter 2: Upper 8 bits

(3)
Counter 0: Control word
Counter 1: Control word
Counter 2: Control word
Counter 0: Lower 8 bits
Counter 0: Upper 8 bits
Counter 1: Lower 8 bits
Counter 1: Upper 8 bits
Counter 2: Lower 8 bits
Counter 2: Upper 8 bits

(4)
Counter 0: Control word
Counter 0: Lower 8 bits
Counter 1: Control word
Counter 2: Control word
Counter 1: Lower 8 bits
Counter 0: Upper 8 bits
Counter 1: Upper 8 bits
Counter 2: Lower 8 bits
Counter 2: Upper 8 bits

6.4 COUNTER READING

The counter value of the TMP82C54 can be read without disturbing the on-going count value. The counter value are read in one of the three methods as described below.

- One is simple I/O Read of the counter selected by A_0 and A_1. In this method, the GATE input must be controlled or the CLK input must be inhibited in order to read a stable count value. The procedure specified in the control word must always be followed. (That is, only the LSB 1 byte must be RW_1/RW_0 read when only the LSB 1 byte is specified; only the MSB 1 byte must be read when only the MSB 1 byte is selected; and, in the case of the continuous bytes form LSB to MSB, the LSB 1 byte and MSB 1 byte (a total of 2 bytes) must be read.)

- The on-going, momentary count value is read without stopping the counter operation. This can be achieved by specifying the counter latch operation in the control word (the counter latch command).
  When reading the value of the specified counter, a stable count value can be obtained by specifying the counter latch operation before reading the value of the specified counter. In this case, the procedure specified in the control word must be followed like the method shown above.

- The count latch is specified be the Read-Back command.

6.4.1 Counter Latch Command

$\overline{CS} = 0$, $A_0 = A_1 = 1$, $\overline{WR} = 0$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC_1	SC_0	0	0	×	×	×	×

0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Read Back Command

× : Don't Care

050489

Figure 6.2 Counter Latch Command

The selected counter holds the value of the down-counter when the counter latch command is executed. The count value remains held until the value is read or the counter is reprogrammed.

If the counter latch command is written to the same counter twice, the second one is ignored. The value to be read is the counter value counted at the time the first count latch command was written.

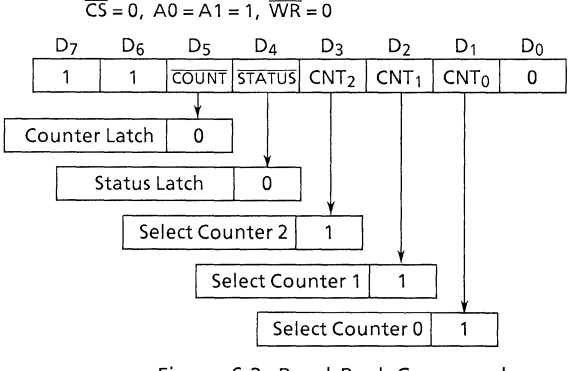
When the counter is in the 2-byte mode, the 2 bytes must be read.

The count value can be written as the following sequence :

1. Least significant byte: Read
2. Least significant byte: Write
3. Most significant byte : Read
4. Most significant byte : Write

6.4.2 Read-back Command

The read-back command is used to get the count value, the mode output pin state, and the null count flag information.



050489

Figure 6.3 Read-Back Command

The read-back command can latch multiple counters simultaneously by selecting the counter by CNT_2-CNT_0 with the $\overline{COUNT}$ bit $D_5=0$ (multiple count latch). This will provide the same operation on each command as the operation in which a count latch command has been merely written to each counter. The count value remains held in the count latch until the value is read or the counter is reprogrammed.

Reading the count value from any counter does not clear the counter latch of the other counters.

When multiple count latch is performed on a counter twice, the second one is ignored. The value to be read is the counter value counted at the time the first multiple count latch was performed.

The read-back command can latch the status information by selecting a counter by CNT_2-CNT_0 with the STATUS bit $D_4=0$. The status information are as follows.

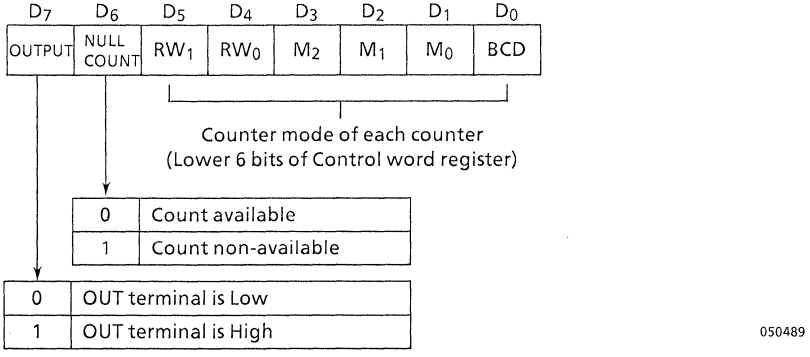


Figure 6.4 Status Data Format

Bits D_5-D_0 are the low-order 6 bits of the control word register currently programmed in that counter. These bits are held in the status register when the mode is set in the control word register and remain unchanged until the control word register is rewritten.

OUTPUT bit D_7 indicates the current OUT output pin state.

NULL COUNT bit D_6 is the flag indicating whether the count value has been loaded from the count register into the down-counter. When this flag is "0", it indicates that the count value has been loaded into the down-counter; when it is "1", it indicates that the count value has not been loaded. The following shows the NULL COUNT flag changes.

1. Write to the control word register NULL COUNT=1
2. Write a count value to the count register (Note): NULL COUNT=1
3. Load from the count register into the down-counter: NULL COUNT=0

Note : When the 2-byte mode is specified by the control word register, the NULL COUNT flag goes "1" upon writing the second byte.

The status latch is held until the status is read or the counter is reprogrammed. When the status latch is specified by the read-back command on a counter twice, the second one is ignored. The data to be read is the data present at the time the first status latch was performed.

The multiple count latch and status latch by the read-back command can be performed simultaneously by setting the $\overline{\text{COUNT}}$ and $\overline{\text{STATUS}}$ bits D_5 and D_4 to "0". This performs the same operation as the one in which multiple count latch and status latch are performed separately. Like the simultaneous latch, each latch in this case is held until it is read or the counter is reprogrammed.

If the multiple latch and status latch are performed on a counter twice, the second ones are ignored. The following shows a read-back command programming example.

Table 6.1 Read-back command programming example

Read back command	Counter 0 latch		Counter 1 latch		Counter 2 latch	
	Count	Status	Count	Status	Count	Status
Initial State	non	non	non	non	non	non
Counter 1 Status Latch	non	non	non	latch	non	non
Counter 2 Count & Status Latch	non	non	non	latch	latch	latch
Counter 0, 2 Status Latch	non	latch	non	latch	latch	ignore
Counter 0 Count Latch	latch	latch	non	latch	latch	latch
Counter 0 Status Count Read	non	non	non	latch	latch	latch
Counter 1 Count & Status Latch	non	non	latch	ignore	latch	latch
Counter 0, 2 Count & State Latch	latch	latch	latch	latch	ignore	ignore

050489

When both multiple count latch and status latch are performed by the read-back command, the status data is read in the first read of the counter regardless of the latch execution sequence. The value of the latched count register is read in the reading of the next 1 or 2 bytes (depending on the counter mode setting)

Table 6.2 Read/Write Addressing

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	Function
0	1	0	0	0	Write Counter 0
0	1	0	0	1	Write Counter 1
0	1	0	1	0	Write Counter 2
0	1	0	1	1	Write Control word
0	0	1	0	0	Read Counter 0
0	0	1	0	1	Read Counter 1
0	0	1	1	0	Read Counter 2
0	0	1	1	1	Non Operation (3 – State)
1	×	×	×	×	
0	1	1	×	×	

× : Don't Care

050489

7. MODE DESCRIPTION

The TMP82C54 has the 6 count modes.
Each count mode has the following characteristics.

- When the control word is written, the output is initialized to the state unique to each mode.
- The GATE is sampled at each rising edge of CLK. In modes 0 and 4, the GATE operates by level sense; in modes 1 and 5, it operates by rising-edge sense; and in modes 2 and 3, it operates by level sense and rising-edge sense respectively.
- The down-counter performs count loading and decrement at the falling edge of CLK.
- The maximum counter value is 0. (2^{16} in binary, 10^4 in decimal)
- The counter keeps decrementing until 0 (terminal count) is reached.
- In modes 0, 1, 4, and 5, count continues even if the count value reaches 0. In modes 2 and 3, the initial value is reloaded upon reaching the terminal count for repeated counting.

Terminology

Clock pulse	:	From the rising edge to falling edge of CLK input.
Trigger	:	The rising edge of the GATE input.
Count loading	:	Loading of count value from the count register into the down-counter.
Mode setting	:	Writing of the control word.
Terminal count	:	Reaching of the down-counter to 0.

The following sample waveform of each mode shows the binary count and read/write LSB programming case. The numbers indicate down-counter values.

7.1 MODE 0: INTERRUPT ON TERMINAL COUNT

The output is initialized to the low level after setting the mode . The output remains low after count loading. When the GATE input is high, the down-counter starts counting.

When the terminal count has been reached, the output goes high, which is held until a new count value is written or another mode is set.

The GATE input only controls the down-counter operation: it does not affect the output state.

The count value is loaded to the counter by the clock pulse following the writing of the count value to the count register.

The down-counter continues counting after the terminal count has been reached.

Rewriting to the count register during counting will cause the following actions.

1. When the first byte is written, the current count is suspended. The output is set to the low level.
2. When the second byte is written, count starts with the next clock pulse upon which the now counting starts.

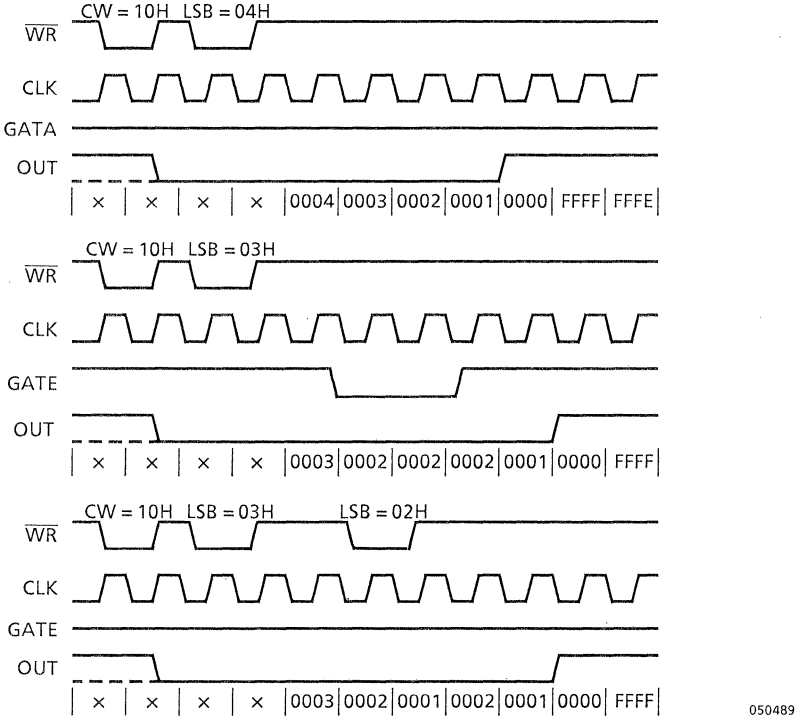


Figure 7.1 Mode 0

7.2 MODE 1: PROGRAMMABLE ONE-SHOT

The output is initialized to the high level after setting this mode and is reset to the low level by the clock pulse following the trigger. Then, the output goes high after the terminal count has been reached. The counter performs the following operations with the trigger:

1. Loading
2. Setting the output to the low level (reset)
3. Start counting

The one-shot pulse of GATE can be retriggered. Therefore, the output remians low until complete counting is made after triggering.

When a new count value is written while the output is low, the width of the one-shot pulse already output is not affected upon the next trigger.

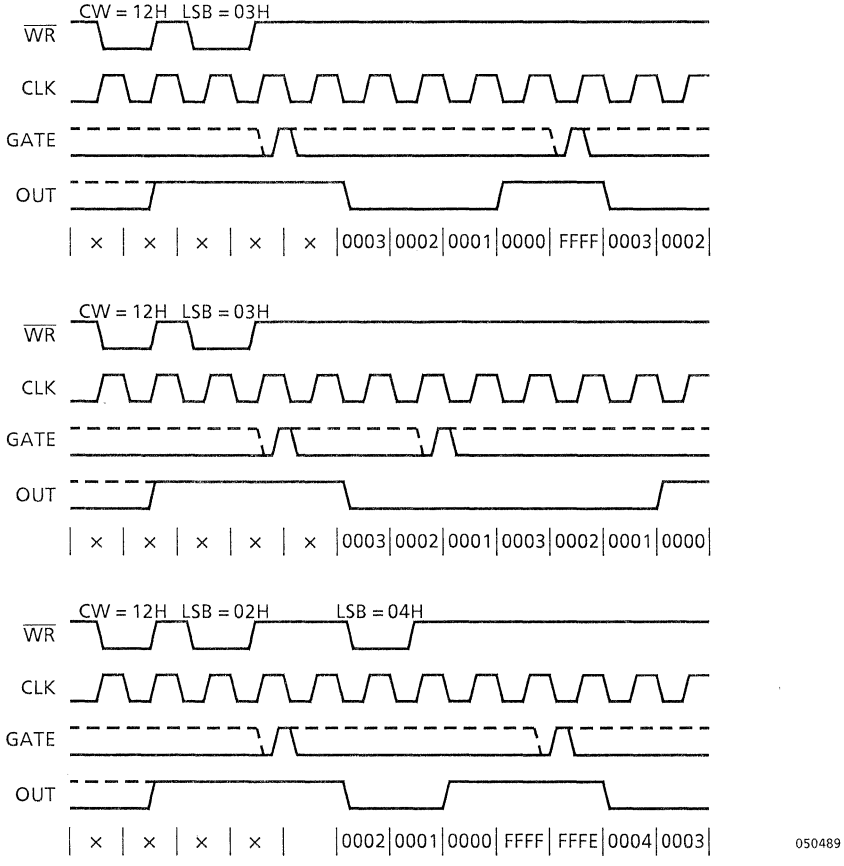


Figure 7.2 Mode1

7.3 MODE 2: RATE GENERATOR

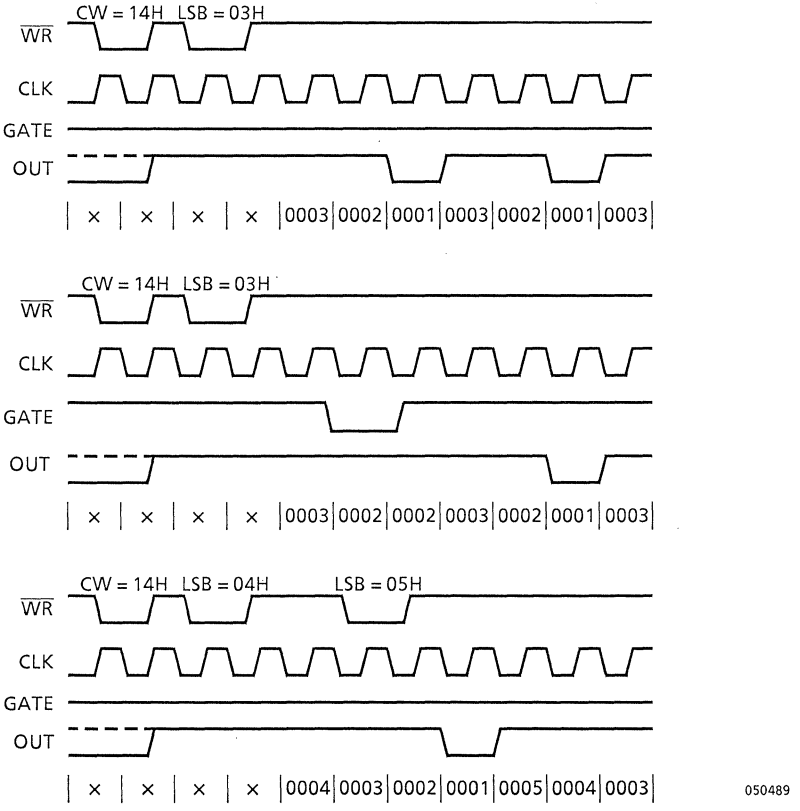
Division-by-N counter is performed.

The output is initialized to the high level after setting this mode, and goes low for 1 clock pulse when the counter has reached 1. Then, the output goes high again and the initial count is loaded for infinite counting.

When the GATE = 1, counting is enabled. When the GATE = 0, counting is disabled and the output is set to the high level. By the trigger, the initial value is loaded into the counter at the next clock pulse, starting new counting. Thus, the gate input can be used for counter synchronization. When mode 2 is set, the output goes high, which is held until count loading is performed, so that synchronization can be provided by software as well.

Rewriting to the count register being counted does not affect the current cycle. A new cycle starts after the end of the current cycle.

The minimum count of this mode is 2.



050489

Figure 7.3 Mode 2

7.4 MODE 3: SQUARE WAVE RATE GENERATOR

The output is initialized to the high level after setting this mode. The half of the count setting value (even number) becomes high and the other half becomes low. The rest is the same as mode 2.

When the GATE = 1, counting is enabled. When the GATE = 0, counting is disabled and the output is set to the high level.

When the count setting value is an even number, counting is performed by decrementing the counter by 2 at the falling edge of the clock input.

When the terminal count has been reached, the output is inverted, the count setting value is reloaded into the counter, the counter is decremented by 2, this process is repeated.

If the count value is an odd number and its output is high, the value which is the setting value - 1 is loaded into the down-counter, which is decremented by 2. The output goes low at the clock pulse following the terminal count and the setting value - 1 is loaded into the down-counter again. Then, the count has been reached, this process is repeated.

Thus, when the count value is odd, the output is high for (N + 1)/2 and low for (N − 1)/2.

The minimum count of this mode is 2.

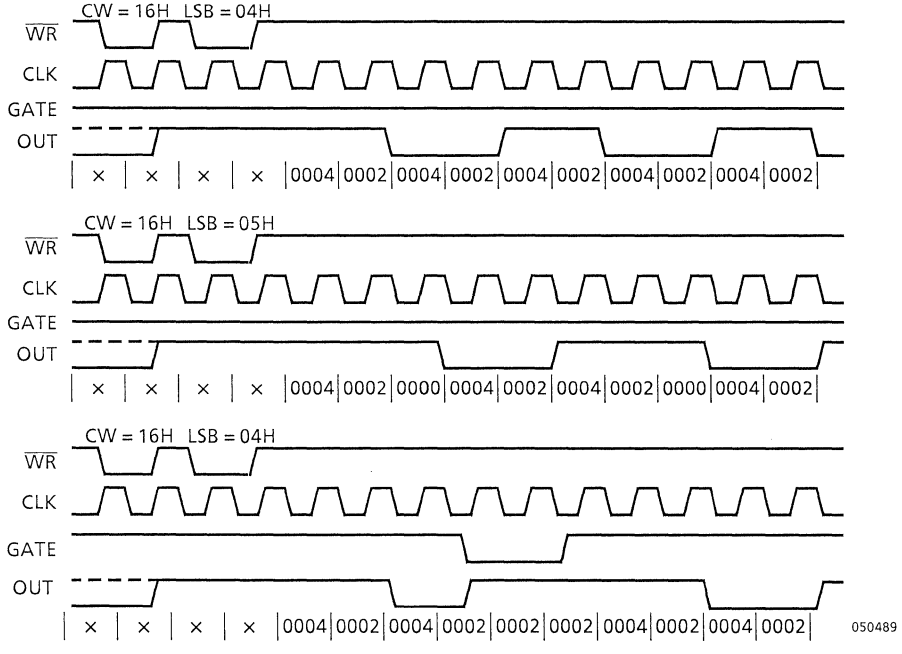


Figure 7.4 Mode 3

7.5 MODE 4: SOFTWARE-TRIGGERED STROBE

The output goes high after setting this mode. When the count value is loaded into the down counter and the GATE input is high, the down-counter starts counting.

When the terminal count has been reached, the output goes low for 1 clock pulse.

When GATE=0, counting is disabled. When GATE=1, counting is enabled. The GATE input dose not affect the output.

When the count value is written to the count register after setting the mode, count loading is performed with the next clock pulse.

When a new count value is written during counting, loading is performed with the following clock pulse and the counting continues with the new count value.

In the 2-byte mode, the following actions take place.

1. Writing of the first byte does not affect the counting.
2. With the clock pulse following the writing of the second byte, count loading starts.

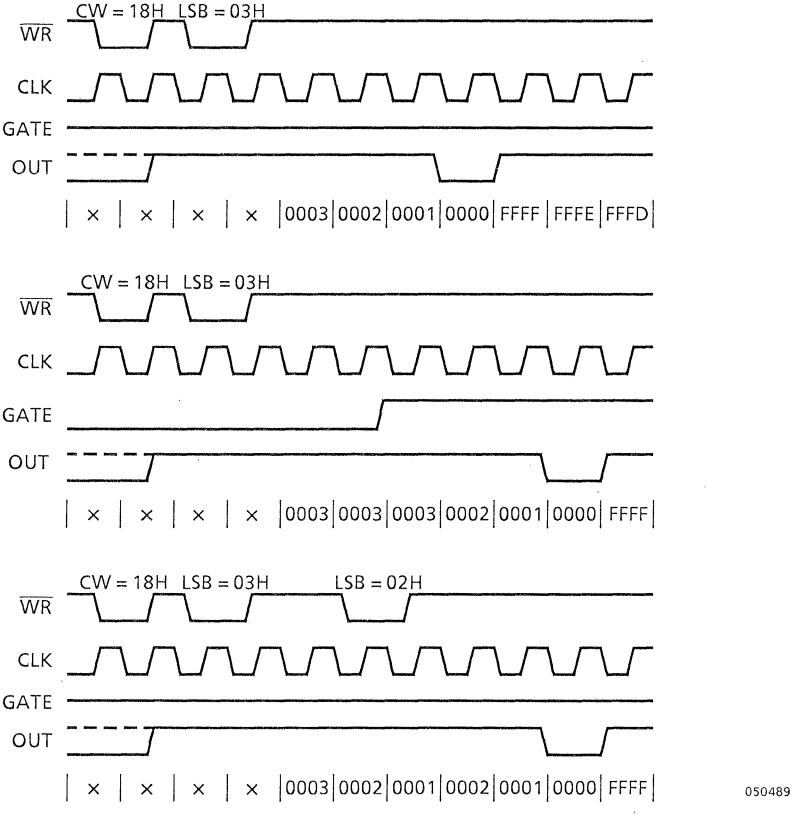


Figure 7.5 Mode 4

7.6 MODE 5: HARDWARE-TRIGGERED STROBE

The output is initialized to the high level after setting this mode. The counter starts counting after the GATE input is triggered. When the terminal count has been reached, the output goes low for 1 clock pulse.

Count loading starts with the clock pulse following the trigger after the mode is set and the count value is written.

The counter can be re-triggered.

The output remains high until a complete count is mode after triggering any GATE input.

If a new count value is written to the count register during counting, the current counting continues. If a trigger is encountered, the new count value is loaded with the following clock pulse and counting starts with the new count value.

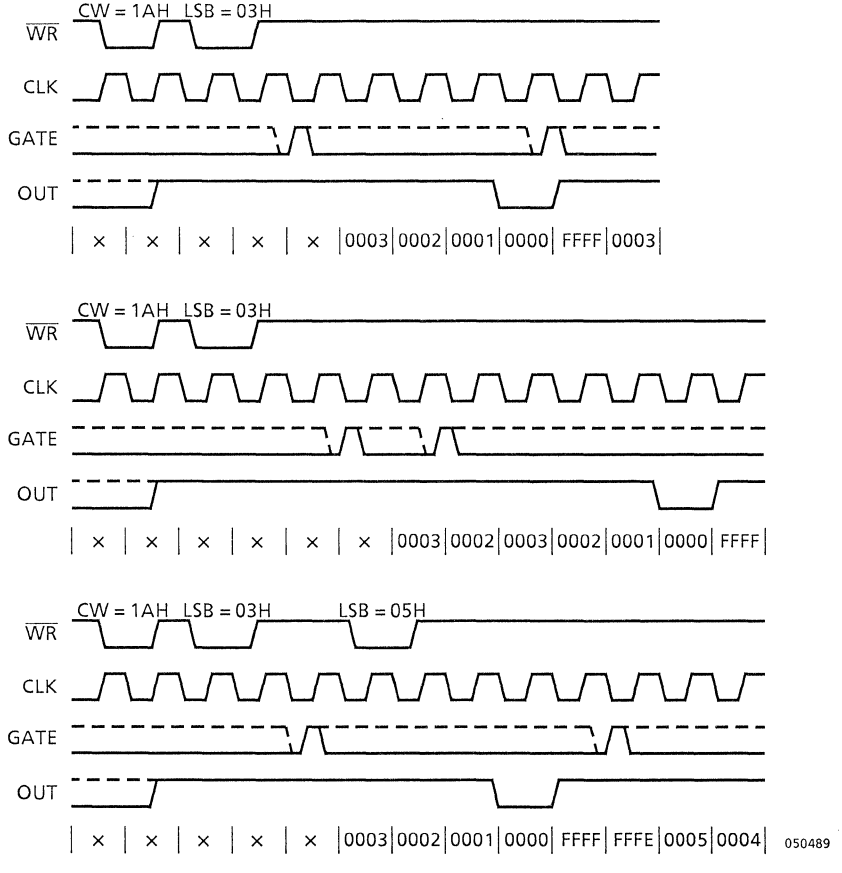


Figure 7.6 Mode 5

Table 7.1 Gate Input Operations

Status Modes	Low or Going Low	Rising	High
0	Disables counting	–	Enable counting
1	–	(1) Initiates counting (2) Resets output after next clock	–
2	(1) Disables counting (2) Sets output immediately High	Initiates counting	Enable counting
3	(1) Disables counting (2) Sets output immediately High	Initiates counting	Enable counting
4	Disables counting	–	Enable counting
5	–	Initiates counting	–

050489

Table 7.2 Minimum and Maximum Count Values

Mode	Min count	Max count
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0
5	1	0

050489

8. ELECTRIC CHARACTERISTICS

8.1 MAXIMUM RATINGS

SYMBOL	ITEMS	TEST CONDITION	RATING	UNIT
V_{CC}	Suply Voltage	With Respect To GND.	−0.5~+7.0	V
V_{IN}	Input Voltage		−0.5~V_{CC}+0.5	V
V_{OUT}	Output Voltage		−0.5~V_{CC}+0.5	V
P_D	Power Dissipation		250	mW
T_{sol}	Solder Temperature		260 (10sec)	℃
T_{stg}	Storage Temperature		−65~+150	℃
T_{opr}	Operating Temperature		−40~+85	℃

050489

8.2 DC CHARACTERISTICS (TA = −40 to +85°C, VCC = 5V ± 10%, VSS (GND) = 0V)

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		−0.5	—	0.8	V
V_{IH}	Input High Voltage		2.2	—	V_{CC}+0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2.2mA	—	—	0.45	V
V_{OH1}	Output High Voltage	I_{OH} = −400μA	2.4	—	—	V
V_{OH2}	Output HIgh Voltage	I_{OH} = −100μA	V_{CC}−0.8	—	—	V
I_{IL}	Input Leak Current	$0V \leqq V_{IN} \leqq V_{CC}$	—	—	±10	μA
I_{OFL}	Output Leakage Current	$0.45V \leqq V_{OUT} \leqq V_{CC}$	—	—	±10	μA
I_{CC1}	Operating Supply Current	CLK = 10MHz $V_{IH} = V_{CC} - 0.2V$ $V_{IL} = 0.2V$	—	—	30	mA
I_{CC2}	Stand-by Supply Current	CLK = DC $V_{IH} = V_{CC} - 0.2V$ $V_{IL} = 0.2V$	—	—	10	μA

050489

8.3 AC CHARACTERISTICS (TA = − 40 to + 85°C, VCC = 5V ± 10%, VSS(GND) = 0V)

8.3.1 Read/Write

SYMBOL	ITEM	P-2/M-2		UNIT
		MIN.	MAX.	
t_{AR}	Address Set-up Time ($\overline{RD}$)	30	—	ns
t_{SR}	$\overline{CS}$ Set-up Time ($\overline{RD}$)	0	—	ns
t_{RA}	Address Hold Time ($\overline{RD}$)	20	—	ns
t_{RR}	$\overline{RD}$ Pulse Width	95	—	ns
t_{RD}	Valid Data ($\overline{RD}$)	—	85	ns
t_{AD}	Valid Data (Address)	—	185	ns
t_{DF}	Data Floating ($\overline{RD}$)	5	65	ns
t_{AW}	Address Set-up Time ($\overline{WR}$)	0	—	ns
t_{SW}	$\overline{CS}$ Set-up Time ($\overline{WR}$)	0	—	ns
t_{WA}	Address Hold Time ($\overline{WR}$)	0	—	ns
t_{WW}	$\overline{WR}$ Pulse Width	95	—	ns
t_{DW}	Data Set-up Time ($\overline{WR}$)	95	—	ns
t_{WD}	Data Hold Time ($\overline{WR}$)	0	—	ns
t_{RV}	Recovery Time	165	—	ns

050489

8.3.2 Clock/Gate

SYMBOL	ITEM	P-2/M-2		UNIT
		MIN.	MAX.	
t_{CLK}	Clock Period	100	DC	ns
t_{PWH}	CLK High Pulse Width	30	—	ns
t_{PWL}	CLK Low Pulse Width	30	—	ns
t_R	CLK Rise Time	—	25	ns
t_F	CLK Fall Time	—	25	ns
t_{GW}	GATE Width High	50	—	ns
t_{GL}	GATE Width LOW	50	—	ns
t_{GS}	GATE Set-up Time (CLK)	40	—	ns
t_{GH}	GATE Hold Time (CLK)	50	—	ns
t_{OD}	Output Delay From CLK	—	100	ns
t_{ODG}	Output Delay From GATE	—	100	ns
t_{WC}	Count Loading Set-up Time (CLK)	80	—	ns
t_{WG}	$\overline{WR}$ Set-up Time (GATE)	0	—	ns
t_{WO}	Output Delay From Command Write	—	240	ns
t_{CL}	CLK Set-up Time (Count Latch)	68	—	ns

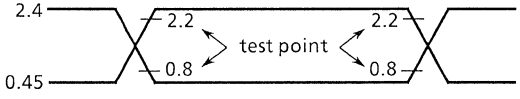
050489

8.4 INPUT CAPACITY (TA = 25°C, VCC = VSS (GND) = 0V)

SYMBOL	ITEMS	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
C_{IN}	Input Capacitance	fc = 1MHz Unmeasured pins, 0V			10	pF
$C_{I/O}$	Inut/Output Capacitance				20	pF

050489

8.5 AC TEST INPUT WAVEFORM



050489

Figure 8.1 AC test input waveform

9. TIMING DIAGRAM

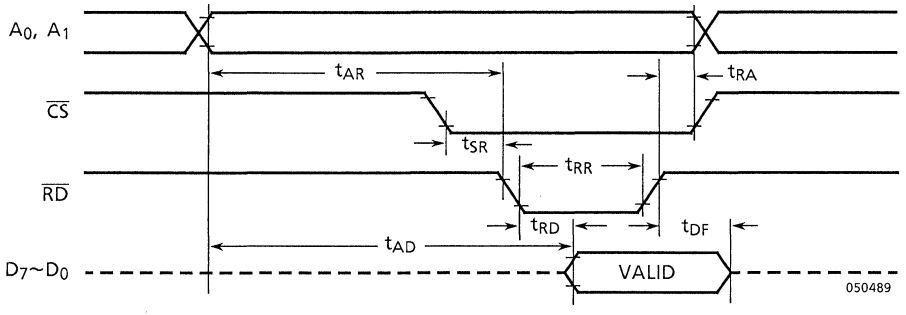


Figure 9.1 Read Operation

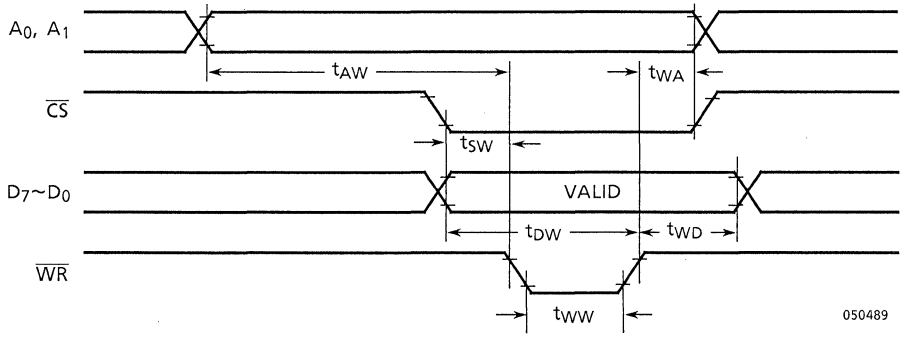


Figure 9.2 Write Operation

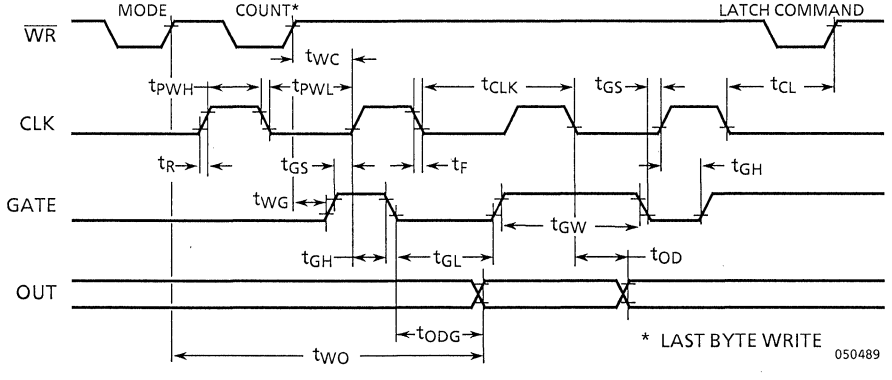


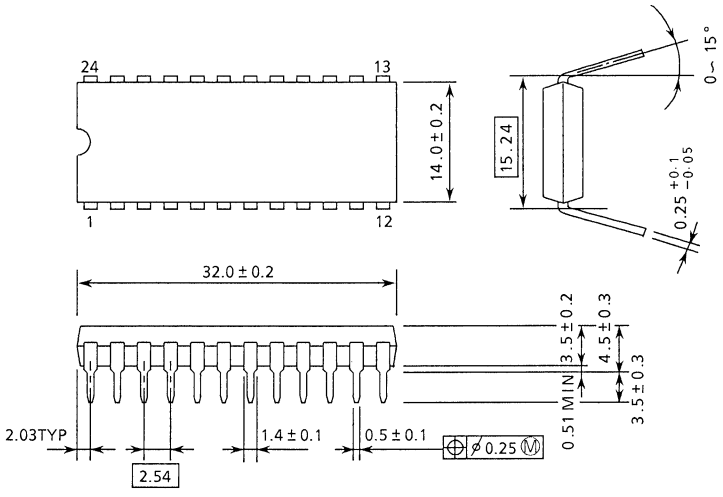
Figure 9.3 Clock and Gate Operations

10. EXTERNAL DIMENSIONS

10.1 24 PIN DIP EXTERNAL DIMENSIONS

DIP24-P-600

Unit : mm

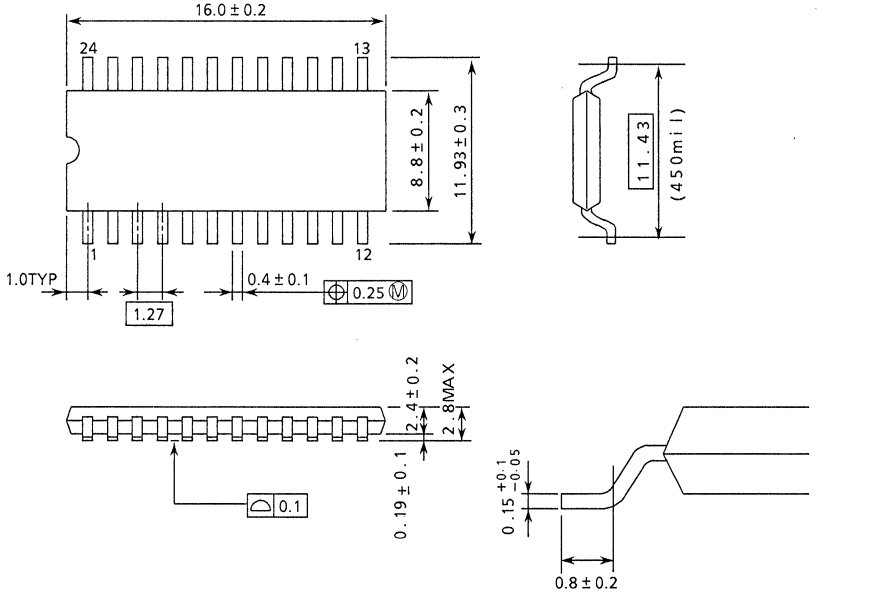


270289

10.2 24 PIN SOP EXTERNAL DIMENSION

SOP24-P-450B

Unit : mm



270289

Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

PROGRAMMABLE INTERVAL TIMER

TMP8253P-5

1. GENERAL DESCRIPTION

The TMP8253P-5 (hereinafter referred to as TMP8253) is a programmable counter/timer chip designed for use as the TLCS-85A microcomputer peripheral. It is organized as 3 independent 16 bit counters, each operates with a count rate of up to 2.6MHz. All modes of operation are software programmable.

2. FEATURES

- Count Binary or BCD
- 3 Independent 16 Bit Counters
- Single +5V Supply
- Count rate DC to 2.6MHz
- 6 programmable Counter Modes
- Compatible with Intel's 8253-5

3. PIN CONNECTIONS

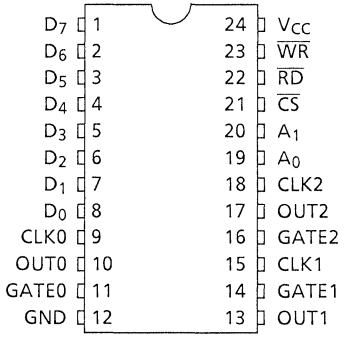
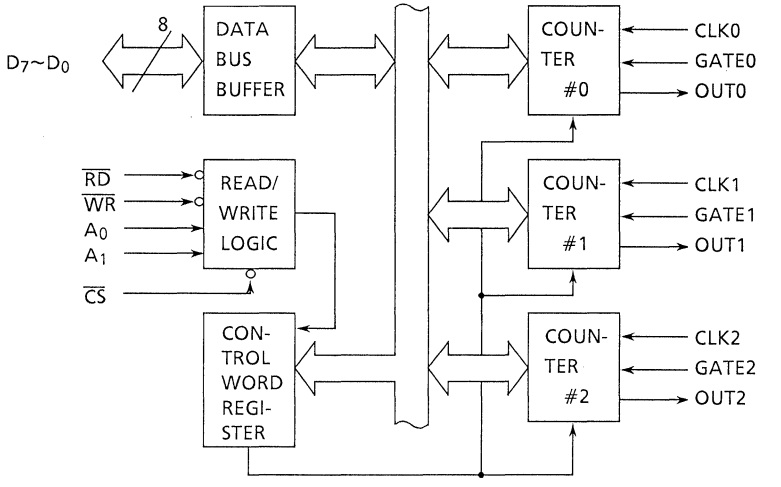


Table 3.1 Pin Names

D_7~D_0	Data Bus (8 bit)
CLK N	Counter Clock Input
GATE N	Counter Gate Input
OUT N	Counter Output
$\overline{RD}$	Read Counter
$\overline{WR}$	Write Command or Data
$\overline{CS}$	Chip Select
A_0~A_1	Counter Select
V_{CC}	+5V
GND	Ground (0V)

050489

4. BLOCK DIAGRAM



050489

5. PIN NAMES AND PIN DESCRIPTION

- GND (Power Supply)

  Ground.

- V_{CC} (Power Supply)

  +5V during operation.

- $\overline{CS}$ (Input)

  A low level input on this pin enables $\overline{RD}$ and $\overline{WR}$ communicaiton between the MPU and the TMP8253. The $\overline{CS}$ input has no effect upon the actual operation of the counters.

- A0, A1 (Input)

  These inputs acts in conjunction with the $\overline{CS}$, $\overline{WR}$, and $\overline{RD}$ pins. This pin is used to select one of the three counters to be operated on and to address the control word register for mode selection.

- $\overline{WR}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP8253 to accept mode information or loading counters from the MPU.

- $\overline{RD}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP8253 to output a counter value onto the data bus for the MPU.

- D0~D7 (Input/Output)

  Bidirectional Data Bus. Mode information, the information loading counter or the count values are transfered via this data bus.

- CLK0~CLK2 (Input)

  Clock inputs to counters. Falling edge on this pin enables the counter to countdown.

- GATE0~GATE2 (Input)

  Gate inputs to counters. The function of this pin differs by the mode selection of counter operation.

- OUT0~OUT2 (Output)

  Outputs from the counters. The output signal from this pin differs by the mode seleciton of counter operation.

6. FUNCTIONAL DESCRIPTION

[Block Description]

Data Bus Buffer

This is 3-state, bidirectional, 8 bit buffer used for interfacing the TMP8253 to the system data bus. The Data Bus Buffer has three functions as follows. Programming the Modes of the TMP8253, Loading the count registers, and Reading the count values.

Read/Write Logic

The Read/Write logic accepts inputs from the system bus and in turn generates control signals for overall device operation.

Table 6.1 Addressing

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	
0	1	0	0	0	Load Counter # 0
0	1	0	0	1	Load Counter # 1
0	1	0	1	0	Load Counter #2
0	1	0	1	1	Write Mode Word
0	0	1	0	0	Read Counter # 0
0	0	1	0	1	Read Counter # 1
0	0	1	1	0	Read Counter # 2
0	0	1	1	1	Data Bus is in High-impedance state
1	×	×	×	×	
0	1	1	×	×	

050489

Control Word Register

The Control Word Reginter is selected when A0, A1 are 11. It then accepts information from the data bus buffer and stores it in a register. The information stored in this register controls the operational Mode of each counter, selection of binary or BCD counting and the loading of each count register. No reading of the contents of the Control Word Register is available.

Counter #0, Counter #1, Counter #2

These three blocks are identical so only a single counter will be described. Each counter consists of a single, 16 bit, presettable, down counter. The counter can operate in either binary or BCD and its input, GATE and output are configured by the selection of Modes (Six MODES : MODE 0 to MODE 5) stored in the Control Word Register. Also the control word handles the loading of the count value so that software overhead can be minimized for these functions.

The reading of the contents of each counter is available to the programmer with simple read operations for event counting applications. Special commands and logic are included in the TMP8253 so that the contens of each count value can be read "on-the-fly" without having to inhibit the clock input.

[MODE Definition]

MODE 0 : Interrupt on Terminal Count

The output will be initially low after the mode set operation. After the count is loaded into the selected count register, the output will remain low and the counter will count. When terminal count is reached the output will go high and remain high until the selected count register is reloaded with the mode or a new count is loaded. The counter continues to decrement after terminal count had been reached.

Rewriting a counter register during counting results in the following :

(1) Write 1st byte stops the current counting.

(2) Write 2nd byte starts the new count.

MODE 1 : Programmable One Shot

The output will go low on the count following the rising edge of the GATE input.

The output will go high on the terminal count. If a new count value is loaded while the output is low it will not affect the duration of the one-shot pulse until the succeeding trigger. The current count can be read at any time without affecting the one-shot pulse.

The one-shot is retriggerable, hence the output will remain low for the full count after any rising edge of the GATE input.

MODE 2 : Rate Generater

Divide by N counter. The output will be low for one period of the input clock. The period from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses the present period will not be affected, but the subsequent period will reflect the new value.

The GATE input, when low, will force the output high. When the GATE input goes high, the counter will start form the initial count. Thus, the GATE input can be used to synchronize the counter.

When this mode is set, the output will remain high until after the count register is loaded. The output then can also be synchronized by software.

MODE 3 : Square Wave Rate Generator

Similar to MODE 2 except that the output will remain high until one half the count has been completed (for even numbers) and go low for the other half of the count.

This is accomplished by decrementing the counter by two on the falling edge of each clock pulse. When the counter reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated.

If the count is odd and the output is high, the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the clock by 2. After timeout, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decriments the counter by 3. Subsequent clock pulses decrement the count by 2 until timeout. Then the whole process is repeated. In this way, if the count is odd, the output will be high for $(N + 1)/2$ counts and low for $(N - 1)/2$ counts.

MODE 4 : Software Triggered Strobe

After the mode is set, the output will be high. When the count is loaded, the counter will begin counting. On terminal count, the output will go low for one input clock period, then will go high again.

If the count register is reloaded between output pulses, counting will continue from the new value, The count will be inhibited while the GATE input is low, Reloading the counter register will restart counting beginning with the new number.

MODE 5 : Hardware Triggered Strobe

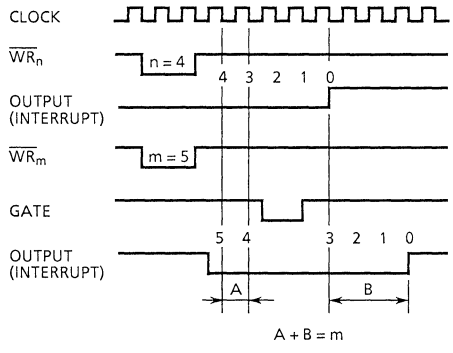
The counter will start counting after the rising edge of the trigger input and will go low for one clock period when the terminal count is reached. The counter is retriggerable. The output will not go low until the full count after the rising edge of any trigger.

Table 6.2 Gate Pin Operaions

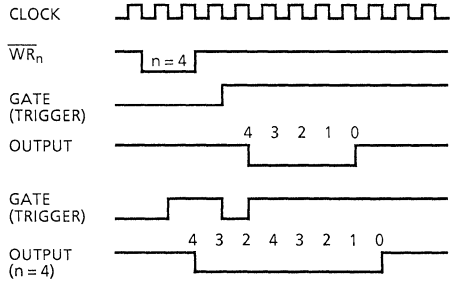
Status Modes	Low or Going Low	Rising	High
0	Disables counting	–	Enables counting
1	–	(1) Initiates counting (2) Resets output after next clock	–
2	(1) Disables counting (2) Sets output immediately High	Initiates counting	Enables counting
3	(1) Disables counting (2) Sets output immediately High	Initiates counting	Enables counting
4	Disables counting	–	Enables counting
5	–	Initiates counting	–

050489

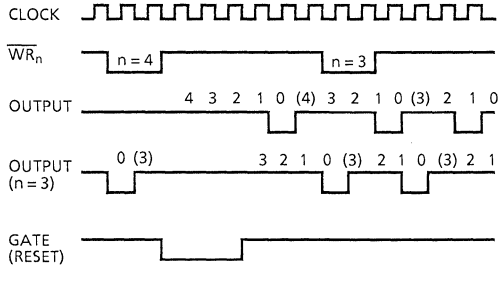
MODE 0 : Interrupt on Terminal Count



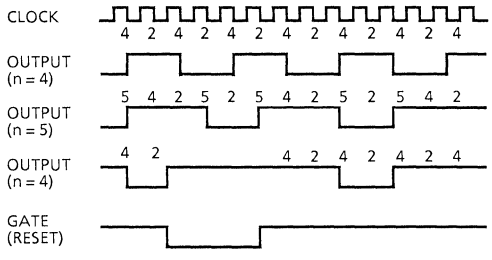
MODE 1: Programmable One-Short



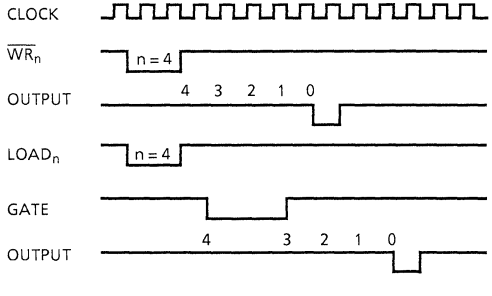
MODE 2 : Rate Generator



MODE 3 : Square Wave Generator



MODE 4 : Software-Triggered Strobe



MODE 5 : Hardware-Triggered Strobe

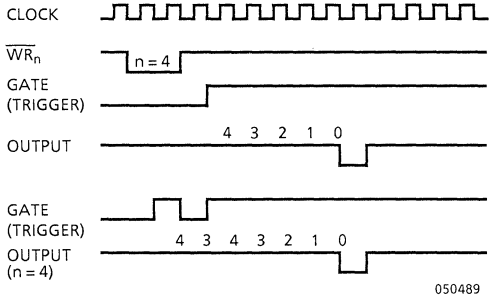
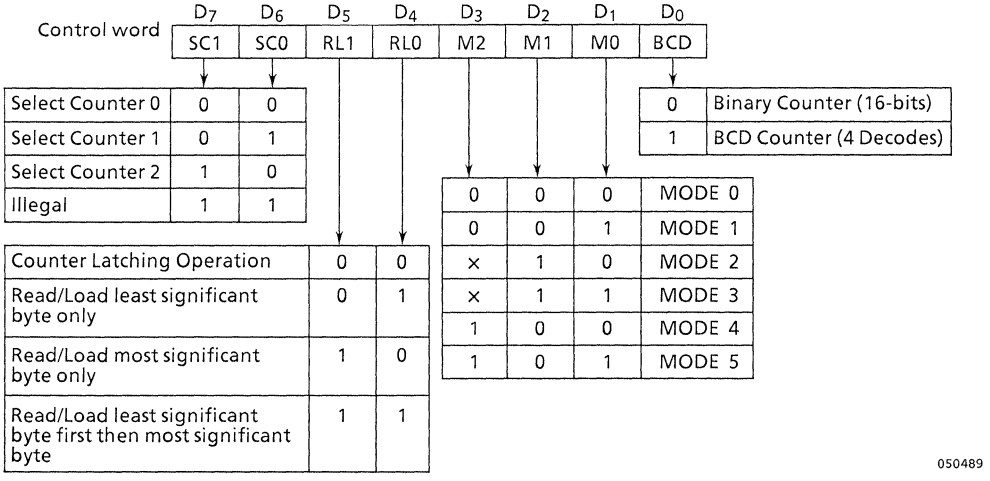


Figure 6.1 TMP8253 Timing Diagrams

7. PROGRAMMING THE TMP8253

All of the Modes for each counter are programmed by the systems software by simple I/O operations.

Each counter of the TMP8253 is individually programmed by writing a control word into the Control Word Register. ($\overline{CS}=0$, A0 = A1 = 1, $\overline{WR}=0$).



Note SC : Select Counter, RL : Read/Load, M : Mode, BCD : Binary Coded Decimal.

The programmer must write out to the TMP8253 a Mode Control Word and the programmed number of count register bytes (1 or 2) prior to actually using the selected counter.

The acctual order of the programming is quite flixible. Writing out ot the Mode Control Word can be in any sequence of counter selection.

The loading of the Count Register with actual count value, however, must be done in exactly the sequence programmed in the Mode Control Word (RL0, RL1).

8. COUNTER LOADING

The count register is not loaded until the count value is written (one or two bytes, depending on the mode selected by the RL bits), followed by a rising edge and a falling edge of the clock.

The count register must be loaded with the number of bytes programmed in the Mode Control Word. The one or two bytes to be loaded in the count register do not have to follow the associated Mode Control Word. They can be programmed at any time following the Mode Control Word loading as long as correct number of bytes is loaded in order.

All counters are down counters. Loading all zeros will result in the maximum count (2^{16} for Binary or 10^4 forBCD). In MODE 0 and MODE 4, the new count will not restart until the load has been completed.

9. READ OPERATIONS

The TMP8253 contains logic that will allow the programmer to easily read the contents of any of the three counters without disturbing the actual count in progress.

There are two methods that the programmer can use to read the value of the counters. The first method involves the use of simple I/O read operations. By controlling the A0, A1 inputs to the TMP8253, the programmer can select the counter to be read. The only requirement with this method is that in order to assure a stable count reading the actual operation of the selected counter must be inhibited either by controlling the GATE input or by external logic that inhibits the clock input.

The contents of the counter selected must be read in the sequence programmed in the Mode Control Word (RL0, RL1). When RL0, RL1 is 11. First I/O read contains the least significant byte (LSB), second I/O read contains the most significant byte (MSB), and the two bytes must be read before any loading WR command can be sent to the same counter.

The second method allows the programmer to read the contents of any counter without affecting or disturbing the counting operation. When the programmer wishes to read the contents of a selected counter "On-the-fly", he loads the MODE register with a special code which latches the present count value into a storage register so that its contents contain an accurate, stable quantity. The programmer then issues a normal read command to the selected counter. The contents of the latched register must be read in the sequence programmed in the Mode Control Word (RL0, RL1). This commands has no effect on the counters mode.

10. PROGRAM EXAMPLE

```
Set up sequence for counter #0
    MVI   A, 00110000B  ..... #0, LSB → MSB, MODE 0, Binary
    OUT   CWAD          ..... The address of Control Word Register
    MVI   A, 53H        ..... LSB for counter #0
    OUT   CNT0          ..... The address of counter #0
    MVI   A, 82H        ..... MSB for counter #0
    OUT   CNT0          ..... The address of counter #0
    :
READ the contents of counter #0
    MVI   A, 0000XXXXB
    OUT   CWAD          ..... Latching count
    IN    CNT0          ..... Read LSB of counter #0
    MOV   L, A
    IN    CNT0          ..... Read MSB of counter #0
    MOV   H, A
    :
RELOAD to counter #0
    MVI   A, 82H        .....
    OUT   CNT0          ..... Load LSB for counter #0
    MVI   A, 53H        .....
    OUT   CNT0          ..... Load MSB for counter #0
```

11. ELECTRIC CHARACTERISTICS

11.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING
V_{CC}	V_{CC} Suply Voltage (with respect to GND (V_{SS}))	− 0.5V to + 7.0V
V_{IN}	Input Voltage (with respect to GND (V_{SS}))	− 0.5V to + 7.0V
V_{OUT}	Output Voltage (with respect to GND (V_{SS}))	− 0.5V to + 7.0V
P_D	Power Dissipation	1W
T_{sol}	Soldering Temperature (Soldering Time 10 sec)	260°C
T_{stg}	Storage Temperature	− 65°C to + 150°C
T_{opr}	Operating Temperature	0°C to 70°C

050489

11.2 DC CHARACATERISTICS (Ta = 0°C to 70°C, V_{CC} = 5.0V ± 5%, GND = 0V)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		− 0.5		0.8	V
V_{IH}	Input High Voltage		2.2		V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2.2mA			0.45	V
V_{OH}	Output High Voltage	I_{OH} = − 400μA	2.4			V
I_{IL}	Input Leak Current	$0V \leqq V_{IN} \leqq V_{CC}$			± 10	μA
I_{OFL}	Output Leak Current	$0.45V \leqq V_{OUT} \leqq V_{CC}$			± 10	μA
I_{CC}	Supply Current				140	mA

050489

11.3 INPUT CAPACITANCE (Ta = 25°C, V_{CC} = GND = 0V)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
C_{IN}	Input Capacitance	f_C = 1 MHz Unmeasured pins, 0V			10	pF
$C_{I/O}$	Inputy/Output Capacitance				20	pF

050489

11.4 AC CHARACTERISTICS

(Ta = 0°C to 70°C, V_{CC} = 5.0V ± 5%, GND = 0V)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
t_{AR}	Address Set up Time ($\overline{RD}\downarrow$)		30			ns
t_{RA}	Address Hold Time ($\overline{RD}\uparrow$)		5			ns
t_{RR}	$\overline{RD}$ Pulse Width		300			ns
t_{RD}	Valid Data ($\overline{RD}\downarrow$)	C_L = 150pF			200	ns
t_{DF}	Data Floating ($\overline{RD}\uparrow$)		25		100	ns
t_{RV}	Recovery Time		1			μs
t_{AW}	Address Set up Time ($\overline{WR}\downarrow$)		30			ns
t_{WA}	Address Hold Time ($\overline{WR}\uparrow$)		30			ns
t_{WW}	$\overline{WR}$ Pulse Width		300			ns
t_{DW}	Data Set up Time ($\overline{WR}\uparrow$)		250			ns
t_{WD}	Data Hold Time ($\overline{WR}\uparrow$)		30			ns
t_{CLK}	Clock Period		380		DC	ns
t_{PWH}	CLK High Pulse width		230			ns
t_{PWL}	CLK Low Pulse Width		150			ns
t_{GW}	GATE Width High		150			ns
t_{GL}	GATE Width Low		100			ns
t_{GS}	GATE Set up Time (CLK↑)		100			ns
t_{GH}	GATE Hold Time (CLK↑)		50			ns
t_{OD}	Output Delay From (CLK↓)	C_L = 150pF			400	ns
t_{ODG}	Output Delay From (GATE↓)	C_L = 150pF			300	ns

050489

Note : AC timings measured at V_{OH} = 2.2V, V_{OL} = 0.8V

12. TIMING DIAGRAM

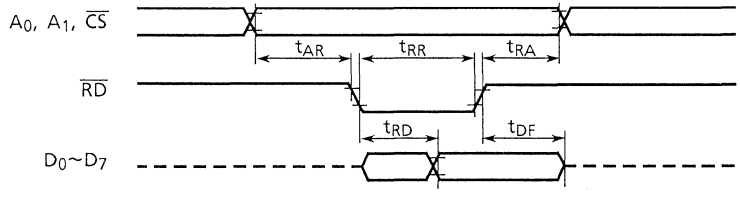


Figure 12.1 Read Timing

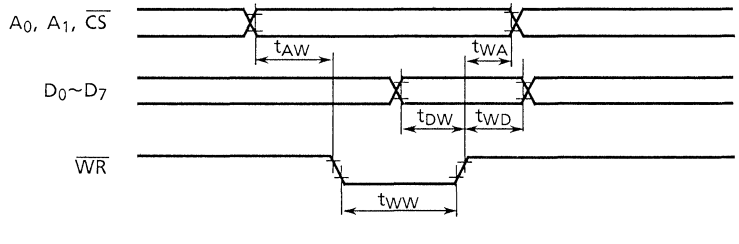


Figure 12.2 Write Timing

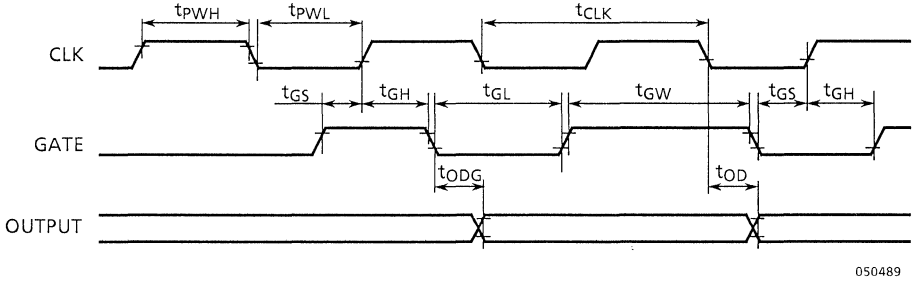


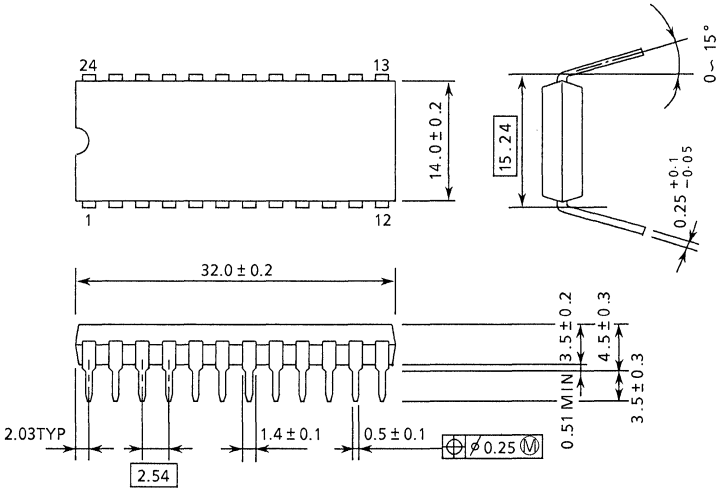
Figure 12.3 Clock & Gate Timing

13. EXTERNAL DIMENSION VIEW

13.1 24PIN DIP

DIP24-P-600

Unit : mm



270289

Note : Lead pitch is 2.54mm and tolerance is ±0.25mm against theoretical center of each lead that is obtained on the basis of No. 1 and No. 24 leads.

CMOS PROGRAMMABLE PERIPHERAL INTERFACE

TMP82C55AP-2/TMP82C55AM-2
TMP82C55AP-10/TMP82C55AM-10

1. GENERAL DESCRIPTION AND FEATURES

The TMP82C55A (hereinafter referred to as PPI) is a CMOS high Speed programmable input/output interface with three 8-bit I/O ports. 24 I/O Ports are divided into two groups (Port A and Port B) which are programmable independently by control words provided by MPU. The PPI has three operation modes (Mode 0, 1 and 2) and is copable of versatile interface between MPU and peripheral devices.

The TMP82C55A is fabricated using Toshiba's CMOS Silicon Gate Technology.

(1) High Speed Version (TRD = 100ns MAX: TMP82C55AP-10/AM-10)

(2) Low power consumption
2mA Type.
10μA Max. (@5V, Stand-by)

(3) 5V ±10% Single power supply

(4) 24 programmable I/O ports

(5) Three operation modes (Mode 0, Mode 1, Mode 2)

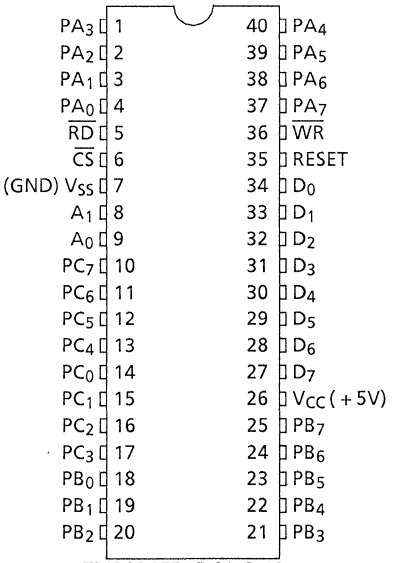
(6) Bit set/reset capability

(7) Up to 8 output ports of port B and C are capable of driving a darlington transistor (Min. −1.0mA @VOH = 1.5V)

(8) Extended operating temperature: −40 °C to +85 °C

(9) Available 40pin Standard DIP and SOP

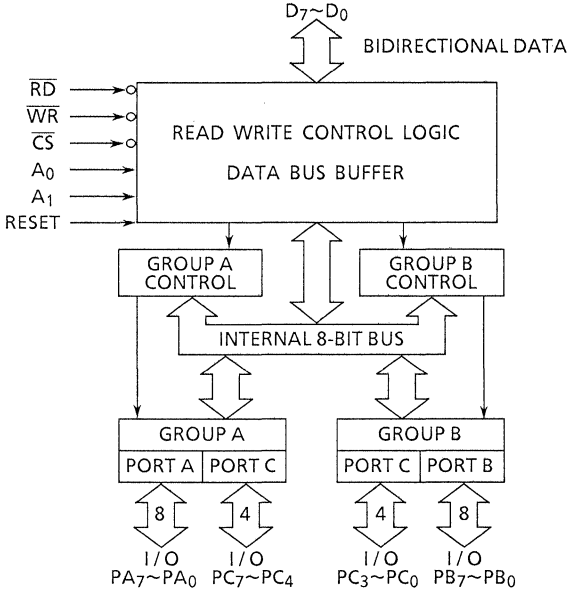
2. PIN CONNECTIONS (TOP VIEW)



TMP82C55AP-2/AP-10
TMP82C55AM-2/AM-10

050489

3. BLOCK DIAGRAM



050489

4. PIN NAMES AND PIN FUNCTIONS

Pin Name	Number of Pin	Input/Output 3-state	Function
D_0~D_7	8	I/O 3-state	3-state bidirectional 8-bit data bus. Used for data transfer with MPU. Also, used for transfer of control words to PPI and status information from PPI.
PA_0~PA_7	8	I/O 3-state	3-state 8-bit I/O Port A. Operation mode and input/output configuration are defined by software. Port A contains the output latch buffer and input latch.
PB_0~PB_7	8	I/O 3-state	3-state 8-bit I/O Port B. Operation mode and input/output configuration are defined by software. Port B contains the output latch buffer and input latch.
PC_0~PC_7	8	I/O 3-state	3-state 8-bit I/O Port C. Operation mode and input/output configuration are defined by software. Port C can be divided into two 4-bit ports by the mode control and also, used as the control signal for Port A and Port B. In this case, 3 bits of PC_0 to PC_2 are used for Port B and 5 bits of PC_3 to PC_7 for Port A.
$\overline{CS}$	1	Input	Chip select input. When this terminal is at "L" level, data transfer PPI and MPU becomes possible. At "H" level, the data bus is placed in the high impedance state and control from the processor is ignored.
$\overline{RD}$	1	Input	Read signal. When this terminal is at "L" level, data that is input into the port is transferred to MPU.
$\overline{WR}$	1	Input	Write signal. When this terminal is at "L" level, data or control word is written into PPI from MPU.
A_0, A_1	2	Input	Used for selecting Port A, B, C and the control registers. Normally, this terminal is connected to low order 2 bits of the address bus.
RESET	1	Input	When this terminals is at "H" level, all internal registers including the control register are cleared. In addition, all ports (Port A, B, C) are placed in the input mode (high impedance) of mode 0.
V_{CC}	1	Power Supply	5V
V_{SS}	1	Power Supply	GND

050489

5. FUNCTIONAL DESCRIPTION

The PPI is a programmable peripheral interface device with three 8-bit ports (Port A, B and C) and two control registers. 24 I/O ports are divided into 12-bit group A and group B. Group A consists of Port A and high order 4 bits of Port C, while Group B consists of Port B and low order 4 bits of Port C. Each group is independently programmable by control words provided from MPU. There are three operation modes available for the PPI. In mode 0, two 8-bit I/O ports and two 4-bit I/O ports can be programmed as input or output ports, respectively. In mode 1, 24 I/O ports are divided into Group A and Group B. 8 bits of each group are used as input or output port and of the remaining 4 bits, 3 bits are used as handshaking and interrupt control signal. Mode 2 is applicable only to group A and the ports are used as a bidirectional 8-bit data bus and 5-bit control signal. In case of Port C being used as the output, any bits of Port C can be set/reset.

There are two control registers; one is used for mode setting and the other for bit set/reset control. The control registers can only be written into. Further, when the reset input (RESET) becomes "1", the control registers are reset and all I/O ports are placed in input mode (high impedance status).

Table 5.1 Basic Operation of TMP82C55A

A_1	A_0	$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	Function
0	0	0	0	1	Data bus ← Port A
0	1	0	0	1	Data bus ← Port B
1	0	0	0	1	Data bus ← Port C
0	0	0	1	0	Port A ← Data bus
0	1	0	1	0	Port B ← Data bus
1	0	0	1	0	Port C ← Data bus
1	1	0	1	0	Control register ← Data bus
×	×	1	×	×	Data bus = 3-state
×	×	0	1	1	Data bus = 3-state
1	1	0	0	1	inhibition of combination

050489

5.1 MODE SELECTION

There are three basic modes of operation that can be selected by control words.

Mode 0-Basic I/O (Group A, Group B)
Mode 1-Strobe input/Strobe output (Group A, Group B)
Mode 2-Two-way bus (Port A only)

Operation modes for Group A and Group B can be independently defined by the control word form the MPU. If D_7 is set to "1" in writing a control word into the PPI, operation mode is selected, while of D_7 = "0", the set/reset function for Port C is selected.

5.1.1 Control word to define operation mode

Figure 5.1 shows the control words to define operation mode of the TMP82C55A.

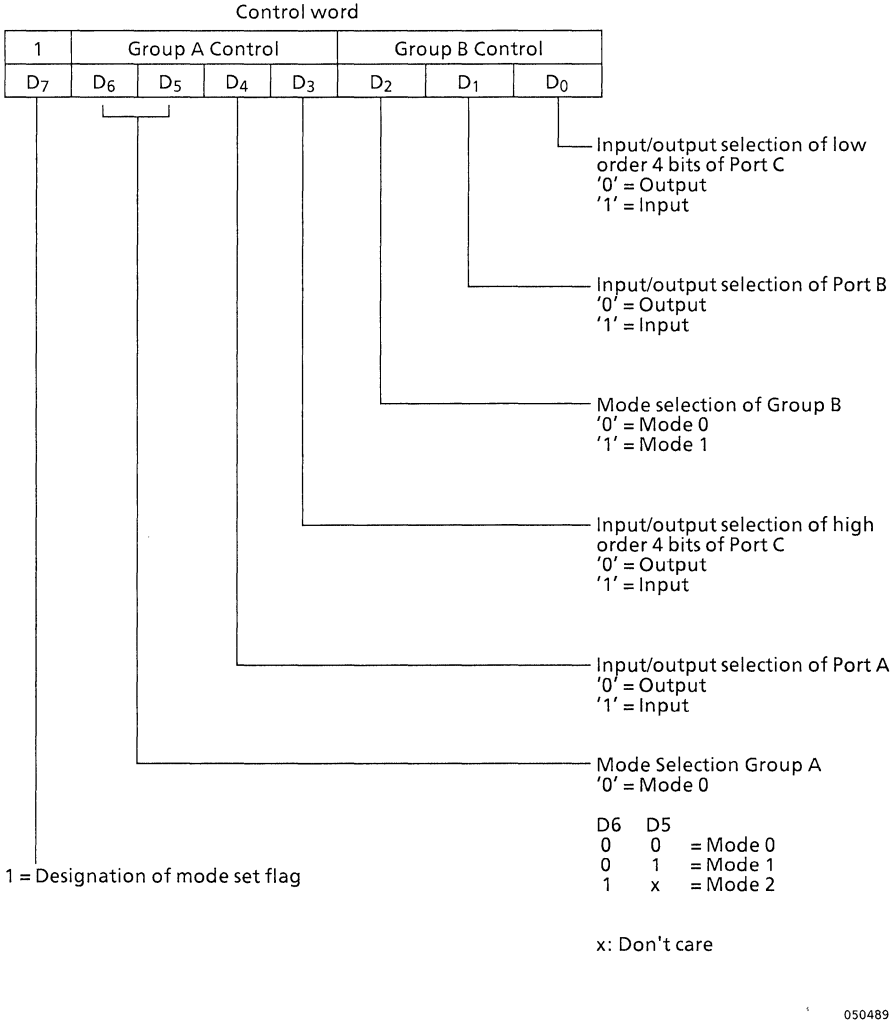


Figure 5.1 Control Word for Mode Selection

5.1.2 Port C bit set/reset control word

Any bit of 8 bits of Port C can be set/reset by Port C bit set/reset control word. Fig. 5.2 shows the Port C bit set/reset control word.

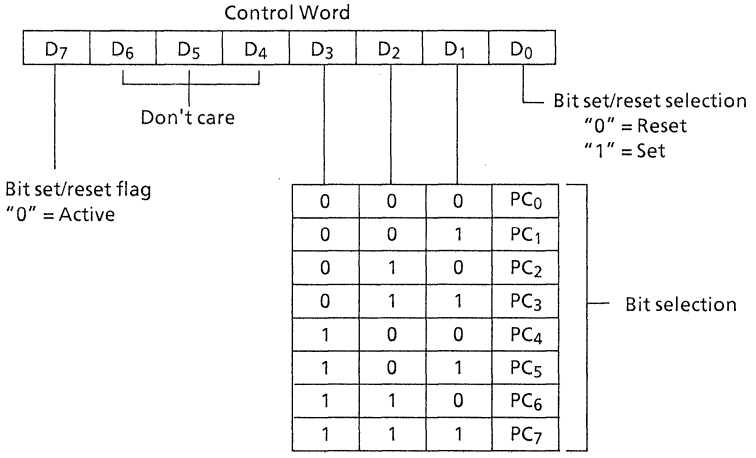


Figure 5.2 Control Word for Bit Set/Reset

5.2 OPERATION MODES

5.2.1 Mode 0 (Basic I/O)

This functional configuration is used for simple input or output operations. No 'handshaking' is required and data is simply written to or read from a specified part. Output data to the ports from MPU are latched out but input data from the ports are not latched.

In Mode 0, 24 I/O terminals are divided into four groups of Port A (8 bits), Port B (8 bits), high order 4 bits of Port C and low order 4 bits of Port C. Each port can be programmed to be input or output. The configuration of each port are determined according to the contents of Bit 4 (D_4), 3 (D_3), 1 (D_1) and 0 (D_0) of the control word for mode selection.

The I/O configuration of each port in Mode 0 are shown in Table 5.2.

Mode Setting Control Word				Port A	Port C (PC_7~PC_4)	Port B	Port C (PC_3~PC_0)
D_4	D_3	D_1	D_0				
0	0	0	0	Out	Out	Out	Out
0	0	0	1	Out	Out	Out	In
0	0	1	0	Out	Out	In	Out
0	0	1	1	Out	Out	In	In
0	1	0	0	Out	In	Out	Out
0	1	0	1	Out	In	Out	In
0	1	1	0	Out	In	In	Out
0	1	1	1	Out	In	In	In
1	0	0	0	In	Out	Out	Out
1	0	0	1	In	Out	Out	In
1	0	1	0	In	Out	In	Out
1	0	1	1	In	Out	In	In
1	1	0	0	In	In	Out	Out
1	1	0	1	In	In	Out	In
1	1	1	0	In	In	In	Out
1	1	1	1	In	In	In	In

050489

Figure 5.3 Port Definition in Mode 0

5.2.2 Mode 1 (Strobe I/O)

In Mode 1, input/output of port data is performed in conjunction with the strobe signals or 'handshaking' signals. Port C is used to control Port A or Port B.

The basic operatings in Mode 1 are as follows:

- Mode 1 can be set for two groups of Group A and Group B.
- Each group consist of 8-bit data port and 4-bit control/data port.
- The 8-bit data port can be set as input or output port.
- The control/data port is used as control or status of the 8-bit data port.

(1) When used as the input port in Mode 1:

- $\overline{STB}$ (Strobe Input)

  At "0", input data is loaded in the internal input latch in the port.

  In this case, a control signal from MPU is not concerned and data is input from the port any time. This data is not read out on the data bus unless MPU executes an input instruction.

- IBF (Input Buffer Full F/F Output)

  When data is loaded in the internal input latch from the port, this output is set to "1". IBF is set ("1") by $\overline{STB}$ input being reset and is reset ("0") by the rising edge of $\overline{RD}$ input.

- INTR (Interrupt Request Output)

  Used for the interrupt process of data loaded in the internal input latch. When $\overline{STB}$ input is at "0" if INTE (INTE flag) in the PPI is in the enabled state ("1") , IBF is set to "1". INTR is set to "1" immediately after the rising edge of this $\overline{STB}$ input and reset to "0" by the falling edge of $\overline{RD}$ input.

  The INTE flag of Group A and Group B are controlled as follows:

  INTEA-Control by bit set/reset of PC_4
  INTEB-Control by bit set/reset of PC_2

(2) When used as the output port in Mode 1:

- $\overline{OBF}$ (Output Buffer Full F/F Output)

  This is a flag which shows that MPU has written data into a specified port. $\overline{OBF}$ is set to becomes "0" at the rising edge of $\overline{WR}$ signal and is set to "1" at the falling edge of $\overline{ACK}$ (Acknowledge input) signal.

- $\overline{ACK}$ (Acknowledge Input)

  $\overline{ACK}$ signal is sent to the PPI as a response from a peripheral device taht received data from the port.

- INTR (Interrupt Request Output)

  When a peripheral device received data from MPU, INTR is set to "1" and the interrupt is requested to MPU. If $\overline{ACK}$ signal is received when INTE flag is in the enable state, $\overline{OBF}$ is set to "1" and INTR signal becomes "1" immediately after the rising edge of $\overline{ACK}$ signal. Further, INTR is reset at the falling edge of $\overline{WR}$ signal when data is written into the PPI by MPU.

  The INTE flags of Group A and Group B are controlled as follows:

  INTEA-Control by bit set/reset of PC_6
  INTEB-Control by bit set/reset of PC_2

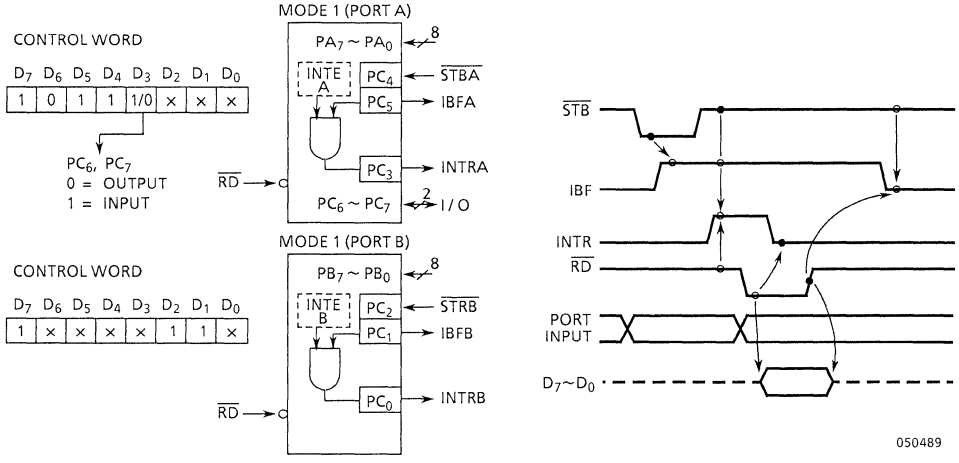
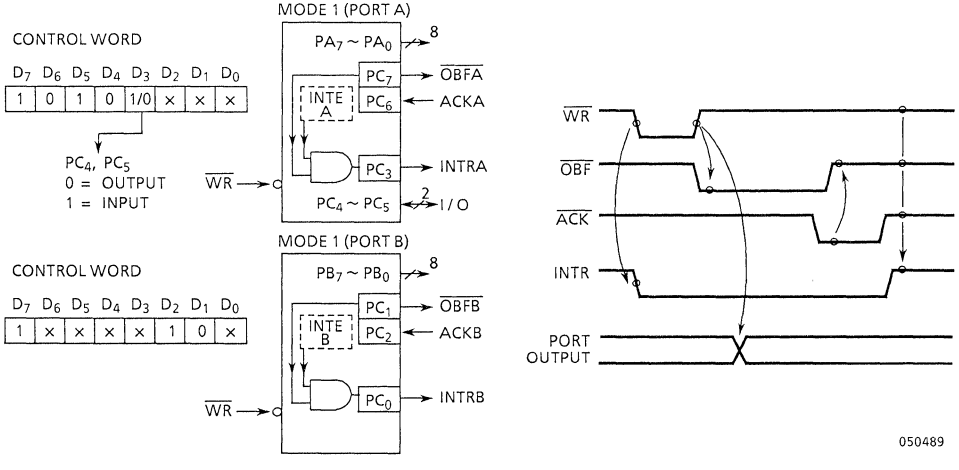


Figure 5.4 Example of Strobe Input in Mode 1



Figuire 5.5 Example of Strobe Output in Mode 1

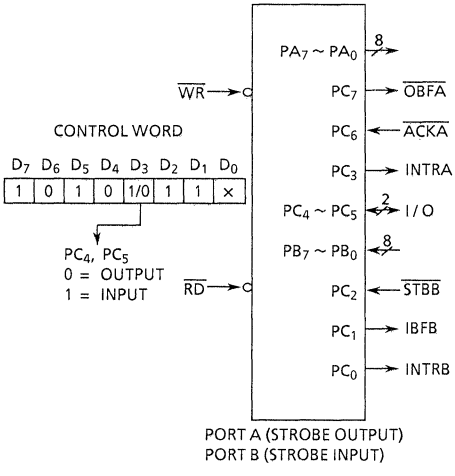
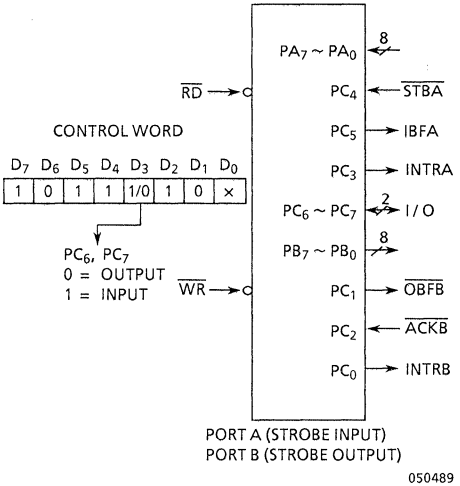


Figure 5.6 Example of Port A Output, Port B Input in Mode 1



050489

Figure 5.7 Example of Port A Input, Port B Output in Mode 1

5.2.3 Mode 2 (Strobed Bidirectional Bus I/O)

In this mode, Port A is used as 8 bits bidirectional bus for data transfer with a peripheral device. This mode is applicable only to Group A, which consists of an 8-bit bidirectional bus (Port A 8-bit) and 5-bit control signals (high order 5 bits of Port C). The bidirectional bus (Port A) has both the internal input and output registers. When group A is set in Mode 2, Group B can be set independently. There are 5 control signals as follows when Group A is used in Mode 2.

- $\overline{OBF}$ (Output buffer Full F/F Output)

  When MPU writes data into of Port A, $\overline{OBF}$ is set to "0" to inform a peripheral device that the PPI is ready to output data. However, Port A is kept in the floating (high impedance) state until $\overline{ACK}$ input signal is received.

- $\overline{ACK}$ (Acknowledge Input)

  When $\overline{ACK}$ signal is set to "0", the data of the 3-state output buffer of Port A is send out. If $\overline{ACK}$ signal is at "1", Port A is in the high impedance state.

- $\overline{STB}$ (Strobe Input)

  When $\overline{STB}$ input is set to "0", the data from peripheral devices are held in the input latch. When the active $\overline{RD}$ signal is input into the PPI, the latched input data are output on the system data bus (D_7-D_0).

- IBF (Input Buffer Full F/F Output)

  When data from peripheral devices are held in the input latch, IBF is set to "1".

- INTR (Interrupt Request Output)

  INTR is the output to request the interrupt to MPU and its function is the same as that in Mode 1. There are two interrupt enable flip-flop (INTE), INTE1 corresponds to INTEA in Mode 1 output and INTE2 to INTEA in Mode 1 input.

  INTE 1-Used to generate INTR signal in conjunction with $\overline{OBF}$ and $\overline{ACK}$ signals, and is controlled by PC_6 bit set/reset.

  INTE 2-Used to generate INTR signal in conjunction with IBF and $\overline{STB}$ signals, and is controlled by PC_4 bit set/reset.

  Fig. 5.8 shows the operating example and the timing diagram in Mode 2.

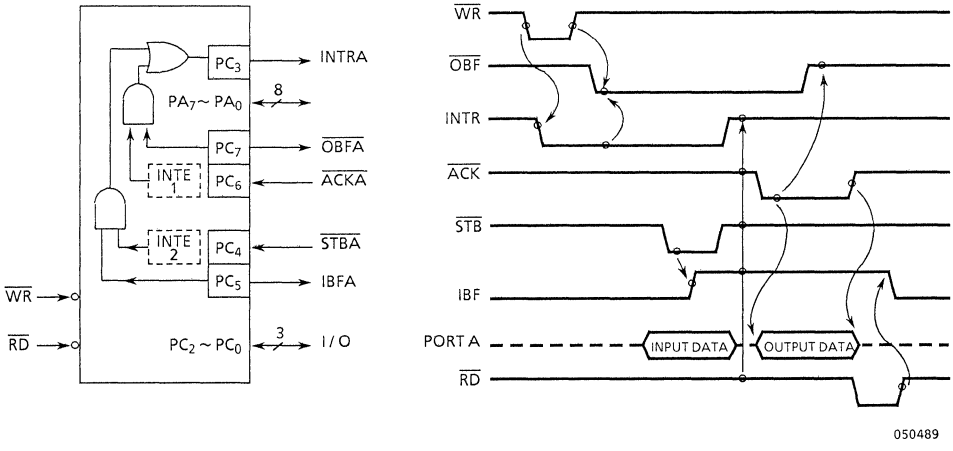


Figure 5.8 Operating Example in Mode 2

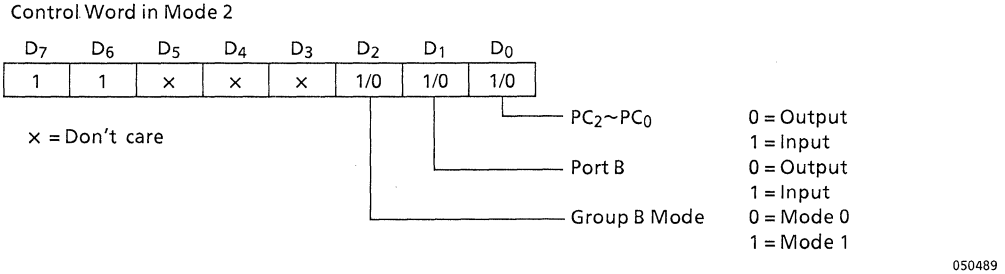


Figure 5.9 Control Word and Configuration in Mode 2

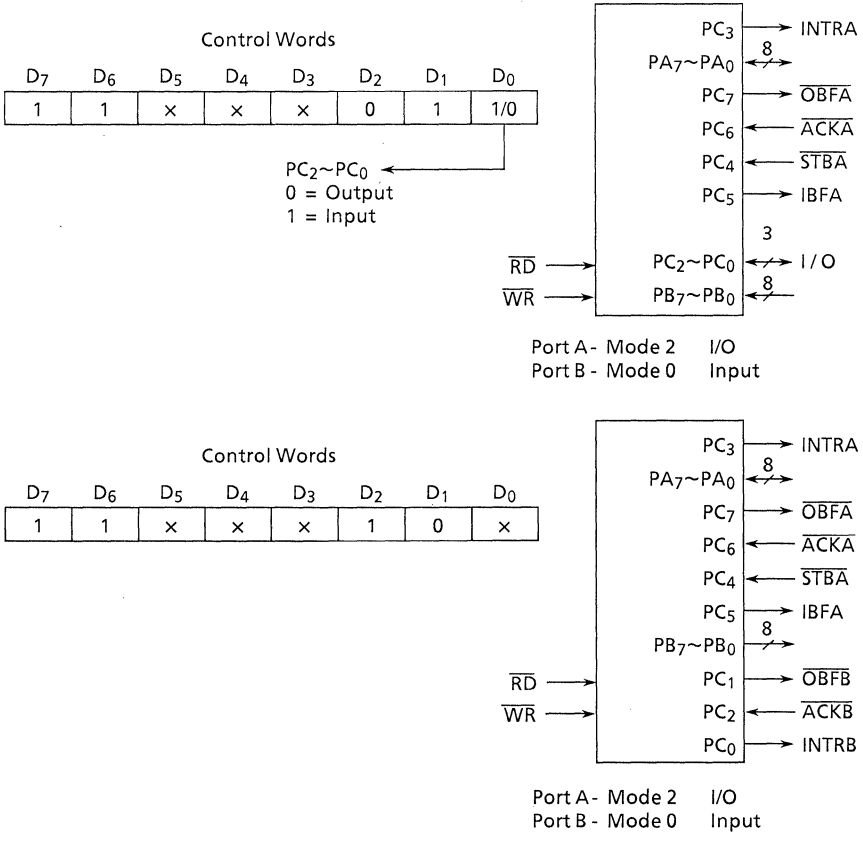


Figure 5.10 Examples in Combination with Mode 2 and Other Mode

5.2.4 Precautions for use in mode 1 and 2

When used in Mode 1 and 2, bits which are not used as control or status in Port C can be used as follow.

If programmed as the input, they are accessed by normal Port C read.

If programmed as the output, high order bits of Port C (PC_7-PC_4) are accessed using the bit set/reset function. As to low order bits of Port C (PC_3-PC_0), in additions to access by the bit set/reset function, only 3 bits can be accessed by normal writing.

5.3 READING PORT C STATUS

When Port C is used as the control port, that is, when Port C is used in Mode 1 or Mode 2, the status information of the control word can be read out by a normal read operation of Port C.

Table 5.2 Status Word Format of Port C

Mode \ Data	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Mode 1 Input	I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB
Mode 1 Output	$\overline{OBFA}$	INTEA	I/O	I/O	INTRA	INTEB	$\overline{OBFB}$	INTRB
Mode 2	$\overline{OBFA}$	INTE1	IBFA	INTE2	INTRA	By Group B Mode		

050489

6. ELECTRICAL CHARACTERISTICS

6.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item	Rating	Unit
V_{CC}	Supply Voltage	− 0.5 to 7.0	V
V_{IN}	Input Voltage	− 0.5 to V_{CC} + 0.5	V
P_D	Power Dissipation	250	mW
T_{SOLDER}	Soldering Temperature (10 sec)	260	℃
T_{STG}	Storage Temperature	− 65 to + 150	℃
T_{OPR}	Operating Temperature	− 40 to + 85	℃

050489

6.2 DC ELECTRICAL CHARACTERISTICS

TA = − 40℃ to + 85℃, V_{CC} = 5V ± 10%, V_{SS} = 0V

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		− 0.5	—	0.8	V
V_{IH}	Input High Voltage		2.2	—	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2.5mA	—	—	0.45	V
V_{OH1}	Output High Voltage	I_{OH} = -400μA	2.4	—	—	V
V_{OH2}	Output High Voltage	I_{OH} = -100μA	V_{CC}-0.8	—	—	V
I_{IL}	Input Leak Current	$0 \leqq V_{IN} \leqq V_{CC}$	—	—	± 10	μA
I_{LO}	Output Leak Current (High Impedance State)	$0 \leqq V_{OUT} \leqq V_{CC}$	—	—	± 10	μA
(Note) I_{DAR}	Darlington Drive Current	V_{EXT} = 1.5V R_{EXT} = 1.1kΩ	− 1.0	—	− 5.0	mA
I_{CC1}	Operating Supply Current	I / Ocycle Time 1μsec	—	2.0	5.0	mA
I_{CC2}	Stand-by Supply Current	$\overline{CS} \geqq V_{CC}$-0.2V $V_{IH} \geqq V_{CC}$-0.2V $V_{IL} \leqq 0.2V$	—	—	10	μA

050489

Note: Applied for optional 8 I/O terminals in Port B and Port C.

6.3 AC ELECTRICAL CHARACTREISTICS

$TA = -40℃$ to $+85℃$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$

SYMBOL	PARAMETER	AP-2 / AM-2		AP-10 / AM-10		UNIT
		MIN.	MAX.	MIN.	MAX.	
t_{AR}	Address set-up time for $\overline{RD}$ fall	0	—	0	—	ns
t_{RA}	Address hold time for $\overline{RD}$ rise	0	—	0	—	ns
t_{RR}	$\overline{RD}$ pulse width	160	—	150	—	ns
t_{RD}	Delay from $\overline{RD}$ fall to decided data output	—	140	—	100	ns
t_{DF}	Time from $\overline{RD}$ rise to data bus floating	0	40	0	40	ns
t_{RV}	Time from $\overline{RD}$ or $\overline{WR}$ rise to next $\overline{RD}$ or $\overline{WR}$ fall	200	—	150	—	ns
t_{AW}	Address set-up time for $\overline{WR}$ fall	0	—	0	—	ns
t_{WA}	Address holding time for $\overline{WR}$ rise	0	—	0	—	ns
t_{WW}	$\overline{WR}$ pulse width	120	—	120	—	ns
t_{DW}	Bus data set-up time for $\overline{WR}$ rise	100	—	100	—	ns
t_{WD}	Bus data holding time for $\overline{WR}$ rise	0	—	0	—	ns
t_{WB}	Delay from $\overline{WR}$ rise to decided data output	—	350	—	350	ns
t_{IR}	Port data set-up time for $\overline{RD}$ fall	0	—	0	—	ns
t_{HR}	Port data holding time for $\overline{RD}$ rise	0	—	0	—	ns
t_{AK}	$\overline{ACK}$ pulse width	300	—	300	—	ns
t_{ST}	$\overline{STB}$ pulse width	350	—	350	—	ns
t_{PS}	Port data set-up time for $\overline{STB}$ rise	0	—	0	—	ns
t_{PH}	Port data holding time for $\overline{STB}$ rise	150	—	150	—	ns
t_{AD}	Delay from $\overline{ACK}$ fall to decided data output	—	300	—	300	ns
t_{KD}	Time from $\overline{ACK}$ rise up to port (Port A in Mode 2) floating	25	250	20	250	ns
t_{WOB}	Delay from $\overline{WR}$ rise to $\overline{OBF}$ fall	—	300	—	300	ns
t_{AOB}	Delay from $\overline{ACK}$ fall to $\overline{OBF}$ rise	—	350	—	350	ns
t_{SIB}	Delay from $\overline{STB}$ fall to IBF rise	—	300	—	300	ns
t_{RIB}	Delay from $\overline{RD}$ fall to $\overline{IBF}$ rise	—	300	—	300	ns
t_{RIT}	Delay from $\overline{RD}$ fall to INTR fall	—	400	—	400	ns
t_{SIT}	Delay from $\overline{ACK}$ rise to INTR rise	—	300	—	300	ns
t_{AIT}	Delay from $\overline{ACK}$ rise to INTR rise	—	350	—	350	ns
t_{WIT}	Delay from $\overline{WR}$ rise to INTR	—	450	—	450	ns

050489

Note : 1. When the power supply is turned ON, reset pulse duration must be active for at least 500 ns or more.

2. AC Measuring Point Input Voltage $V_{IH} = 2.4V$, $V_{IL} = 0.45V$
Output Voltage $V_{OH} = 2.2V$, $V_{OL} = 0.8V$
$CL = 150pF$.

6.4 CAPACITANCE

$T_A = 25℃$, $V_{CC} = V_{SS} = 0V$

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
C_{IN}	Input Capacitance	f_C = 1MHz (*)	—	—	10	pF
C_{OUT}	Output Capacitance		—	—	20	pF

050489

(*) : All terminals except that to be measured should be earthed.

7. TIMING DIAGRAM

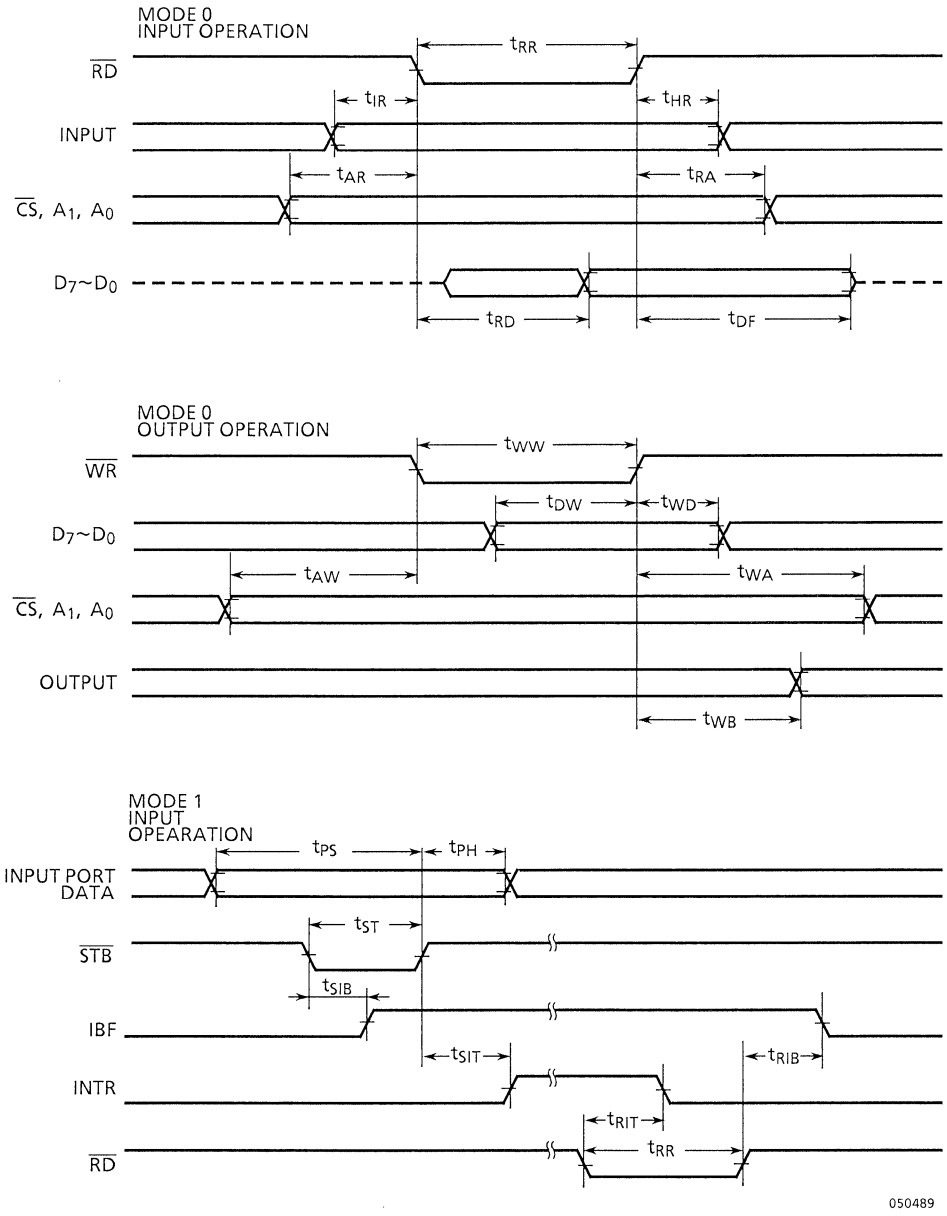


Figure 7.1 Timing Diagram

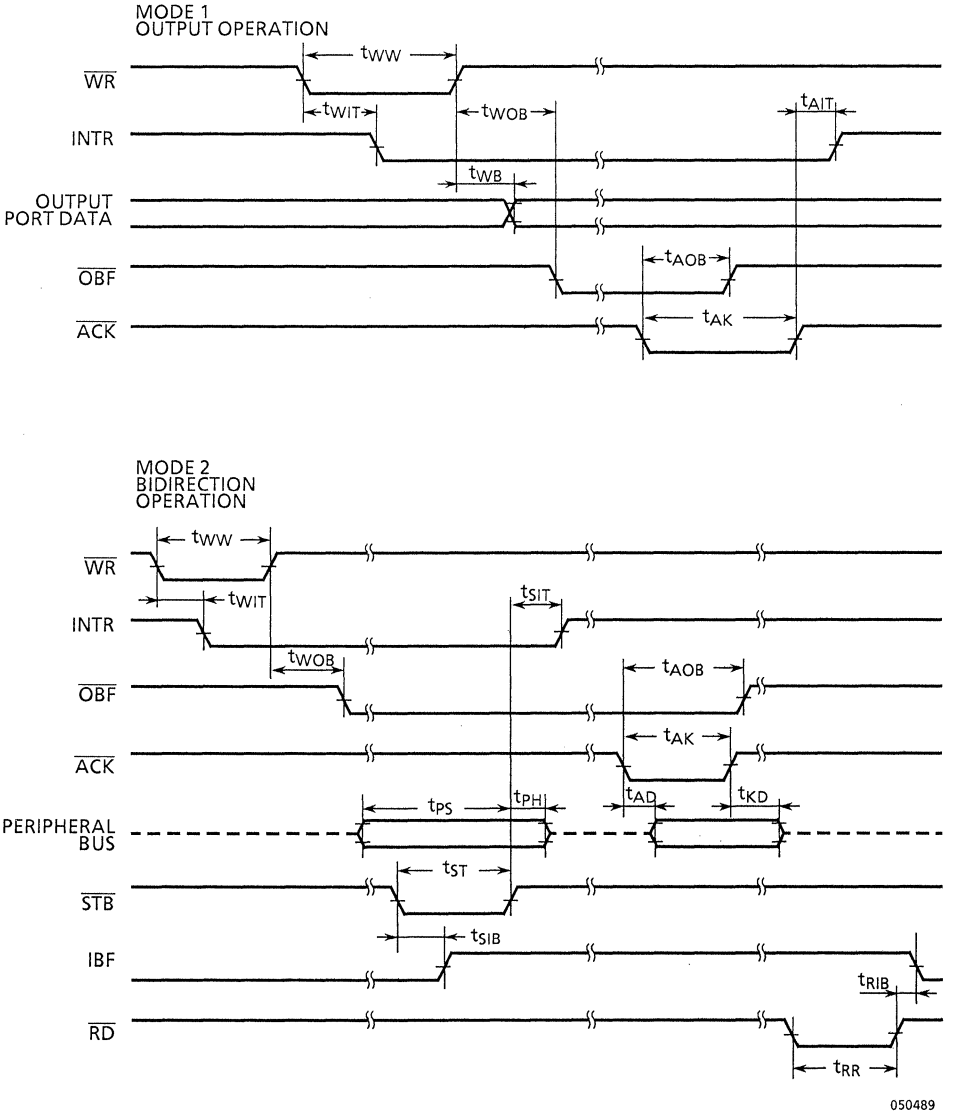


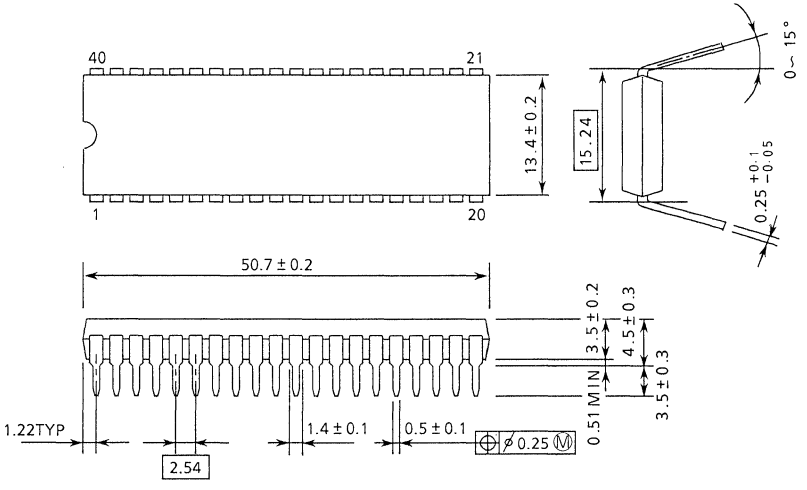
Figure 7.2 Timing Diagram

8. PACKAGE DIMENSION

8.1 PLASTIC PACKGE

DIP40-P-600

Unit : mm



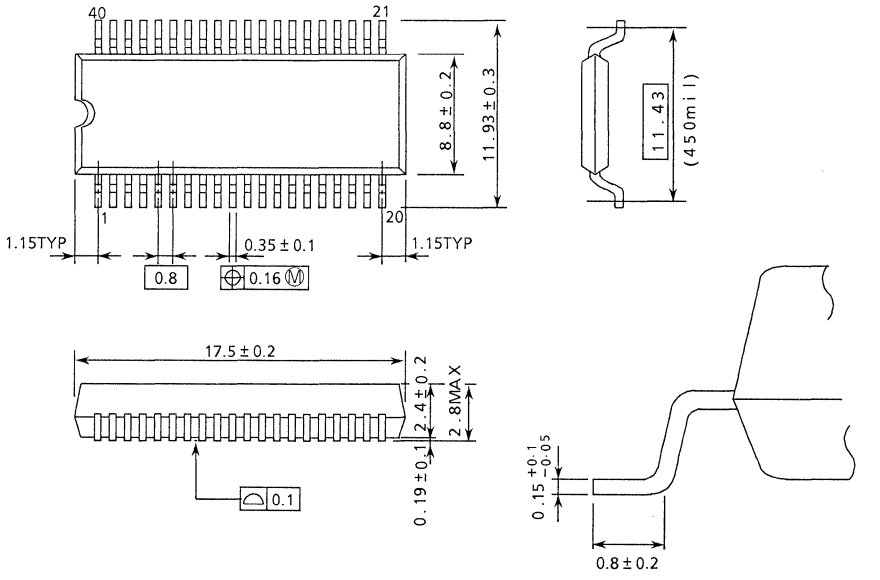
270289

Note : Each lead pitch is 2.54mm, and all the leads are located witchin ±0.25mm from their theoretical positions with respect to No.1 and No.4 leads.

8.2 40PIN SMALL OUTLINE PACKAGE

SSOP40-P-450

Unit : mm



270289

PROGRAMMABLE PERIPHERAL INTERFACE

TMP8255AP-5

1. GENERAL DESCRIPTION AND FEATURES

The TMP8255A (hereinafter referred to as PPI) is a high Speed programmable input/output interface with three 8-bit I/O ports. 24 I/O ports are divided into two groups (Port A and Port B) which are programmable independently by control words provided by MPU. The PPI has three operation modes (Mode 0, 1 and 2) and is capable of versatile interface between MPU and peripheral devices.

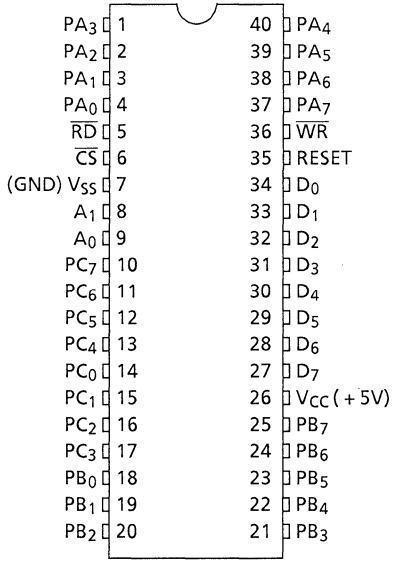
(1) 5V ± 5% Single power supply

(2) 24 programmable I/O ports

(3) Three operation modes (Mode 0, Mode 1, Mode 2)

(4) Bit set/reset capability

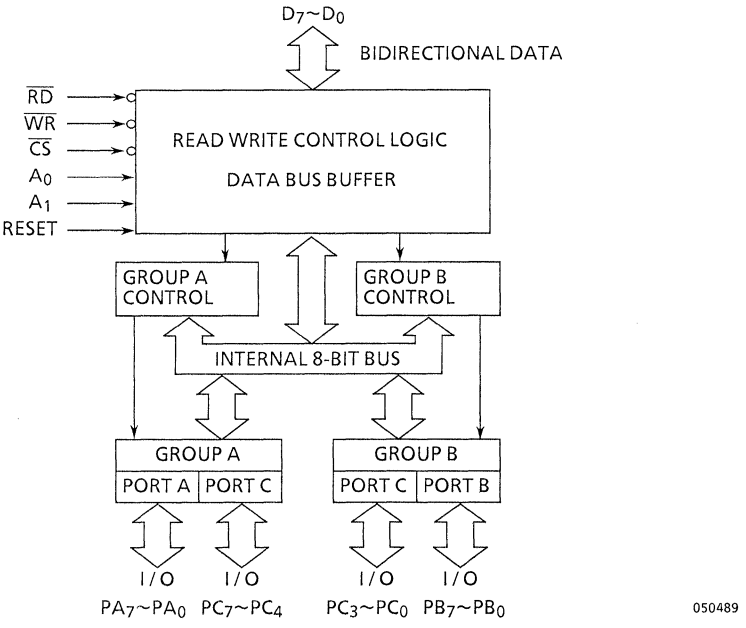
2. PIN CONNECTIONS (TOP VIEW)



TMP8255AP-5

050489

3. BLOCK DIAGRAM



4. PIN NAMES AND PIN FUNCTIONS

Pin Name	Number of Pin	Input/Output 3-state	Function
D_0~D_7	8	I/O 3-STATE	3-state bidirectional 8-bit data bus. Used for data transfer with MPU. Also, used for transfer of control words to PPI and status information from PPI.
PA_7~PA_0	8	I/O 3-STATE	3-state 8-bit I/O Port A. Operation mode and input/output configuration are defined by software. Port A contains the output latch buffer and input latch.
PB_7~PB_0	8	I/O 3-STATE	3-state 8-bit I/O Port B. Operation mode and input/output configuration are defined by software. Port B contains the output latch buffer and input latch.
PC_7~PC_0	8	I/O 3-STATE	3-state 8-bit I/O Port C. Operation mode and input/output configuration are defined by software. Port C can be divided into two 4-bit ports by the mode control and also, used as the control signal for Port A and Port B. In this case, 3 bits of PC_0 to PC_2 are used for Port B and 5 bits of PC_3 to PC_7 for Port A.
$\overline{CS}$	1	Input	Chip select input. When this terminal is at "L" level, data transfer PPI and MPU becomes possible. At "H" level, the data bus is placed in the high impedance state and control from the processor is ignored.
$\overline{RD}$	1	Input	Read signal. When this terminal is at "L" level, data that is input into the port is transferred to MPU.
$\overline{WR}$	1	Input	Write signal. When this terminal is at "L" level, data or control word is written into PPI from MPU.
A_0, A_1	2	Input	Used for selecting Port A, B, C and the control registers. Nurmally, this terminal is connected to low order 2 bits of the address bus.
RESET	1	Input	When this terminal is at "H" level, all internal registers including the control register are cleared. In addition, all ports (Port A, B, C) are placed in the input mode (high impedance) of mode 0.
V_{CC}	1	Power Supply	5V
V_{SS}	1	Power Supply	GND

050489

5. FUNCTIONAL DESCRIPTION

The PPI is a programmable peripheral interface with three 8-bit ports (Port A, B and C) and two control registers. 24 I/O ports are divided into 12-bit group A and group B. Group A consists of Port A and high order 4 bits of Port C, while Group B consists of Port B and low order 4 bits of Port C. Each group is independently programmable by control words provided from MPU. There are three operation modes available for the PPI. In mode 0, two 8-bit I/O ports and two 4-bit I/O ports can be programmed as input or output ports, respectively. In mode 1, 24 I/O ports are divided into Group A and Group B. 8 bits of each group are used as input or output port and of the remaining 4 bits, 3 bits are used as handshaking and interrupt control signal. Mode 2 is applicable only to group A and the ports are used as a bidirectional 8-bit data bus and 5-bit control signal. In case of Port C being used as the output, any bits of Port C can be set/reset.

There are two control registers; one is used for mode setting and the other for bit set/reset control. The control registers can only be written into. Further, when the reset input (RESET) becomes "1", the control registers are reset and all I/O ports are placed in input mode (high impedance status).

Table 5.1 Basic Operation of TMP8255A

A_1	A_0	$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	Function
0	0	0	0	1	Data bus ← Port A
0	1	0	0	1	Data bus ← Port B
1	0	0	0	1	Data bus ← Port C
0	0	0	1	0	Port A ← Data bus
0	1	0	1	0	Port B ← Data bus
1	0	0	1	0	Port C ← Data bus
1	1	0	1	0	Control register ← Data bus
×	×	1	×	×	Data bus = 3-state
×	×	0	1	1	Data bus = 3-state
1	1	0	0	1	inhibition of combination

050489

5.1 MODE SELECTION

There are three basic modes of operation that can be selected by control words.

Mode 0-Basic I/O (Group A, Group B)
Mode 1-Strobe input/Strobe output (Group A, Group B)
Mode 2-Two-way bus (Port A only)

Operation modes for Group A and Group B can be independently defined by the control word from the MPU. If D_7 is set to "1" in writing a control word into the PPI, on operation mode is selected, while of D_7 = "0", the set/reset function for Port C is selected.

5.1.1 Control word to define operation mode

Figure 5.1 shows the control words to define operation mode of the TMP8255A.

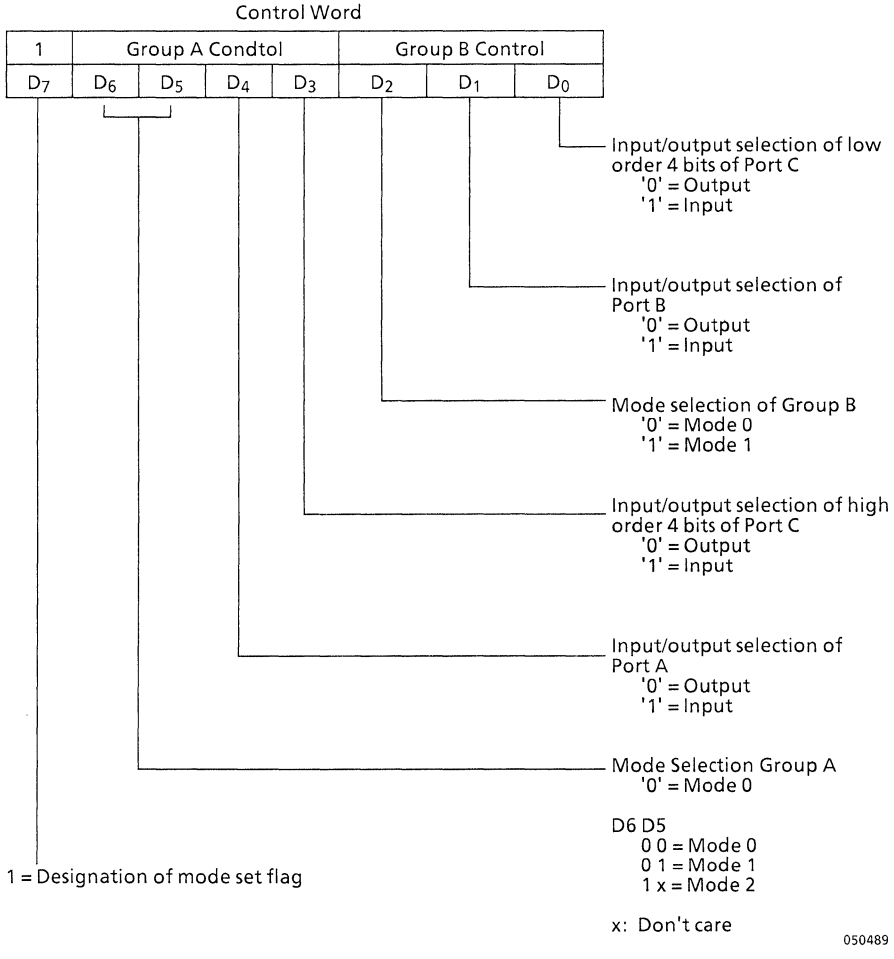


Figure 5.1 Control Word for Mode Selection

5.1.2 Port C bit set/reset control word

Any bit of 8 bits of Port C can be set/reset by Port C bit set/reset control word. Figure 5.2 shows the Port C bit set/reset control word.

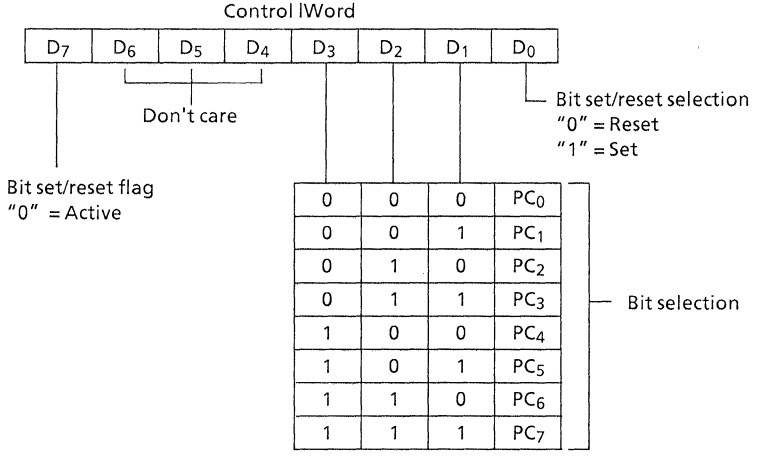


Figure 5.2 Control Word for Bit Set/Reset

5.2 OPERATION MODES

5.2.1 Mode 0 (Basic I/O)

This functional configuration is used for simple input or output operations. No 'handshaking' is required and data is simply written to or read from a specified part. Output data to the ports from MPU are latched out but input data from the ports are not latched.

In Mode 0, 24 I/O ports are divided into four groups of Port A (8 BITS), Port B (8 bits), high order 4 bits of Port C and low order 4 bits of Port C. Each port can be programmed to be input or output. The configuration of each port are determined according to the contents of Bit 4 (D_4), 3 (D_3), 1 (D_1) and 0 (D_0) of the control word for mode selection.

The I/O configuration of each port in Mode 0 are shown in Table 5.2.

Node Setting Control Word				Port A	Port C (PC_7~PC_4)	Port B	Port C (PC_3~PC_0)
D_4	D_3	D_1	D_0				
0	0	0	0	Out	Out	Out	Out
0	0	0	1	Out	Out	Out	In
0	0	1	0	Out	Out	In	Out
0	0	1	1	Out	Out	In	In
0	1	0	0	Out	In	Out	Out
0	1	0	1	Out	In	Out	In
0	1	1	0	Out	In	In	Out
0	1	1	1	Out	In	In	In
1	0	0	0	In	Out	Out	Out
1	0	0	1	In	Out	Out	In
1	0	1	0	In	Out	In	Out
1	0	1	1	In	Out	In	In
1	1	0	0	In	In	Out	Out
1	1	0	1	In	In	Out	In
1	1	1	0	In	In	In	Out
1	1	1	1	In	In	In	In

050489

Figure 5.3 Port definition in Mode 0

5.2.2 Mode 1 (Strobe I/O)

In Mode 1, input/output of port data is performed in conjunction with the strobe signals or 'handshaking' signals. Port C is used to control Port A or Port B.

The basic operatings in Mode 1 are as follows:

- Mode 1 can be set for two groups of Group A and Group B.
- Each group consist of 8-bit data port and 4-bit control/data port.
- The 8-bit data port can be set as input or output port.
- The control/data port is used as control or status of the 8-bit data port.

(1) When used as the input port in Mode 1:

- $\overline{STB}$ (Strobe Input)

  At "0", input data is loaded in the internal input latch in the port.
  In this case, a control signal from MPU is not concerned and data is input from the port any time. This data is not read out on the data bus unless MPU executes an input instruction.

- IBF (Input Buffer Full F/F Output)

  When data is loaded in the internal input latch from the port, this output is set to "1". IBF is set ("1") by $\overline{STB}$ input being reset and is reset ("0") by the rising edge of $\overline{RD}$ input.

- INTR (Interrupt Request Output)

  Used for the interrupt process of data loaded in the internal input latch. When $\overline{STB}$ input is at "0" if INTE (INTE flag) in the PPI is in the enabled state ("1") , IBF is set to "1". INTR is set to "1" immediately after the rising edge of this $\overline{STB}$ input and reset to "0" by the falling edge of $\overline{RD}$ input.

  The INTE flags of Group A and Group B are controlled as follows:

  INTEA-Control by bit set/reset of PC_4

  INTEB-Control by bit set/reset of PC_2

(2) When used as the output port in Mode 1:

- $\overline{OBF}$ (Output Buffer Full F/F Output)

  This is a flag which shows that MPU has written data into a specified port. $\overline{OBF}$ is set to becomes "0" at the rising edge of $\overline{WR}$ signal and is set to "1" at the falling edge of $\overline{ACK}$ (Acknowledge input) signal.

- $\overline{ACK}$ (Acknowledge Input)

  $\overline{ACK}$ signal is sent to the PP1 as a response from a peripheral device that received data from the port.

- INTR (Interrupt Request Output)

  When a peripheral device received data from MPU, INTR is set to "1" and the interrupt is requested to MPU. If $\overline{ACK}$ signal is received when INTE flag is in the enable state, $\overline{OBF}$ is set to "1" and INTR signal becomes "1" immediately after the rising edge of $\overline{ACK}$ signal. Further, INTR is reset at the falling edge of $\overline{WR}$ signal when data is written into the PPI by MPU.

  The INTE flags of Group A and Group B are controlled as follows:

  INTEA-Control by bit set/reset of PC_6

  INTEB-Control by bit set/reset of PC_2

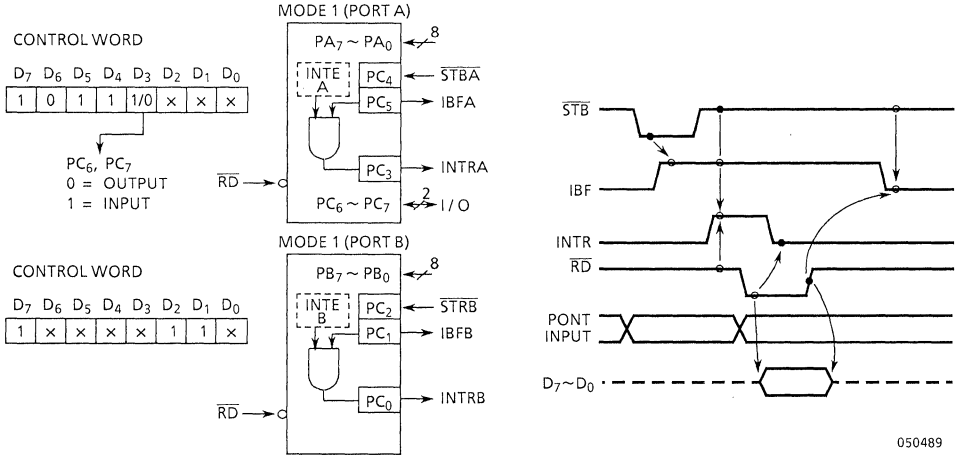


Figure 5.4 Example of Strobe Input in Mode 1

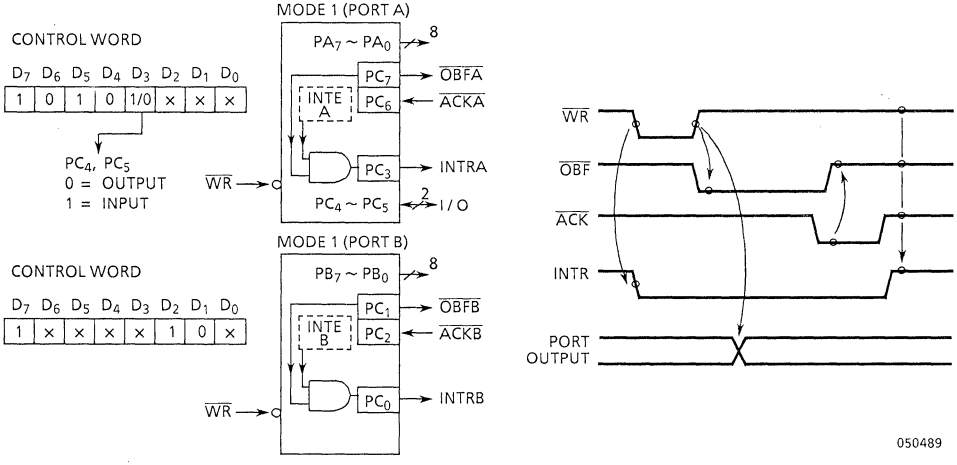


Figure 5.5 Example of Strobe Output in Mode 1

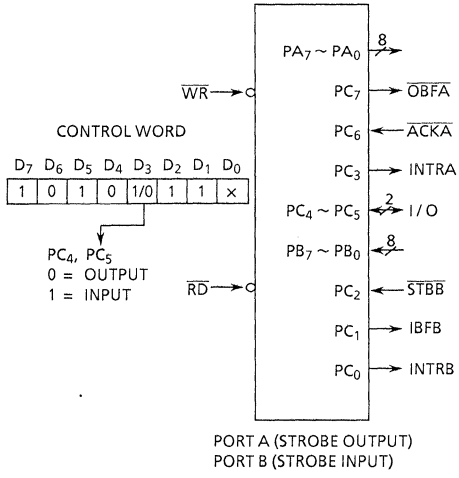
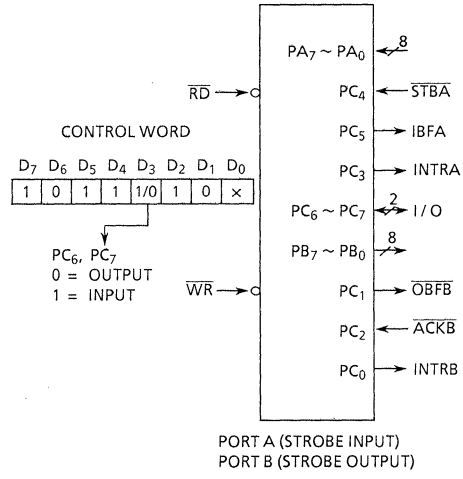


Figure 5.6 Example of Port A output, Port B Input in Mode 1



050489

Figure 5.7 Example of Port A Input, Port B Output in Mode 1

5.2.3 Mode 2 (Strobed Bidirectional Bus I/O)

In this mode, Port A is used as 8 bits bidirectional bus for data transfer with a peripheral device. This mode is applicable only to Group A, which consists of an 8-bit bidirectional bus (Port A 8-bit) and 5-bit control signals (high order 5 bits of Port C). The bidirectional bus (Port A) has both the internal input and output registers. When group A is set in Mode 2, Group B can be set independently. These are 5 control signals as follows when Group A is used in Mode 2.

- $\overline{OBF}$ (Output buffer Full F/F Output)

  When MPU writes data into of Port A, $\overline{OBF}$ is set to "0" to inform a peripheral device that the PPI is ready to output data. However, Port A is dept in the floating (high impedance) state until $\overline{ACK}$ input signal is received.

- $\overline{ACK}$ (Acknowledge Input)

  When $\overline{ACK}$ signal is set to "0", the data of the 3-state output buffer of Port A is send out. If $\overline{ACK}$ signal is at "1", Port A is in the high impedance state.

- $\overline{STB}$ (Strobe Input)

  When $\overline{STB}$ input is set to "0", the data from peripheral devices are held in the input latch. When the active $\overline{RD}$ signal is input into the PPI, the latched input data are output on the system data bus (D_7-D_0).

- IBF (Input Buffer Full F/F Output)

  When data from peripheral devices are held in the input latch, IBF is set to "1".

- INTR (Interrupt Request Output)

  INTR is the output to request the interrupt to MPU and its function is the same as that in Mode 1. There are two interrupt enable flip-flop (INTE), INTE1 corresponds to INTEA in Mode 1 output and INTE2 to INTEA in Mode 1 input.

  INTE 1-Used to generate INTR signal in conjunction with $\overline{OBF}$ and $\overline{ACK}$ signals, and is controlled by PC_6 bit set/reset.

  INTE 2-Used to generate INTR signal in conjunction with IBF and $\overline{STB}$ signals, and is controlled by PC_4 bit set/reset.

Figure 5.8 shows the operating example and the timing diagram in Mode 2.

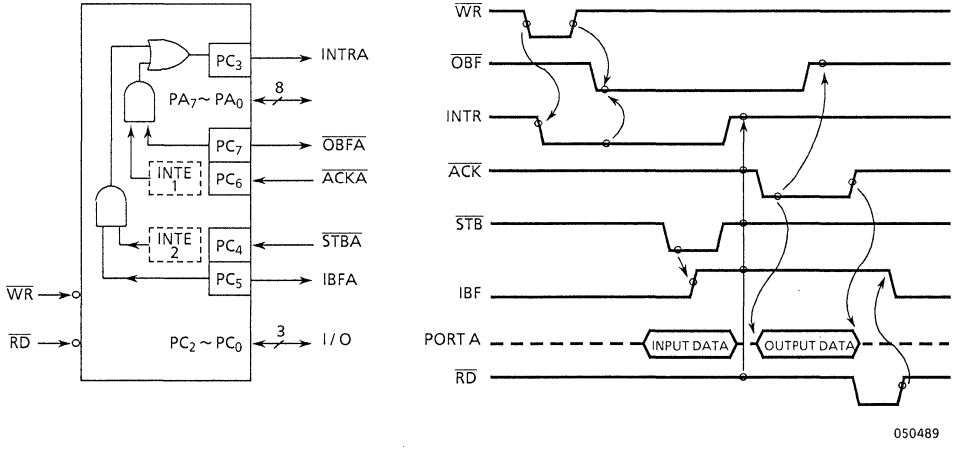


Figure 5.8 Operating example in Mode 2

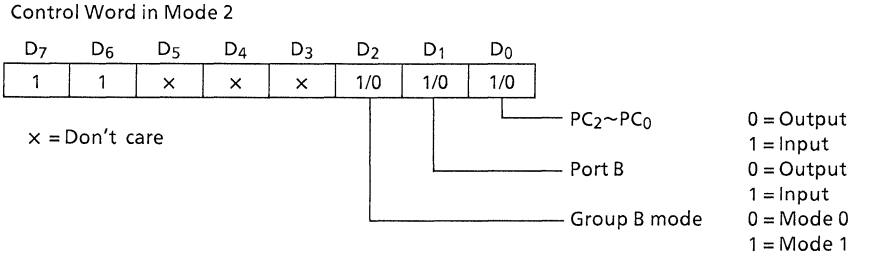


Figure 5.9 Control Word and Configuration in Mode 2

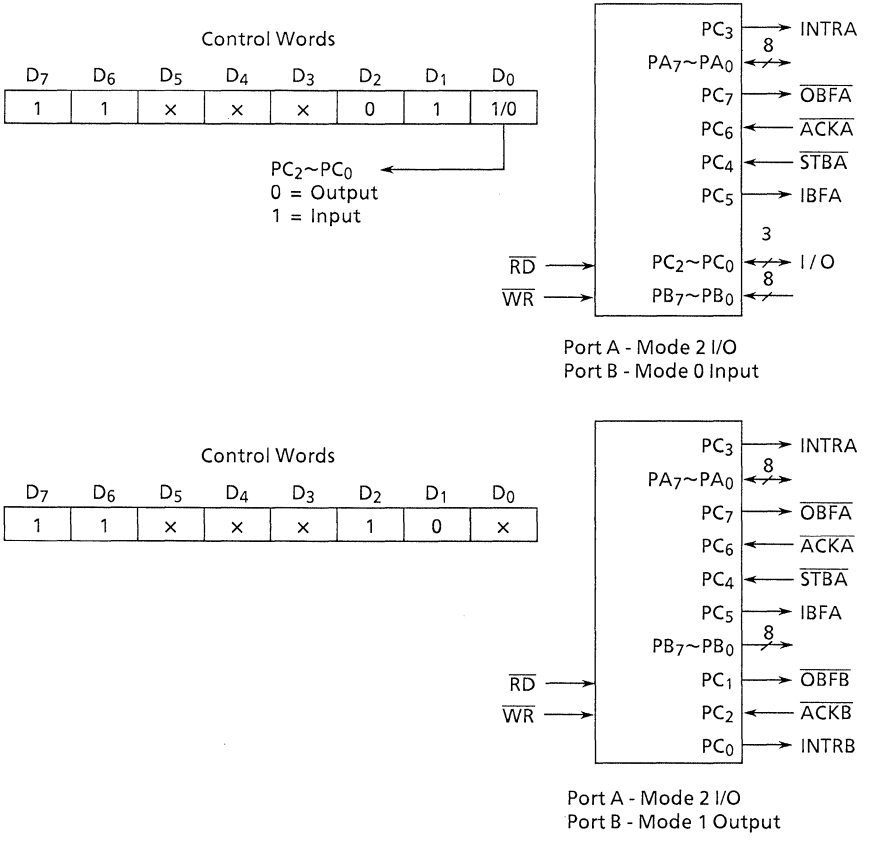


Figure 5.10 Example in Combination with Mode 2 and Other Mode

5.2.4 Pecautions for use in Mode 1 and 2

When used in Mode 1 and 2, bits which are not used as control or status in Port C can be used as follows.

If programmed as the input, they are accessed by normal Port C read.

If Programmed as the output, high order bits of Port C (PC_7-PC_4) are accessed using the bit set/reset function. As to low order bits of Port C (PC_3-PC_0), in additions ot access by the bit set/reset function, 3 bits only can be accessed by normal writing.

5.3 READING PORT C STATUS

When Port C is used as the control port, that is, when Port C is used in Mode 1 or Mode 2, the status information of the control word can be read out by a normal read operation of Port C.

Table 5.2 Status Word Format of Port C

Data Mode	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Mode 1 Input	I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB
Mode 1 Output	$\overline{OBFA}$	INTEA	I/O	I/O	INTRA	INTEB	$\overline{OBFB}$	INTRB
Mode 2	$\overline{OBFA}$	INTE1	IBFA	INTE2	INTRA	By Group B Mode		

050489

6. ELECTRICAL CHARACTERISTICS

6.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item	Rating	Unit
V_{CC}	Supply Voltage	− 0.5 to 7.0	V
V_{IN}	Input Voltage	− 0.5to V_{CC} + 7.0	V
P_D	Power Dissipation	1	W
TSOLDER	Soldering Temperature (10sec)	260	℃
T_{STG}	Strobe Temperature	− 65 to + 150	℃
T_{OPR}	Operating Temperature	0 to + 70	℃

050489

6.2 DC ELECTRICAL CHARACTERISTICS

TA = 0°C to 70°C, V_{CC} = 5V ± 5%, V_{SS} = 0V

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		− 0.5	—	0.8	V
V_{IH}	Input High Voltage		2.2	—	V_{CC}	V
V_{OL}	Output Low Voltage (DB) (PER)	I_{OL} = 2.5mA I_{OL} = 1.7mA	— —	— —	0.45 0.45	V V
V_{OH}	Output High Voltage (DB) (PER)	I_{OH} = − 400μA I_{OH} = − 200μA	2.4 2.4	— —	— —	V V
I_{IL}	Input Leak Current	$0 \leqq V_{IN} \leqq V_{CC}$	—	—	± 10	μA
I_{OFL}	Output Leak Current (High Impedance State)	$0 \leqq V_{OUT} \leqq V_{CC}$	—	—	± 10	μA
(Note 1) I_{DAR}	Darlington Drive Current	V_{EXT} = 1.5V R_{EXT} = 750Ω	− 1.0	—	− 4.0	mA
I_{CC}	Operating Supply Current	I/O cycle Time 1 usec	—	—	120	mA

050489

Note : Applied for optional 8 I/O terminals in Port B and Port C.

6.4 AC ELECTRICAL CHARACTREISTICS

TA = 0°C to 70°C, VCC = 5V ± 5%, VSS = 0V

SYMBOL	PARAMETER	TMP8255AP-5		UNIT
		MIN.	MAX.	
t_{AR}	Address set-up time for $\overline{RD}$ fall	0	—	ns
t_{RA}	Address hold time for $\overline{RD}$ rise	0	—	ns
t_{RR}	$\overline{RD}$ pulse width	300	—	ns
t_{RD}	Delay from $\overline{RD}$ fall to decided data output	—	200	ns
t_{DF}	Time from $\overline{RD}$ rise to data bus floating	10	100	ns
t_{RV}	Time from $\overline{RD}$ or $\overline{WR}$ rise to next $\overline{RD}$ or $\overline{WR}$ fall	850	—	ns
t_{AW}	Address set-up time for $\overline{WR}$ fall	0	—	ns
t_{WA}	Address holding time for $\overline{WR}$ rise	20	—	ns
t_{WW}	$\overline{WR}$ pulse width	300	—	ns
t_{DW}	Bus data set-up time for $\overline{WR}$ rise	100	—	ns
t_{WD}	Bus data holding time for $\overline{WR}$ rise	30	—	ns
t_{WB}	Delay from $\overline{WR}$ rise to decided data output	—	350	ns
t_{IR}	Port data set-up time for $\overline{RD}$ fall	0	—	ns
t_{HR}	Port data holding time for $\overline{RD}$ rise	0	—	ns
t_{AK}	$\overline{ACK}$ pulse width	300	—	ns
t_{ST}	$\overline{STB}$ pulse width	500	—	ns
t_{PS}	Port data set-up time for $\overline{STB}$ rise	0	—	ns
t_{PH}	Port data holding time for $\overline{STB}$ rise	180	—	ns
t_{AD}	Delay from $\overline{ACK}$ fall to decided data output	—	300	ns
t_{KD}	Time from $\overline{ACK}$ rise up to port (Port A in Mode2) floating	20	250	ns
t_{WOB}	Delay from $\overline{WR}$ rise to $\overline{OBF}$ fall	—	650	ns
t_{AOB}	Delay from $\overline{ACK}$ fall to $\overline{OBF}$ rise	—	350	ns
t_{SIB}	Delay from $\overline{STB}$ fall to $\overline{IBF}$ rise	—	300	ns
t_{RIB}	Delay from $\overline{RD}$ fall to $\overline{IBF}$ rise	—	300	ns
t_{RIT}	Delay from $\overline{RD}$ fall to INTR fall	—	400	ns
t_{SIT}	Delay from $\overline{ACK}$ rise to INTR rise	—	300	ns
t_{AIT}	Delay from $\overline{ACK}$ rise to INTR rise	—	350	ns
t_{WIT}	Delay from $\overline{WR}$ rise to INTR fall	—	450	ns

050489

Note : 1. When the power supply is turned ON, reset pulse duration must be active for at least 500 ns or more.

2. AC Measuring Point Input Voltage $V_{IH}=2.0V, V_{IL}=0.8V$
Output Voltage $V_{OH}=2.0V, V_{OL}=0.8V$
$CL=150pF$.

6.4 CAPACITANCE

$T_A = 25°C$, $V_{CC} = V_{SS} = 0V$

SYMBOL	ITEM	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
C_{IN}	Input Capacitance	f_C = 1MHz (*)	—	—	10	pF
$C_{I/O}$	I/O Capacitance		—	—	20	pF

050489

* : All terminals except that to be measured should be earthed.

7. TIMING DIAGRAM

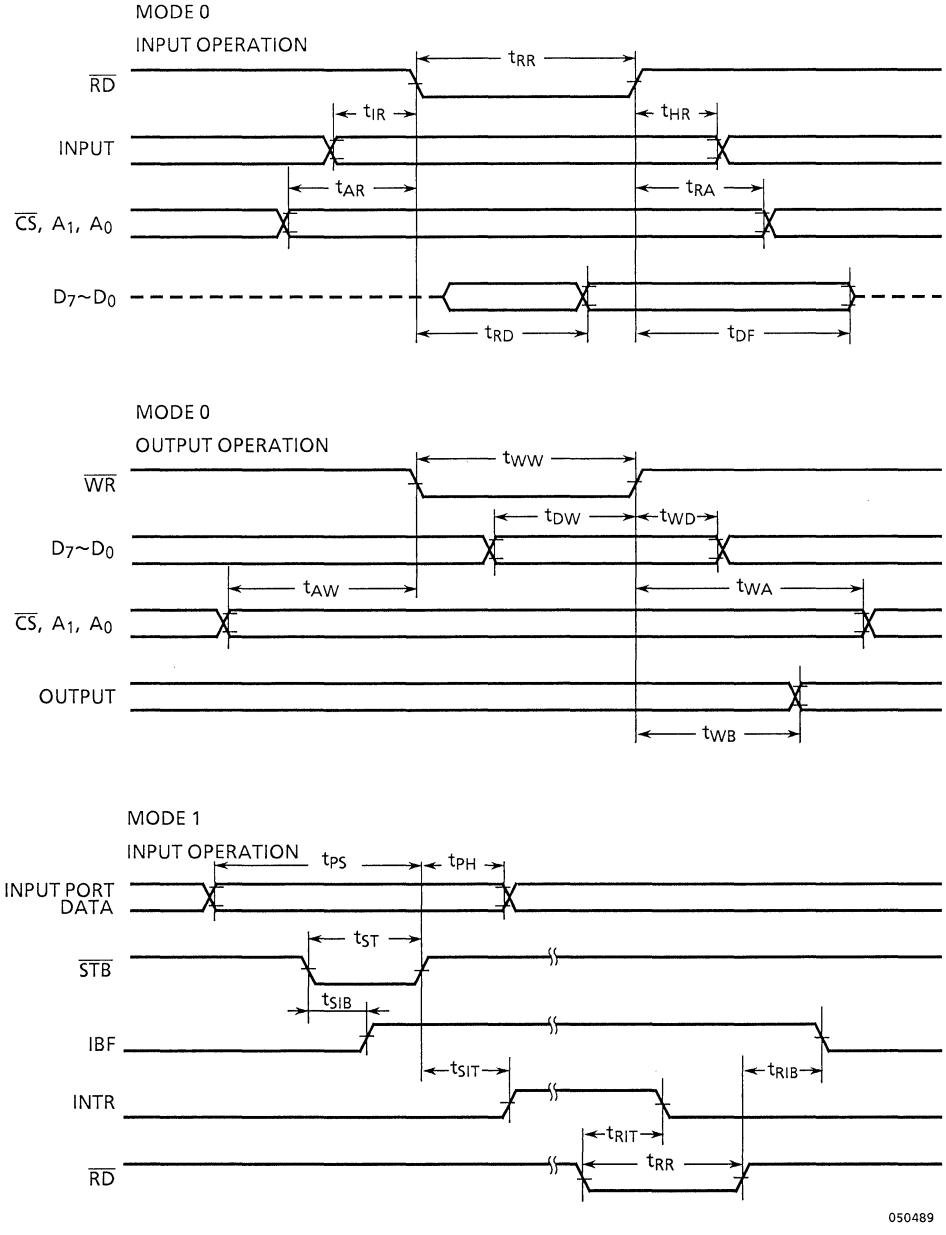


Figure 7.1 Timing diagram

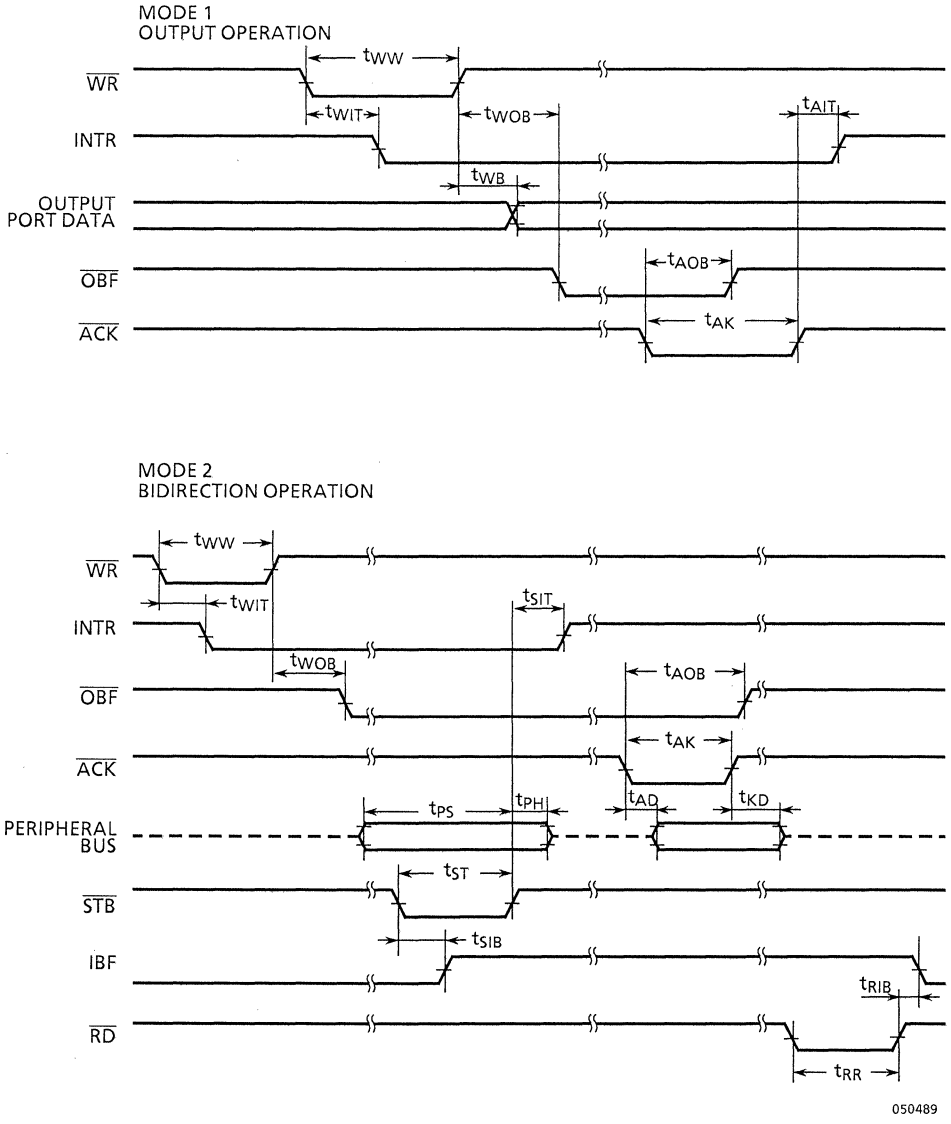


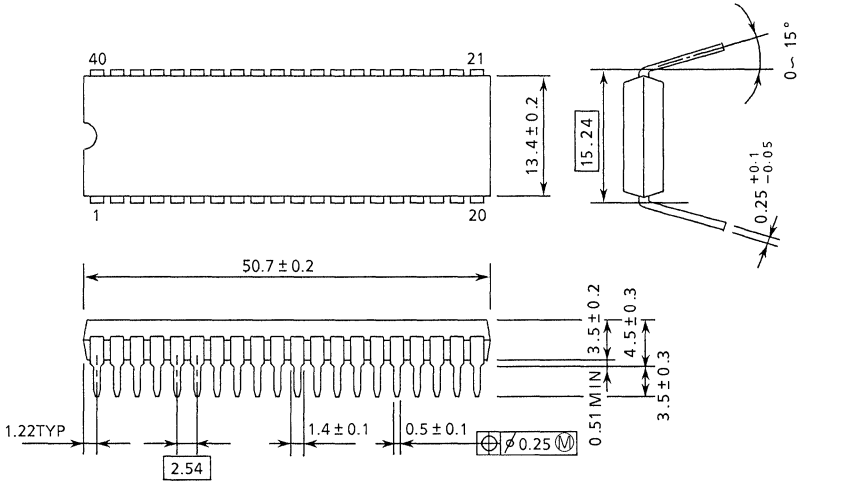
Figure 7.2 Timing diagram

8. PACKAGE DIMENSION

8.1 PLASTIC PACKGE

DIP40-P-600

Unit : mm



270289

Note: Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.40 leads.

TOSHIBA MOS TYPE DEGITAL
INTEGRATED CIRCUIT
Silicon Monolithic CMOS Silicon Gate

TMP82C59AP-2/TMP82C59AM-2

PROGRAMMABLE INTERRUPT CONTROLLER

1. GENERAL DESCRIPTION

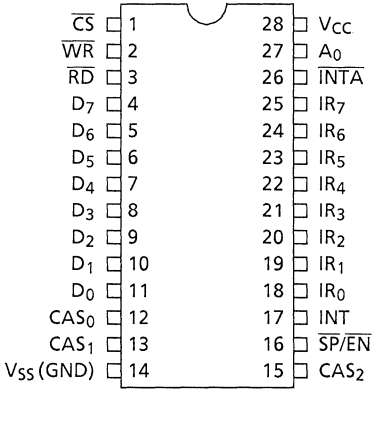
TMP82C59AP-2/AM-2 (hereinafter referred to as TMP82C59A) is a programmable interrupt controller. It handles up to eight vectored priority interrupts for the MPU. It is cascadable for up to 64 vectored priority interrupts without additional circuitry.

FEATURES

- Eight Level Priority Controller.
- Expandable to 64 Level.
- Interrupt Modes, Interrupt Mask, Vectored Address Programmable.
- Single +5V Power Supply.
- Supports 8085A, 8086 Microcomputer Interrupt Sequence.
- TTL Compatible.

2. PIN CONNECTIONS (TOP VIEW)

TMP82C59AP-2/TMP82C59AM-2



050489

3. PIN NAMES AND DESCRIPTION

Table 3.1

Pin Name	Input / Output	Function
$\overline{CS}$	Input	Chip Select Input. A low on this pin enables $\overline{RD}$ and $\overline{WR}$ communication between the MPU and the TMP82C59A. $\overline{INTA}$ functions are independent of $\overline{CS}$.
$\overline{WR}$	Input	Write Control Input. A low on this pin when $\overline{CS}$ is low enables the TMP82C59A to receive command words from MPU.
$\overline{RD}$	Input	Read Control Input. A low on this pin when $\overline{CS}$ is low enables the TMP82C59A to output status onto the data bus for the MPU.
D_0 to D_7	Input / Output	Bidirectional Data Bus. Command status and interrupt-vector information is transferred via this bus.
CAS_0 to CAS_2	Input / Output	Cascade Lines. The CAS lines from a private TMP82C59A bus to control a multiple TMP82C59A structure. These pins are outputs for a master TMP82C59A and inputs for a slave TMP82C59A.
$\overline{SP}$ / $\overline{EN}$	Input / Output	Slave Program / Enable buffer. This is a dual function pin. In the buffered mode, it can be used as an output to control buffer transceivers ($\overline{EN}$). In the non-buffered mode, it is used as an input to designate a master TMP82C59A ($\overline{SP}$ = 1) or a slave one ($\overline{SP}$ = 0).
INT	Output	Interrupt Request Output. This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the MPU. It is connected to MPU's interrupt pin.
IR_0 to IR_7	Input	Interrupt Request Inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on a IR input (Level Triggered Mode.)
$\overline{INTA}$	Input	Interrupt Acknowledge Input. This pin is used to output interrupt-vector data onto the data bus by a sequence of interrupt acknowledge pulses issued by the MPU.
A_0	Input	Address Line. This pin acts in conjunction with the $\overline{CS}$, $\overline{WR}$ and $\overline{RD}$ pins. It is used by the TMP82C59A to decipher various command words the MPU writes and status the MPU wishes to read. It is typically connected to the MPU A_0 address line.
VCC		+ 5V Power Supply
VSS		Ground

050489

4. FUNCTIONAL DESCRIPTION

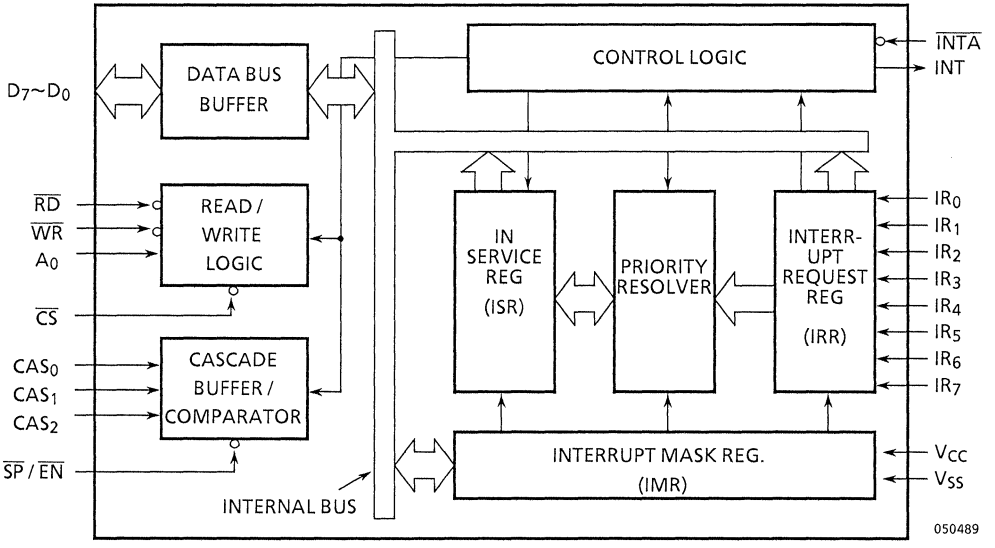


Figure 4.1 BLOCK DIAGRAM

The TMP82C59A is connected to the system bus as shown in Figure 4.2 and operates as an interrupt controller.

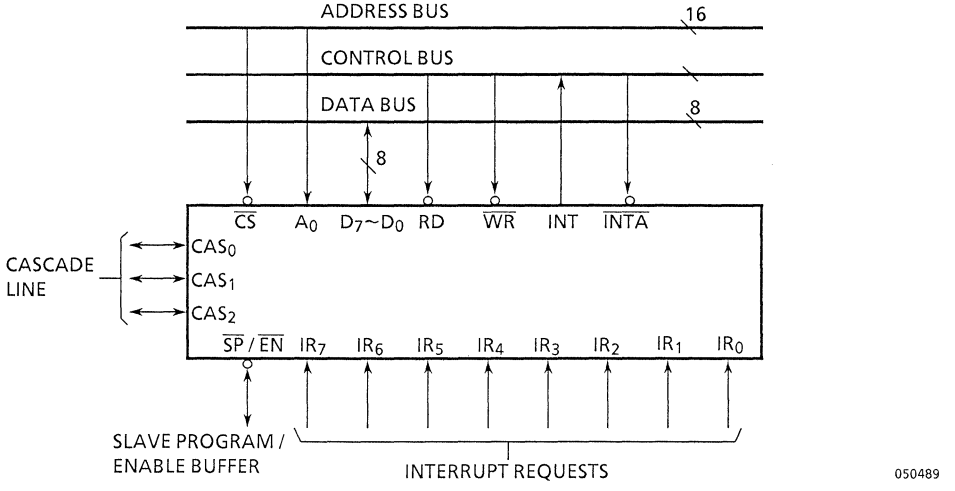


Figure 4.2 Interface to the System Bus

4.1 GENERAL DESCRIPTION

Whenever an interrupt request is received via IRn, the TMP82C59A, judging its mask status and priority, set INT high for requesting interrupt to MPU. Then, according to response signal ($\overline{INTA}$ signal) from MPU or the system controller, the TMP82C59A outputs CALL op-code and vectored address data on to the data bus. MPU starts the interrupt service routine and the TMP82C59A stores which interrupt request has been serviced. At the end of the service routine, MPU resets it and informs the TMP82C59A of its end.

Table 4.1 Basic Operation

A_0	D_4	D_3	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	READ OPERATION
0			0	1	0	IRR, ISR or Interrupt request level → Data bus
1			0	1	0	IMR → Data bus
A_0	D_4	D_3	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	WRITE OPERATION
0	0	0	1	0	0	Data bus → OCW2
0	0	1	1	0	0	Data bus → OCW3
0	1	×	1	0	0	Data bus → ICW1
1	×	×	1	0	0	Data bus → OCW1, ICW2, ICW3 or ICW4
A_0	D_4	D_3	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	HIGH IMPEDANCE
×	×	×	1	1	0	Data bus (D_7 to D_0) High impedance
×	×	×	×	×	1	Data bus (D_7 to D_0) High impedance

050489

4.2 SYSTEM CONFIGURATION

The TMP82C59A consists of the following components.

(1) Interrupt Request Register (IRR) and Inservice Register (ISR)

(2) Priority Resolver

(3) Interrupt Mask Register (IMR)

(4) Data Bus Buffer

(5) Read/Write Logic

(6) Cascade Buffer/Comparator

(7) Interrupt Request Register (IRR) and Inservice Register (ISR)

Interrupt requests from IRn inputs are processed by 2 registers, IRR and ISR. IRR holds interrupt request from the respective IRn inputs while ISR holds all interrupt levels that are being serviced by the MPU. Contents of IRR and ISR can be read out by the MPU.

(2) Priority Resolver

The priority resolver is the block that decides the interrupt request to be sent to MPU by judging priority. If the interrupt mask register (IMR) bit corresponding to IRn input is not set to 1, IRR sends the interrupt requests to the priority resolver.

Normally, when the interrupt request level having the highest priority among these interrupt request is higher than the content of ISR, that is, the priority of the interrupt request being serviced by the MPU, the TMP82C59A sends INT signal to the MPU. When $\overline{INTA}$ signal is input as a response from the MPU the TMP82C59A sends CALL op-code and the vectored address corresponding to an interrupt request of the highest priority to MPU, and resetting IRR bit corresponding to this interrupt request sets ISR bits. The MPU processes the interrupt service, sends a command to the TMP82C59A to accept interrupt requests of lower priority at the end of the interrupt service, and resets the corresponding ISR bits. The priority resolver has a register to assign the interrupt request input of lowest priority.

(3) Interrupt Mask Register (IMR)

The interrupt mask register normaly acts only on IRR, and disables interrupt requests from the masked IRn input. The mask for an interrupt request input does not affect its lower priority interrupt requests. In the special mask mode, this register also acts on ISR and enables acceptance of lower level interrupt requests than the interrupt request being serviced. The content of IMR can be read out.

(4) Data Bus Buffer

The data bus buffer consists of 8 bit 3 state bidirectional bus buffer interfacing with the system bus. Command words, status information CALL op-code and vectored addresses are transfered via this bus buffer.

(5) Read/Write Logic

This circuit controls the functions for decoding and accepting command words from MPU and for feeding status information to the data bus.
In addition, this circuit controls operations including ICW (Initialization Command Word) register and OCW (Operation Set Command Word) register.

$\overline{CS}$: Low level input to $\overline{CS}$ enables $\overline{RD}$ or $\overline{WR}$ input operation.

$\overline{WR}$: When $\overline{WR} = \overline{CS} = 0$, a command write to the TMP82C59A is enabled.

$\overline{RD}$: When $\overline{RD} = \overline{CS} = 0$, the contents of ISR, IRR and IMR and interrupt level in the poll mode can be read.

A_0 : A_0 is used together with $\overline{WR}$ and $\overline{RD}$ signals for command write or status readout. It acts as a select signal for the one of command words or status information. It is normally connected to the one of address lines.

(6) Cascade Buffer/Comparator

When programmed as a slave device, this block stores the identification code as the slave and compares this identification code with the data on the 3 bit cascade lines (CAS_0-2). When both agree, the slave interpretes that the slave itself is selected. In the case of the master, an identification signal corresponding to the accepted interrupt request inputs of the slave device are output for a period from the first $\overline{INTA}$ signal to the last $\overline{INTA}$ signal (second or third signal).

4.3 INTERRUPT SEQUENCE

(1) When the 8085A is used as the MPU

(a) When one or more interrupt request become high level, IRR bits corresponding to that input are set.

(b) The TMP82C59A judges the mask status and priority of these interrupt and outputs INT signal to MPU as necessary.

(c) MPU outputs $\overline{INTA}$ signal in response to INT signal.

(d) Upon receipt of $\overline{INTA}$ signal, the TMP82C59A outputs CALL op-code on the data bus.

(e) Since 'CALL' is a 3-byte instruction, additional two $\overline{INTA}$ signals are consecutively sent from MPU.

(f) Upon receipt of these two $\overline{INTA}$ signals, the TMP82C59A outputs the programmed vector address corresponding to the highest priority interrupt request. The TMP82C59A outputs the low-order address and then, the high-order address. Furthermore, the TMP82C59A sets the ISR bit corresponding to the interrupt request and resets IRR bit.

(g) The above operations complete CALL instruction and MPU executes the interrupt service. In AEOI mode, ISR bits are automatically reset immediately after the above operations. Otherwise, ISR bits are kept in the set status till EOI command is input.

(2) When the 8086 is used as MPU

(a) to (c) Same as (a) to (c) for the 8085A.

(d) Even when $\overline{INTA}$ signal is received, the TMP82C59A keeps the data bus in high impedance state.

(e) Another $\overline{INTA}$ signal is sent from MPU. The TMP82C59A outputs 8 bit pointer on the data bus, and sets the corresponding ISR bit and resets the IRR bit.

(f) The above operations complete the interrupt acknowledge cycle. In AEOI mode, the ISR bit is automatically reset after the final $\overline{INTA}$ signal is received. Otherwise, ISR bits are kept in the set status till EOI command is input.

Further, if there is no interrupt request at the time of step (d) of the above interrupt sequence, (i.e., the request was too short in duration), the TMP82C59A performs the same operations as those when interrupt request are generated at IR_7, but ISR bits are not set.

4.4 INTERRUPT SEQUENCE OUTPUT

(1) When the 8085A is used as the MPU

CALL op-code is output on the data bus upon receipt of the first $\overline{INTA}$ signal and the low-order vectored address and the high-order vectored address on the data bus upon receipt of the second and third $\overline{INTA}$ signals, respectively.

The vectored address A_5 to A_{15} on Table 4.3, 4.4 must be programmed in advance on the TMP82C59A. The remaining bits of the vectored addresses are produced by the TMP82C59A corresponding to interrupt request.

(2) When the 8086 is used as the MPU

When the first $\overline{INTA}$ signal is received, the data bus is placed in the high impedance state. When the second $\overline{INTA}$ signals is received, 8 bit pointer is output on the data bus. The 8 bit pointers T_7 to T_3 shown in Table 4.5 must be programmed in advance on the TMP82C59A. The remaining bits are automatically produced by the TMP82C59A corresponding to interrupts.

(1) 8085A MODE

Table 4.2 For First $\overline{INTA}$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	0	0	1	1	0	1

050489

Table 4.3 For Second $\overline{INTA}$

IR	INTERVAL = 4							
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
IR_7	A_7	A_6	A_5	1	1	1	0	0
IR_6	A_7	A_6	A_5	1	1	0	0	0
IR_5	A_7	A_6	A_5	1	0	1	0	0
IR_4	A_7	A_6	A_5	1	0	0	0	0
IR_3	A_7	A_6	A_5	0	1	1	0	0
IR_2	A_7	A_6	A_5	0	1	0	0	0
IR_1	A_7	A_6	A_5	0	0	1	0	0
IR_0	A_7	A_6	A_5	0	0	0	0	0

050489

IR	INTERVAL = 8							
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
IR_7	A_7	A_6	1	1	1	0	0	0
IR_6	A_7	A_6	1	1	0	0	0	0
IR_5	A_7	A_6	1	0	1	0	0	0
IR_4	A_7	A_6	1	0	0	0	0	0
IR_3	A_7	A_6	0	1	1	0	0	0
IR_2	A_7	A_6	0	1	0	0	0	0
IR_1	A_7	A_6	0	0	1	0	0	0
IR_0	A_7	A_6	0	0	0	0	0	0

050489

Table 4.2 For Third $\overline{INTA}$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8

050489

(2) 8086 MODE

Table 4.5 Second $\overline{INTA}$

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
IR_7	T_7	T_6	T_5	T_4	T_3	1	1	1
IR_6	T_7	T_6	T_5	T_4	T_3	1	1	0
IR_5	T_7	T_6	T_5	T_4	T_3	1	0	1
IR_4	T_7	T_6	T_5	T_4	T_3	1	0	0
IR_3	T_7	T_6	T_5	T_4	T_3	0	1	1
IR_2	T_7	T_6	T_5	T_4	T_3	0	1	0
IR_1	T_7	T_6	T_5	T_4	T_3	0	0	1
IR_0	T_7	T_6	T_5	T_4	T_3	0	0	0

050489

4.5 PROGRAMMING TMP82C59A

The TMP82C59A accepts the following 2 types of command words.

(1) Initialization Command Words (ICW)

Prior to operating the TMP82C59A, it is necessary to program this command.

(2) Operation Command Words (OCW)

This command is for operating the TMP82C59A in various operating modes and is programmable anytime during the TMP82C59A is in operation.

(1) ICW

There are 4 kinds of commands; ICW1, ICW2, ICW3 and ICW4.
Each of these command is not programmable independently. The initialization is made according to the initialization command sequence shown in Figure 4.3. ICW3 is used for cascade connection and ICW4 is for setting optional modes.

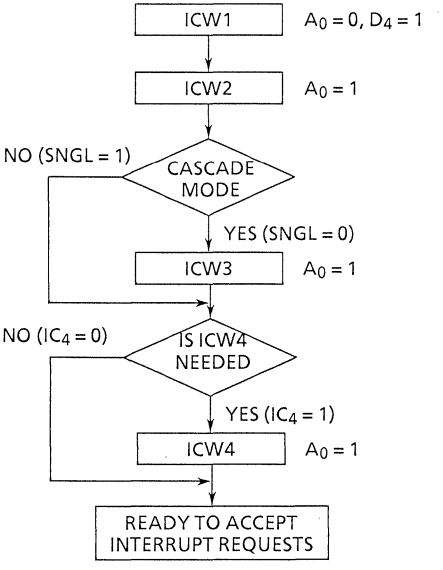


Figure 4.3 Initialization Command Sequence

ICW1

When $A_0=0$ and $D_4=1$, the initialization is interpreted as ICW1 and the initialization of following 5 items are made independently of content of the command:

[1] The interrupt mask register (IMR) is cleared.

[2] The interrupt request input IR_7 becomes the lowest priority.

[3] The special mask mode is cleared and IRR is assigned as the register for reading status information.

[4] When $IC_4=0$, all function bits of ICW4 are set at "0".

[5] The edge detection circuit of the interrupt request terminal is cleared.

ICW1 makes the assignment of vector addresses A_7 to A_5, assignment as to whether the interrupt request input is to be made in the edge trigger mode or the level trigger mode (LTIM), assignment of CALL address intervals when the 8085A is used as MPU (refer to Table 4.3) (ADI), assignment as to whether the cascade connection to be made (SNGL) and assignment as to whether ICW4 is needed (IC4).

ICW2

ICW2 assigns high-order vector addresses A_8 to A_{15} when the 8085A is used as MPU or 8-bit pointers T_3 to T_7 when the 8086 is used as MPU. TMP82C59A interpretes a command written with A_0 input made to "H" level after ICW1 written as ICW2.

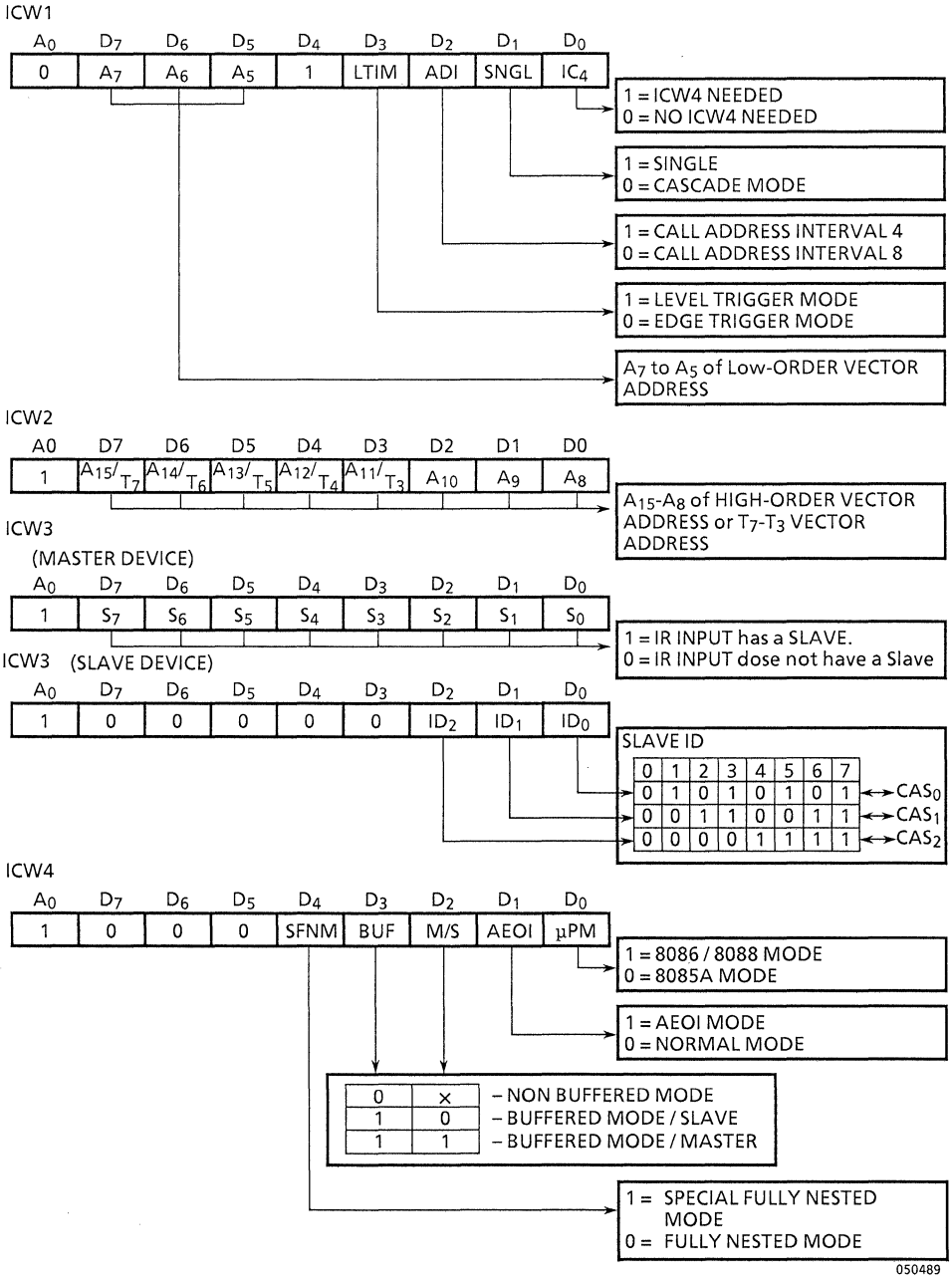


Figure 4.4 ICW Format

ICW3

This is a command required for cascade connection of plural number of the TMP82C59As. When SNGL = 0 in ICW1, the TMP82C59A interpretes a command written with A_0 input made at "H" level after ICW2 as ICW3.

[1] Master Mode

In the master mode, the TMP82C59A specifies individually as to whether a slave device is added to each interrupt request input.
If the 8085A is used as MPU when addition of a slave device is specified, the master device outputs CALL op-code on the data bus upon receipt of the first $\overline{INTA}$ signal and simultaneously outputs the slave identification code to the cascade line.

The master device becomes high impedance at the second and third $\overline{INTA}$ signals, and the slave devices selected by the identification code outputs vector address on the data bus. When the 8086 is used as MPU, both the master and slave devices become high impedance at the first $\overline{INTA}$ signal. Simultaneously, the master device outputs the slave identification code to the cascade line. The master device also become high impendance at the second $\overline{INTA}$ signal and the selected slave device outputs a pointer on the data bus. When it is specified that no slave device is added, the master device outputs both CALL op-code and vector address as a response to $\overline{INTA}$ signal and simultaneously outputs "L" signal to 3 cascade lines. This is the same as the identification code of the slave device connected to IR_0 and therefore, in the case of the interrupt request input without the slave device, added, no slave device can be added to IR_0.

Further, to specify the master or slave, the $\overline{SP}/\overline{EN}$ terminal must be set at "H" level or BUF must be set at 1 and M/S at 1 by ICW4.

[2] Slave Mode

In the slave mode, the TMP82C59A specifies the slave identification code.
The slave device compares its identification code with the identification code sent from the master device via the cascade line and if they agree, outputs vector addresses on the data bus upon receipt of the second and third $\overline{INTA}$ signals.
Further, to specify the slave mode, the $\overline{SP}/\overline{EN}$ terminal must be set at "L" level or BUF at 1 and M/S at 0 by ICW4.

ICW4

ICW4 is effective only when $IC_4 = 1$ in ICW1.
Although ICW4 is effective for assignment of the special fully nested mode (SFNM), assignment of the buffer mode (BUF) and in the buffer mode, this command makes the assignment of the master/slave (M/S), automatic EOI (AEOI) and MPU mode. When $IC_4 = 0$ in ICW1, all function bits of ICW4 are set at "0".

(2) OCW

There are 3 kinds of commands: OCW1, OCW2, and OCW3. Any time after ICW is programmed, these command can be programmed to set the TMP82C59A in various operation modes.

OCW1

After ICW is set, the TMP82C59A interpretes the operation set command to be OCW1when $A_0 = 1$. This command is used for setting the content of the interrupt mask register (IMR). The OCW1 format is shown in Figure 4.5.

OCW2

The TMP82C59A interpretes the operation set command to be OCW2 when $A_0 = 0$, $D_4 = 0$ and $D_3 = 0$. This command is used for outputting EOI. L_2 to L_0 are effective only in the case of specific EOI and specific rotation.
The OCW2 format is shown in Figure 4.5.

OCW3

The TMP82C59A interpretes the operation set command to be OCW3 when $A_0 = 0$, $D_4 = 0$ and $D_3 = 1$. This command is used for assigning the special mask mode, the poll mode and register for status information readout, that is, assigning IRR or ISR. The OCW3 format is shown in Figure 4.5.

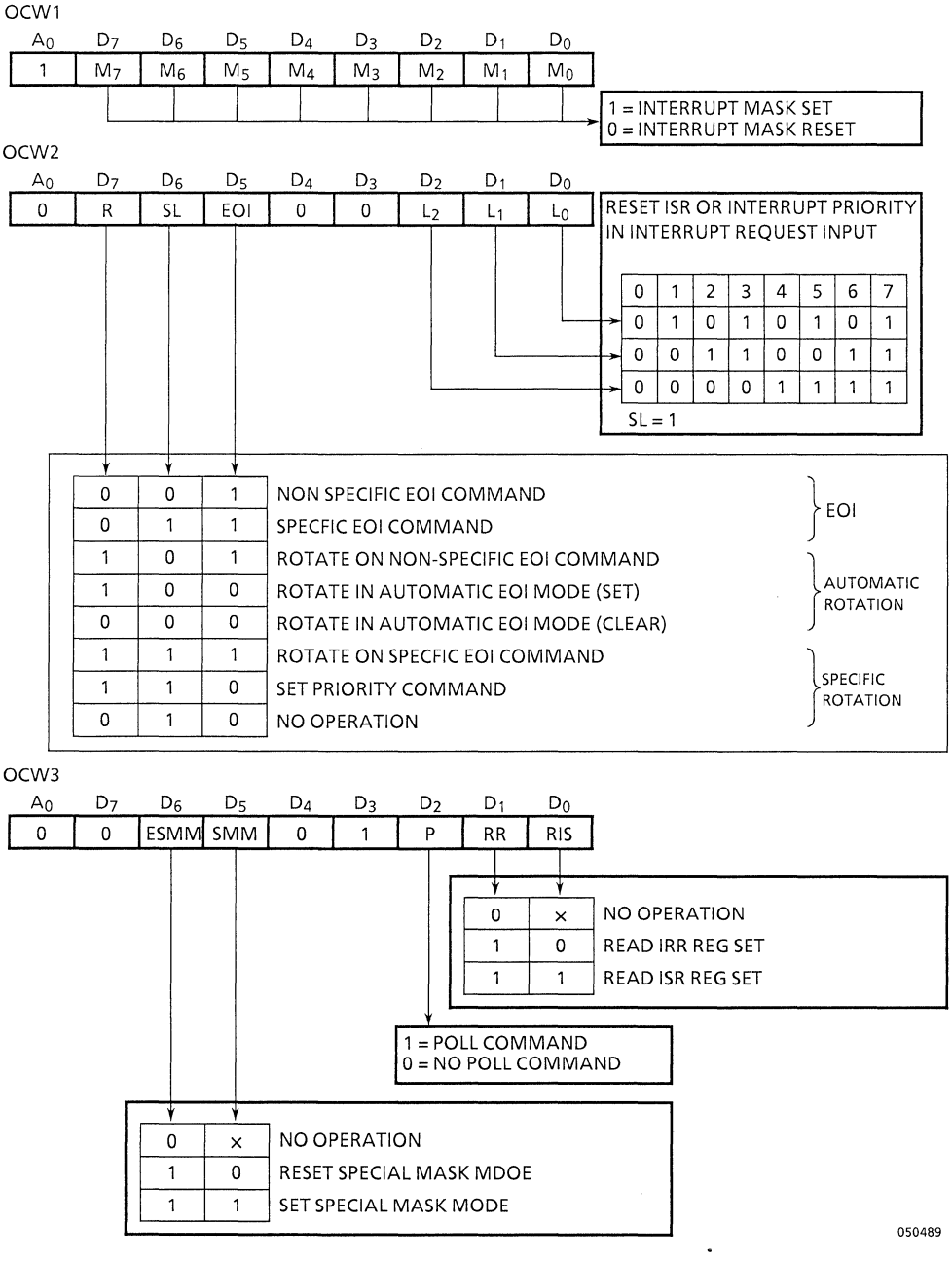


Figure 4.5 OCW Format

4.6 EXPLANATION OF MODES AND COMMANDS

(1) FULLY NESTED MODE

Unless the other modes are specified, the TMP82C59A operates in this mode. Under this mode, IR_0 is the highest priority level and IR7 bocomes the lowest.

When $\overline{INTA}$ signal is input, vector address corresponding to an interrupt and request having the highest priority at the time is output together with CALL op-code on the data bus and furthermore, corresponding ISR bits are kept set till EOI command is input to the TMP82C59A before MPU returns from the service routine or to the final leading edge of $\overline{INTA}$ pulse in AEOI mode. As long as these ISR bits are kept set, lower priority interrupt requests are ignored. Priority can be changed by OCW2.

(2) EOI (END OF INTERRUPT)

EOI command is used to reset ISR bits. It is necessary for MPU to output EOI command before returning from the service routine.

When AEOI is set in ICW4, ISR bit are automatically reset at the leading edge of the final $\overline{INTA}$ pulse and it is therfore not necessary to output EOI command. As ISR bits are set in both the master and slave devices when cascade connected, it is necessary to output EOI command to both master device and the slave device corresponding to the master device.

EOI command is available in 2 kinds: non-specific EOI and specific EOI commands. When non-specific EOI command is output to the TMP82C59A, ISR bit having the highest priority among ISR bit is reset. However, in the special mask mode it is not possible to reset ISR bit that are masked by IMR by the non-specific EOI command, and ISR bit having the highest priority among the unmasked ISR bits is reset. On the other hand, it is possible to specify ISR bit to be reset by the specific EOI command by a program. EOI command is executed by OCW2.

(3) AEOI (AUTOMATIC EOI) MODE

In this mode, the non-specific EOI operation is automatically executed at the leading edge of the final $\overline{INTA}$ signal.

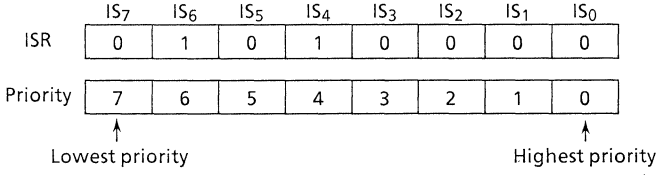
Therefore, this mode cannot be used for nested interruptions. In addition, this mode also cannot be used in the slave TMP82C59A. The TMP82C59A can be set in AEOI mode by setting AEOI bit in ICW4 to 1.

(4) AUTOMATIC ROTATION

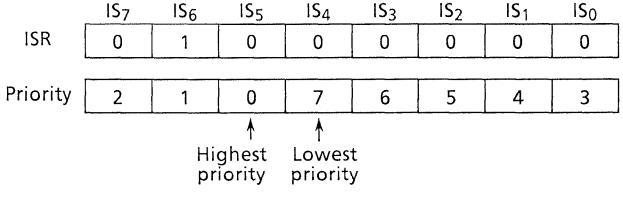
This mode is effective in the application to give equal priority to the interrupt devices. In this mode, whenever the interrupt service ends, priority of each interrupt request is updated so that the serviced interrupt request is set at the lowest priority. Priority of interrupt request input IRn (n=0 to 7) that has been serviced becomes the lowest priority level 7 and becomes high in order toward IR_0 and then, IR_7 and next IR_{n+1} become the highest priority level 0. (Rotation Priority)

For instance, when the interrupt request IR_4 is seviced as shown in Figure 4.6, the priority of each interrupt request input is updated.
This mode specifies R=1, SL=0 and EOI=1 by OCW2 at the end of service. Further, in case of AEOI mode, when R=1, SL=0 and EOI=0 are specified by OCW2, the internal flip-flop is set and the TMP82C59A operates in this mode. If R=0, SL=0 and EOI=0 are specified by OCW2, this mode is cleared.

before Rotation (highest priority interrupt request IR_4 is being serviced.)



After ROTATION (Interrupt request IR_4 is being serviced.)



050489

Figure 4.6 The Example of Interrupt Priority Transition in Automatic Rotation Mode

(5) SPECIFIC ROTATION

In the automatic rotation mode, priority of each interrupt request input is updated whenever interrupt requests are serviced. Under this mode it is possible to change priority by specifying an interrupt request input to be set at the lowest priority by a program. Priority is determined according to the ratation priority. In this mode, R and SL are set at 1 by OCW2 and interrupt request input that is to be lowest priority at L_2 to L_0 is specified. Priority can be changed simultaneously with EOI command or independently regardless of EOI command.

(6) INTERRUPT MASK

Each interrupt request input can be masked individually by the interrupt mask register (IMR). Content of IMR can be specified by OCWI.

(7) SPECIAL MASK MODE

Normally when an interrupt service routine is being executed, lower priority interrupt requests than the interrupt request being serviced are ignored unless ISR bits are reset by EOI command. This special mode is used for an application in which an interrupt request of lower priority is approved during the service. In this mode, IMR also acts as the mask for ISR. That is, the TMP82C59A processes an interrupt request by assuming that ISR bit and IRR bit corresponding to IMR bit set at "1" have not been set. This mode is set by setting ESMM = 1 and SMM = 1 by OCW3.

Further, when ESMM = 1 and SMM = 0 are assigned by OCW3, this mode is cleared to the normal mode. The IMR programming is made by OCW1.

(8) POLL COMMAND

This mode is used in a state where the internal interrupt enable flip-flop of MPU is disabled and no interrupt is authorized. The service to the device is made by using the poll command. The poll command specifies P = 1 in OCW3. The mode becomes now the poll mode. When the read operation ($\overline{RD}$ = 0, $\overline{CS}$ = 0) is made on the TMP82C59A, the following output is made on the data bus:

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
I	–	–	–	–	W_2	W_1	W_0

W_0~W_2 : Binary code of highest priority interrupt request among interrupt requests to the interrupt request inputs.

I : There is an interrupt request to MPU when I = 1.

050489

Figure 4.7 Poll Mode Data Format

The TMP82C59A interpretes $\overline{RD}$ signal as the interrupt acknowledge and when D_7 = 1 is output, sets corresponding ISR bit. This poll mode is valid for a period from $\overline{WR}$ (P = 1 in OCW3) to next $\overline{RD}$ ($\overline{CS}$ = 0). Further, an interrupt request to be serviced is determined at the time when the mode is made to the poll mode and end even when a new or high priority interrupt request is sent between $\overline{WR}$ and $\overline{RD}$, it is not accepted.

(9) READING STATUS

MPU is capable of reading the contents of 3 registers (IRR, ISR, IMR). When the reading operation is made at $A_0 = 0$, the content of IRR or ISR can be read out. Selection of IRR and ISR is made by OCW3. When RR is set at 1 and RIS at 0, IRR is assigned and when RR and RIS are set at 1, ISR is assigned.

This assignment is kept stored without necessity for performing at every reading operation. IMR is read when $A_0 = 1$. If the poll mode is specified before the reading operation, the poll command has priority.

(10) EDGE TRIGGERED MODE / LEVEL TRIGGERED MODE

This mode is selected by LTIM of ICWI.

When LTIM is 0, the edge triggered mode is selected and interrupt request is triggered at the leading edge of the interrupt request signal and kept continued by holding "H" level. When LTIM is 1, the level triggered mode is selected and interrupt request is recognized by "H" level of the interrupt request signal. For both modes it is necessary to hold the interrupt request input at "H" level by triggering it till the fast $\overline{INTA}$ signal is output from MPU. If the interrupt request input is at "L" level when $\overline{INTA}$ signal is output from MPU, the same operations as those when interrupt requests are generated at IR_7 are performed but ISR bits are not set.

(11) SPECIAL FULLY NESTED MODE

This mode is used to give priority to the interrupt request input for the slave devices when they are cascade connected.

This mode is assigned to the master TMP82C59A when SFNM is 1 in ICW4.

With the exception of the following 2 points, this mode is identical to the fully nested mode.

[1] Even when an interrupt request from a slave device is being serviced, the master device accepts a higher priority interrupt request from the same slave device without ignoring it. (In the fully nested mode, a higher priority interrupt request from the slave device that is now being serviced is ignored and interrupt requests from a higher priority slave device only are accepted.)

[2] When an interrupt request from a slave device is being serviced, it is necessary to check by a software as to whether the interrupt request is only one interrupt request from that slave device.

When the service ended, after the non-specific EOI is output to that slave device, MPU has to check whether all ISR bits of that slave device are "0". If they are all "0", that slave has no interrupt request being serviced and therefore, the non-specific EOI is output to the master device to allow acceptance of interrupt request from the lower priority slave devices.

Otherwise, the non-specific EOI must not be output to the master device.

(12) BUFFERED MODE

This mode is to output an enable signal to a data bus buffer from the $\overline{SP}/\overline{EN}$ terminal when the data bus buffer is needed for the data bus on a large system. Under this mode, "L" level signal is output to the $\overline{SP}/\overline{EN}$ terminal whenever the data bus output of the TMP82C59A is enabled. The assignment of this mode is made by ICW4 simultaneously with the assignment of the master/slave devices.

(13) CASCADE MODE

The TMP82C59A is able to process interrupt requests up to 64 levels by one master and 8 slave devices.

The cascading is shown in Figure 4.8. The master TMP82C59A selects the slave devices by the identification codes fully using 3 cascade lines. INT output of each slave device is connected to the interrupt request inputs of the master device. Further, the identification codes corresponding to respective connections are assigned for the slave devices by ICW3.

When interrupt request are generated at the interrupt request inputs of the slave devices and accepted, the master device outputs the identification code to the slave device at the first $\overline{INTA}$ signal trailing edge to output vector address or pointer. This identification code is kept maintained to the leading edge of the final $\overline{INTA}$ signal. Normally, the master device outputs "L" level signal to all cascade line. EOI command must be output twice; to the master and second, to the slave corresponding to the interrupt service. Further, an address decoder is required to activate to the $\overline{CS}$ input of each TMP82C59A.

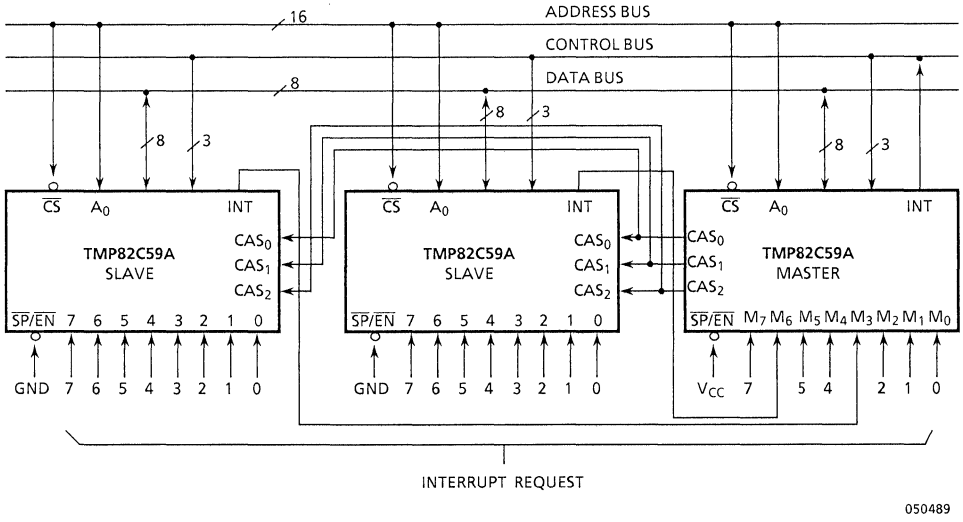


Figure 4.8 CASCADING

5.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING
VCC	VCC Supply Voltage (with respect to VSS (GND))	-0.5 to + 7V
VIN	Input Voltage	-0.5 to V_{CC} + 0.5V
VOUT	Output Voltage	-0.5 to V_{CC} + 0.5V
PD	Power Dissipation	250mW
Tsol	Soldering Temperature (Soldering Time 10 sec)	260℃
Tstg	Storage Temperature	-65℃ to + 150℃
Topr	Operating Temperature	-40℃ to + 85℃

050489

5.2 DC CHARACTERISTICS

Ta = − 40 to + 85°C, VCC = 5 ± 10%, Vss (GND) = 0V, Unless otherwise noted.

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
VIL	Input Low Voltage		-0.5	—	0.8	V
VIH	Input High Voltage		2.2	—	VCC + 0.5	V
VOL	Output Low Voltage	IOL = 2.2mA	—	—	0.45	V
VOH1	Output High Voltage	IOH1 = -400μA	2.4	—	—	V
VOH2	Output High Voltage	IOH2 = -100μA	VCC-0.8	—	—	V
IIL	Input Leak Current	0V≦VIN≦VCC	—	—	± 10	μA
IIOFL	Output Leak Current	0.45V≦VIN≦VCC	—	—	± 10	μA
ILIR	Input Current (IR)	VIN = 0V	—	—	-300	μA
		VIN = VCC	—	—	10	μA
ICC1	Operating Supply Current	I/O CYCLE = 1μS VIH = VCC-0.2V VIH = 0.2V	—	—	5	mA
ICC2	Stand-by Supply Current	VIH = VCC-0.2V VIL = 0.2V	—	—	10	μA

050489

5.3 INPUT CAPACITANCE

Ta = 25°C, VCC = VSS (GND) = 0V

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
CIN	INPUT CAPACITANCE	fC = 1 MHz Unmeasured pins, 0V	—	—	10	pF
CI/O	INPUT/OUTPUT CAPACITANCE		—	—	20	pF

050489

5.4 AC CHARACTERISTICS

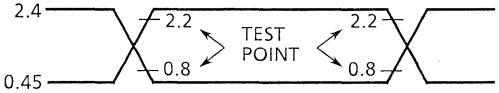
Ta = −40°C to +85°C, VCC = 5V ± 10%, VSS (GND) = 0V

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
TAHRL	A_0 / $\overline{CS}$ Setup Time ($\overline{RD}$)		0	—	—	ns
TRHAX	A_0 / $\overline{CS}$ Hold Time ($\overline{RD}$)		0	—	—	ns
TRLRH	$\overline{RD}$ Pulse Width		160	—	—	ns
TAHWL	A_0 / $\overline{CS}$ Setup Time ($\overline{WR}$)		0	—	—	ns
TWHAX	A_0 / $\overline{CS}$ Hold Time ($\overline{WR}$)		0	—	—	ns
TWLWH	$\overline{WR}$ Pulse Width		120	—	—	ns
TDVWH	D_0 to D_7 Setup Time ($\overline{WR}$)		120	—	—	ns
TWHDX	D_0 to D_7 Hold Time ($\overline{WR}$)		0	—	—	ns
TJLJH	Input IR Low Level Pulse width (Edge Trigger Mode)		100	—	—	ns
TCVIAL	Casecade Setup Time (Second or Third $\overline{INTA}$)		40	—	—	ns
TRHRL	$\overline{RD}$ to Next Command		160	—	—	ns
TWHWL	$\overline{WR}$ to Next Command		190	—	—	ns
TCHCL	End of Command to next Command (Not Same)		250	—	—	ns
	End of $\overline{INTA}$ sequence to next $\overline{INTA}$ sequence					
TRLDV	Valid Data Delay ($\overline{RD}$ / $\overline{INTA}$)	D_7 to D_0 CL = 100pF	—	—	120	ns
TRHDZ	Data Floating ($\overline{RD}$ / $\overline{INTA}$)		10	—	85	ns
TJHIH	Interrupt Output Delay (IR)	INT CL = 100pF	—	—	300	ns
TIALCV	Valid Cascade Delay ($\overline{INTA}$)		—	—	360	ns
TRLEL	Enable Active ($\overline{RD}$ / $\overline{INTA}$)	CAS_0 to 2 CL = 100pF	—	—	100	ns
TRHEH	Enable Inactive ($\overline{RD}$ / $\overline{INTA}$)		—	—	150	ns
TAHDV	Valid Data Delay (A_0 / $\overline{CS}$)		—	—	200	ns
TCVDV	Valid Data Delay (CAS_0 to CAS_2)		—	—	200	ns

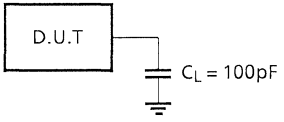
050489

AC CHARACTERISTICS TEST CONDITION

AC Testing I / O Waveform



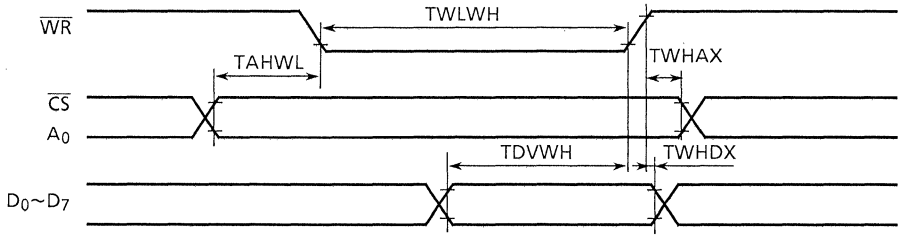
AC Testing Load Circuit



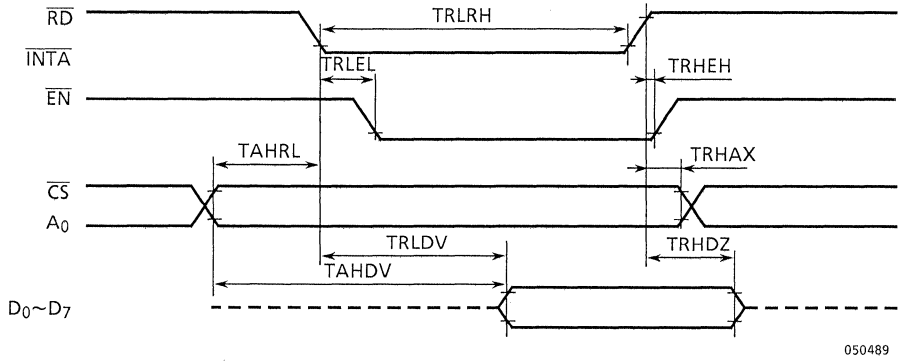
050489

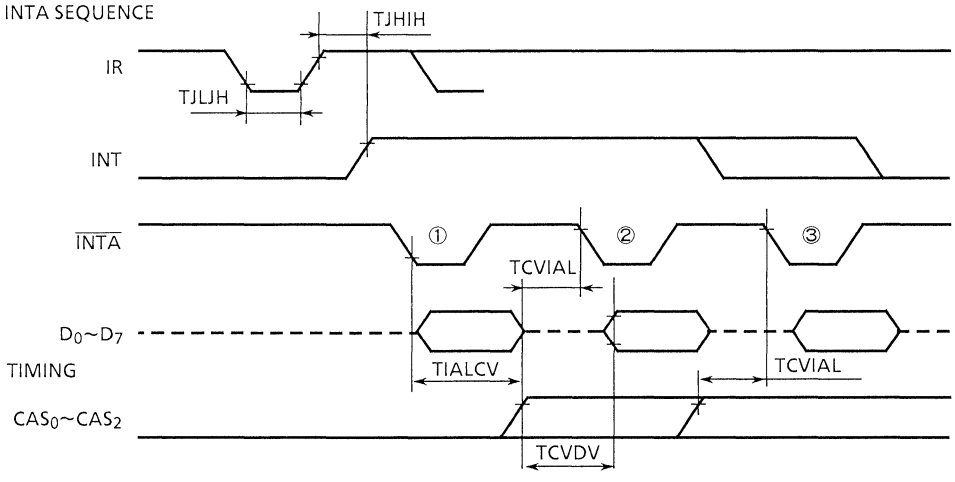
6. TIMING WAVEFORMS

WRITE OPERATION

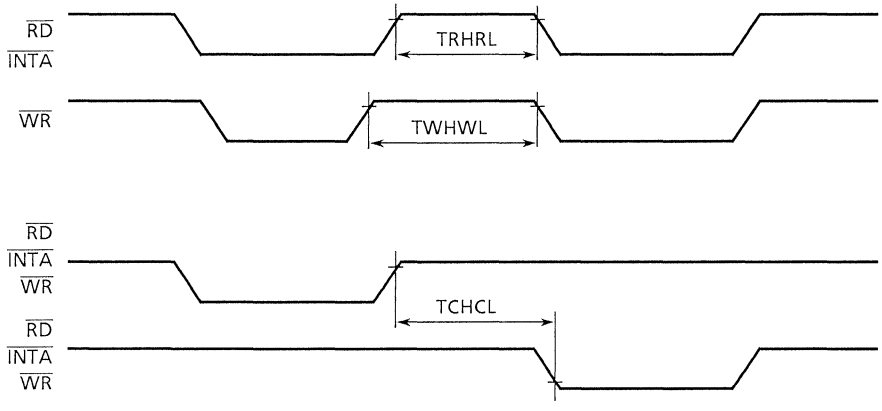


READ AND INTA OPERATION





OTHER TIMING



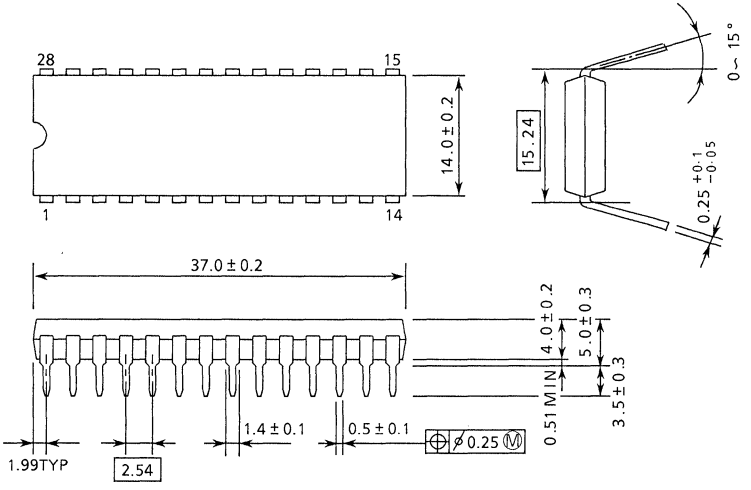
050489

7. EXTERNAL DIMENSION VIEW

7.1 28 pins PRASTIC DIP

DIP28-P-600

Unit : mm



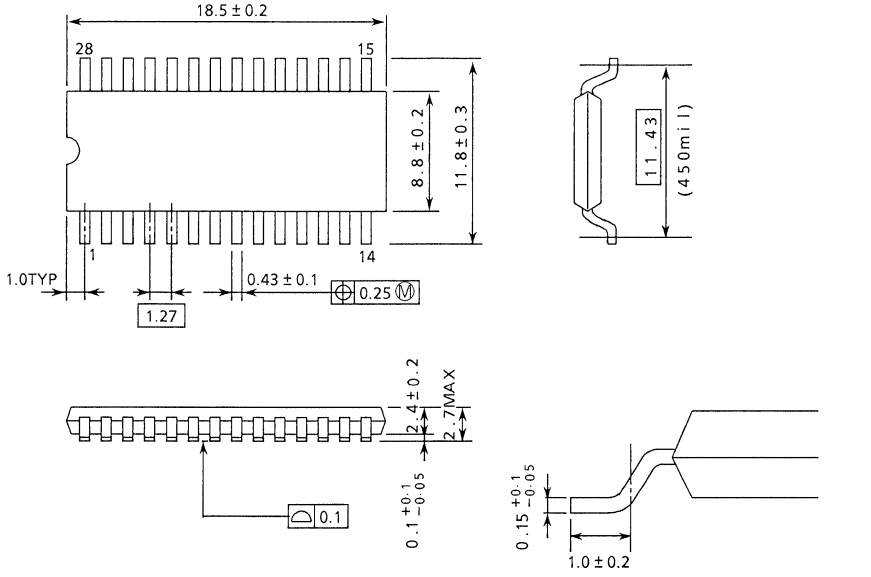
270289

Note : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical position with respect to NO. 1 and NO. 28 leads.

7.2 28 pins SMALL OUTLINE PACKAGE

SOP28-P-450

Unit : mm



270289

Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

TOSHIBA MOS TYPE DEGITAL
INTEGRATED CIRCUIT
Silicon Monolithic N-Channel Silicon Gate MOS

TMP8259AP

PROGRAMMABLE INTERRUPT CONTROLLER

1. GENERAL DESCRIPTION

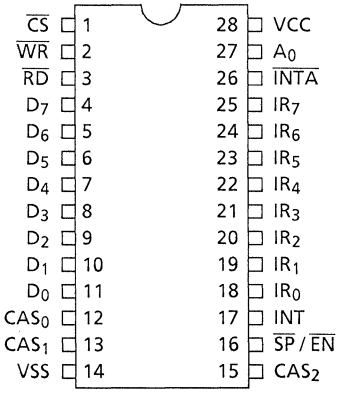
TMP8259AP is a programmable interrupt controller. It handles up to eight vectored priority interrupts for the MPU. It is cascadable for up to 64 vectored priority interrupts without additional circuitry.

FEATURES

- Eight Level Priority Controller.
- Expandable to 64 Level.
- Interrupt Modes, Interrupt Mask, Vectored Address Programmable.
- Single +5V Power Supply.
- Supports 8085A, 8086 Microcomputer Interrupt Sequence.

2. PIN CONNECTIONS (TOP VIEW)

TMP8259AP



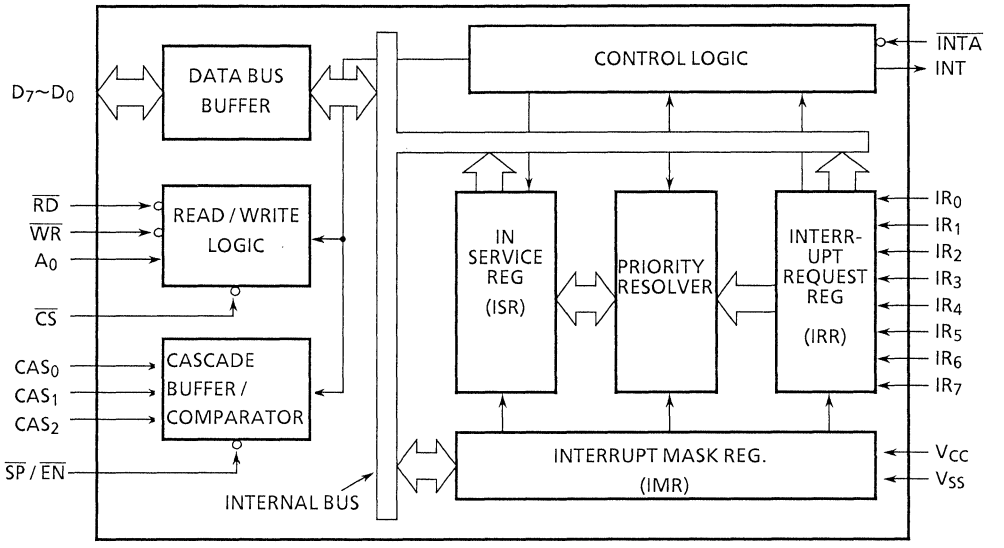
050489

3. PIN NAMES AND PIN DESCRIPTION

Pin Name	Input / Output	Function
$\overline{CS}$	Input	Chip Select Input. A low on this pin enables $\overline{RD}$ and $\overline{WR}$ communication beween the MPU and the TMP8259AP. $\overline{INTA}$ functions are independent of $\overline{CS}$.
$\overline{WR}$	Input	Write Control Input. A low on this pin when $\overline{CS}$ is low enables the TMP8259AP to accept command words from MPU.
$\overline{RD}$	Input	Read Control Input. A low on this pin when $\overline{CS}$ is low enables the TMP8259AP to release status onto the data bus for the MPU.
D_0 to D_7	Input / Output	Bidirectional Data Bus. Command status and interrupt-vector information is transferred via this bus.
CAS_0 to CAS_2	Input / Output	Casecade Lines. The CAS lines from a private TMP8259AP bus to control a multiple TMP8259AP structure. These pins are outputs for a master TMP8259AP and inputs for a slave TMP8259AP.
$\overline{SP}$ / $\overline{EN}$	Input / Output	Slave Program / Enable buffer. This is a dual function pin. When in the buffered mode is can be used as on Output to control buffer transceivers ($\overline{EN}$). When not in the buffered mode it is used as an input to designate a master TMP8259AP ($\overline{SP}$ = 1) or a slave ($\overline{SP}$ = 0).
INT	Output	Interrupt Request Output. This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the MPU. It is connected to MPU's interrupt pin.
IR_0 to IR_7	Input	Interrupt Request Inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on a IR input (Level Triggered Mode.)
$\overline{INTA}$	Input	Interrupt Ackowledge INPUT. This pin is used to enable TMP8259AP interrupt-vector data onto the data bus by a sequence of interrupt ackowledged pulses issued by the MPU.
A_0	Input	A_0 address line. This pin acts in conjunction with the $\overline{CS}$, $\overline{WR}$ and $\overline{RD}$ pins. It is used by the TMP8259AP to decipher various command words the MPU writes and status the MPU wishes to read. It is typically connected to the MPU A_0 address line.
VCC		+ 5V Power Supply
VSS		Ground

050489

4. BLOCK DIAGRAM



5.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING
VCC	VCC Supply Voltage (with respect to VSS (GND))	-0.5 to +7V
VIN	Input Voltage	-0.5 to +7V
PD	Power Dissipation	1W
Tsol	Soldering Temperature (Soldering Time 10 sec)	260℃
Tstg	Storage Temperature	-65℃ to +150℃
Topr	Operating Temperature	0℃ to 70℃

050489

5.2 DC CHARACTERISTICS

Ta = 0 to +70°C, VCC = 5V ± 10%, Vss (GND) = 0V

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
VIL	Input Low Voltage		-0.5	—	0.8	V
VIH	Input High Voltage		2.2	—	VCC + 0.5	V
VOL	Output Low Voltage	IOL = 2.2mA	—	—	0.45	V
VOH	Output High Voltage	IOH = -400μA	2.4	—	—	V
VOH (INT)	Output High Voltage (INT)	IOH = -100μA	3.5	—	—	V
		IOH = -400μA	2.4	—	—	V
IIL	Input Leak Current	0V≦VIN≦VCC	—	—	±10	μA
IIOFL	Output Leak Current	0.45V≦VIN≦VCC	—	—	±10	μA
ILIR	Input Current (IR)	VIN = 0V	—	—	-300	μA
		VIN = VCC	—	—	10	μA
ICC	Operating Supply Current		—	—	85	mA

050489

5.3 INPUT CAPACITANCE

Ta = 25°C, VCC = VSS

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
CIN	INPUT CAPACITANCE	fC = 1 MHz Unmeasured pins, 0V	—	—	10	pF
CI/O	INPUT/OUTPUT CAPACITANCE		—	—	20	pF

050489

5.4 AC CHATACTERISTICS

Ta = 0°C to + 70°C, VCC = 5V ± 10%, VSS = 0V

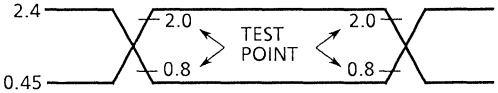
SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
TAHRL	A_0 / $\overline{CS}$ Setup Time ($\overline{RD}$)		0	—	—	ns
TRHAX	A_0 / $\overline{CS}$ Hold Time ($\overline{RD}$)		0	—	—	ns
TRLRH	$\overline{RD}$ Pulse Width		235	—	—	ns
TAHWL	A_0 / $\overline{CS}$ Setup Time ($\overline{WR}$)		0	—	—	ns
TWHAX	A_0 / $\overline{CS}$ Hold Time ($\overline{WR}$)		0	—	—	ns
TWLWH	$\overline{WR}$ Pulse Width		290	—	—	ns
TDVWH	D_0 toD_7 Setup Time ($\overline{WR}$)		240	—	—	ns
TWHDX	D_0 toD_7 Hold Time ($\overline{WR}$)		0	—	—	ns
TJLJH	Input IR Low Level Pulse width (Edge Trigger Mode)		100	—	—	ns
TCVIAL	Casecade Setup Time (Second or Third $\overline{INTA}$)		55	—	—	ns
TRHRL	$\overline{RD}$ to Next Command		160	—	—	ns
TWHWL	$\overline{WR}$ to Next Command		190	—	—	ns
*TCHCL	End of Command to next Command (Not Same)		500	—	—	ns
	End of $\overline{INTA}$ sequence to next $\overline{INTA}$ sequence					
TRLDV	Valid Data Delay ($\overline{RD}$ / $\overline{INTA}$)	D_7 to D_0 CL = 100pF	—	—	200	ns
TRHDZ	Data Floating ($\overline{RD}$ / $\overline{INTA}$)		10	—	100	ns
TJHIH	Interrupt Output Delay (IR)	INT CL = 100pF	—	—	350	ns
TIALCV	Valid Cascade Delay ($\overline{INTA}$)		—	—	565	ns
TRLEL	Enable Active ($\overline{RD}$ / $\overline{INTA}$)	CAS_0 to 2 CL = 100pF	—	—	125	ns
TRHEH	Enable Inactive ($\overline{RD}$ / $\overline{INTA}$)		—	—	150	ns
TAHDV	Valid Data Delay (A_0 / $\overline{CS}$)		—	—	200	ns
TCVDV	Valid Data Delay (CAS_0 to CAS_2)		—	—	300	ns

050489

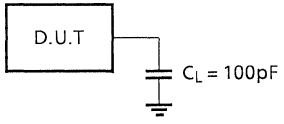
* Worst case timing for TCHCL in an actual microprocessor system is typically much greater than 500 ns (i.e. 8085A = 1.6μs, 8085A – 2 = 1μs, 8086 = 1μs, 8086 – 2 = 625μs)

AC CHARACTERISTICS TEST CONDITION

AC Testing I / O Waveform



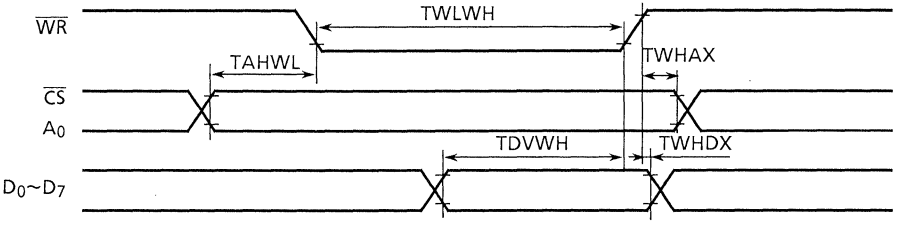
AC Testing Load Circuit



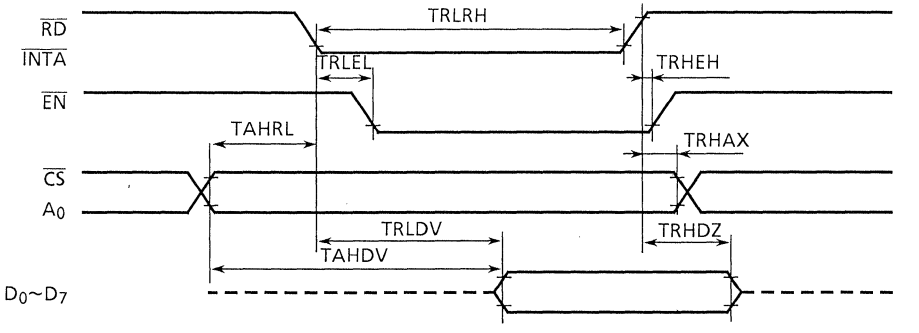
050489

6. TIMING WAVEFORMS

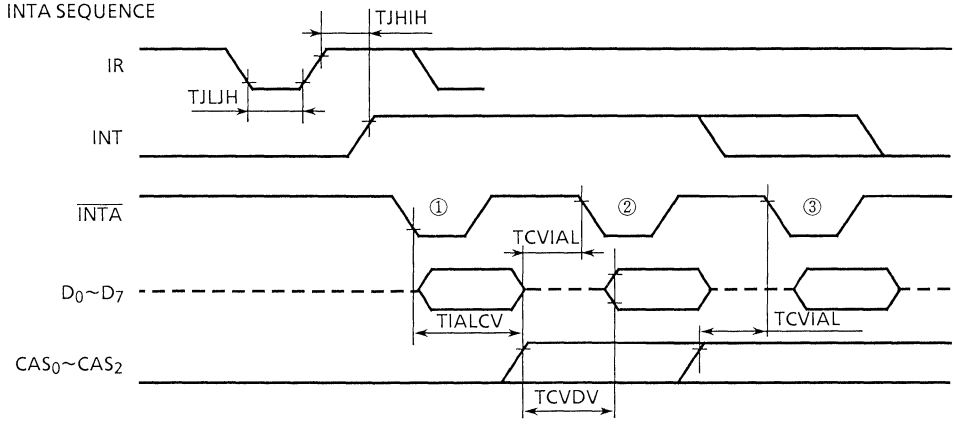
WRITE OPERATION

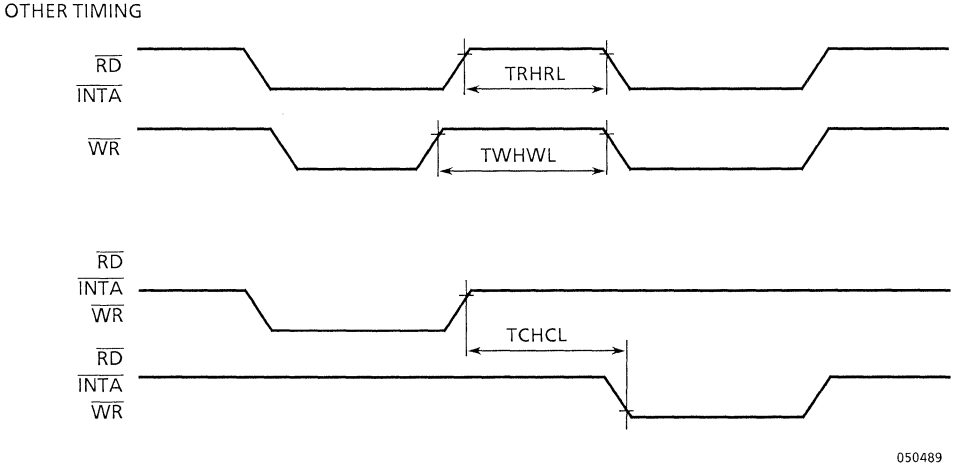


READ AND INTA OPERATION



050489


INTA SEQUENCE
TJHIH
IR
TJLJH
INT
INTA
①
②
③
TCVIAL
D0~D7
TIALCV
TCVIAL
CAS0~CAS2
TCVDV


OTHER TIMING
RD
INTA
TRHRL
WR
TWHWL
RD
INTA
WR
TCHCL
RD
INTA
WR

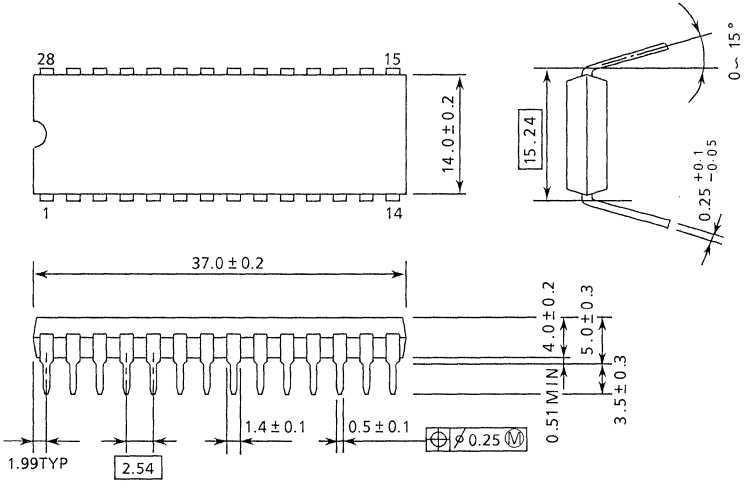
050489

7. EXTERNAL DIMENSION VIEW

28 pins PRASTIC DIP

DIP28-P-600

Unit : mm



270289

Note: Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical position with respect to No.1 and No.28 leads.

MULTIMODE DMA CONTROLLER

TMP82C37AP-5/TMP82C37AM-5/TMP82C37AT-5

1. GENERAL DESCRIPTION

The TMP82C37AP-5/AM-5/AT-5 (hereinafter referred to as TMP82C37A) is a multimode direct memory access (DMA) controller. The TMP82C37A improves the system function by directly transferring information between the system memory and external devices. Memory-to-Memory data transfer capability is also provided.

The TMP82C37A is provided with versatile programmable control functions in order to improve data throughput.

The TMP82C37A is used with an 8-bit address register connected externally. The TMP82C37A has four built-in independent channels and it is possible to expand channels through cascade connection.

There are three basic data transfer modes which are programmable by the user. Each channel is programmable individually and autoinitialization is possible by End of Process ($\overline{EOP}$) signal.

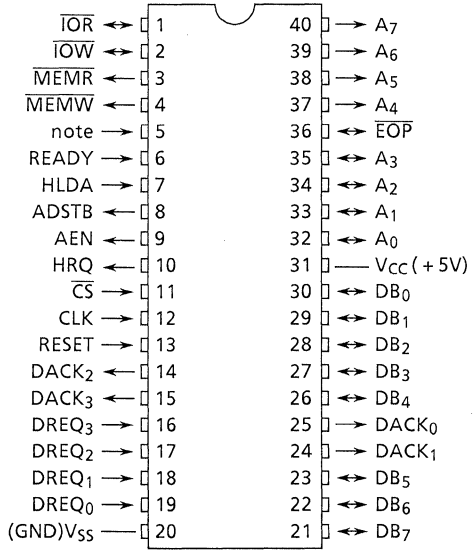
Each channel has the maximum 64K capability for both address and word count. $\overline{EOP}$ signal is capable of terminating data transfer between DMA and memories. $\overline{EOP}$ signal is useful for block search or verify or for terminating erroneous service.

2. FEATURES

- Four independent DMA channels available
- Three transfer modes available; block, demand, and single transfer modes
- Independent auto initialize function provided to each of all channels
- Memory-to-Memory transfer
- Address increment or decrement
- All DMA request disabled by disabling the master system
- Individual DMA request enable/disable control
- Unrestricted channel expansion by cascade connection
- End of Process ($\overline{EOP}$) input for terminating transfer
- Software DMA Request
- Polarity control provided for DREQ signal and DACK signal
- Option for increasing transfer speed up to 2.5M word/sec (@5MHz)
- Single +5V power supply
- Low power consumption 5mA TYP. @5MHz
- Extend operating temperature −40°C to +85°C

3. PIN CONNECTION (TOP VIEW)

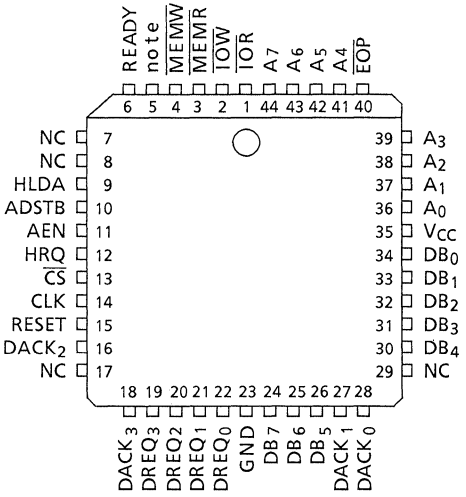
3.1 TMP82C37AP-5 (DIP), TMP82C37AM-5 (SOP)



050489

Note : PIN 5 must be connected to V_{CC} or opened.

3.2 TMP82C37AT-5 (PLCC)



050489

Note : PIN 5 must be connected to Vcc or must be opened.
NC : No Connection

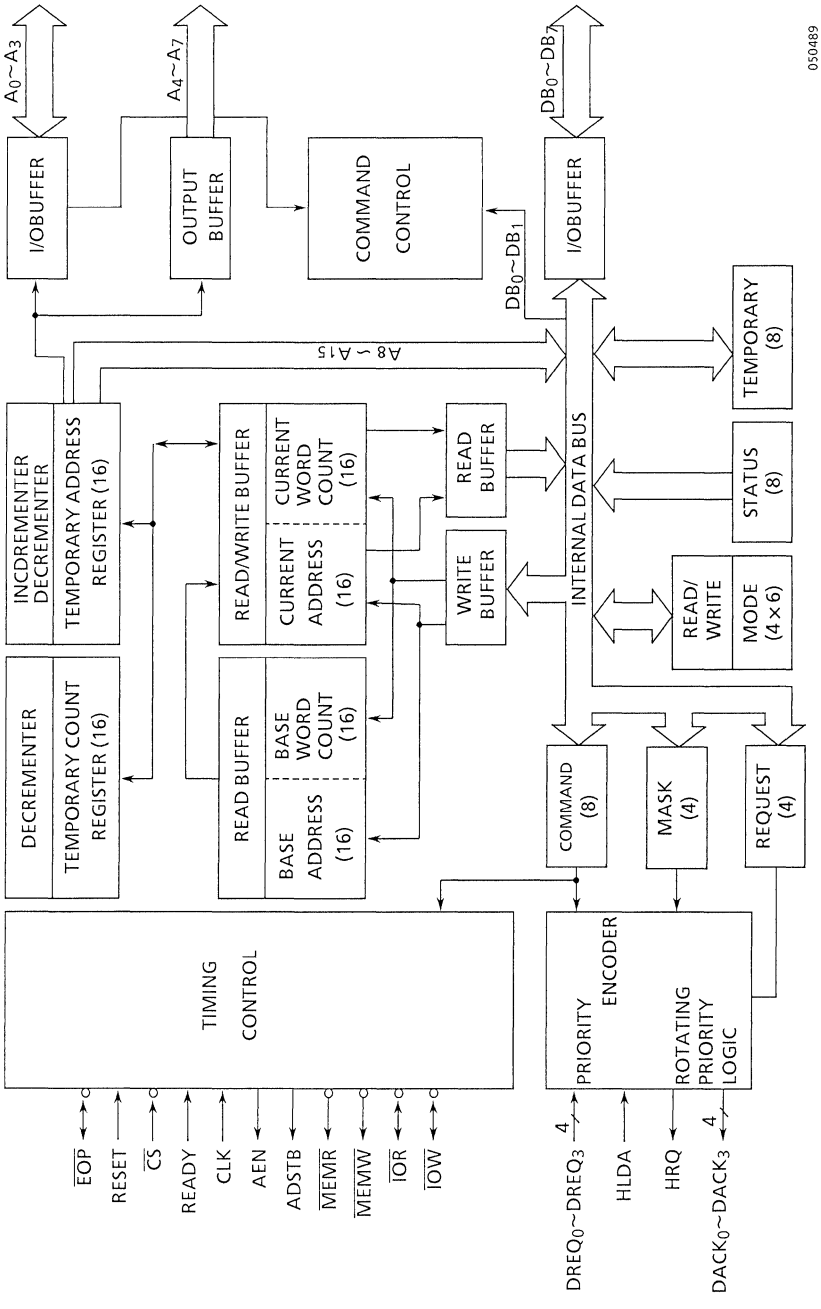


Figure 3.1 Block Diagram of TMP82C37A

4. PIN NAME & FUNCTION

- V_{CC}

  +5V power supply

- V_{SS}

  Ground

- CLK (Clock, Input)

  This input controls the internal operation and data transfer rate of the TMP82C37A.

- $\overline{CS}$ (Chip Select, Input)

  This input is low active and used to select the TMP82C37A as an I/O device during an I/O read or I/O write by the host MPU. If $\overline{IOR}$ or $\overline{IOW}$ is toggled following each transfer when a host MPU and the TMP82C37A are transferring data mutually, $\overline{CS}$ may be kept at LOW.

- RESET (Reset, Input)

  This input is asynchronous input to clear the command, status, request and temporary registers. In addition, this input is used to clear First/Last flip-flops and set the mask register. Following the reset, the TMP82C37A is placed in the idle cycle.

- READY (Ready, Input)

  This input is used to extend the memory or I/O read and write pulses in DMA cycle in order to adapt to low speed memories or I/O peripheral devices.

- HLDA (Hold Acknowledge, Input)

  By this signal, the TMP82C37A knows that the system bus control is turned over from MPU.

- $DREQ_0$-$DREQ_3$ (DMA Request, Input)

  DMA request signals are input from peripheral circuits. If priority is fixed, the highest priority is given to $DREQ_0$ and the lowest priority to $DREQ_3$. Polarity of DREQ is programmable. DREQ becomes high active by RESET.

- DB_0-DB_7 (Data Bus, Input/Output)

  The Data Bus are bidirectional 3-state lines connecnted to the system data bus. During MPU is in I/O read state, output is enabled and contents of the registers (address, status, temporary and word count) are output to MPU. During MPU is in I/O write state, the data bus serves as input and it becomes possible to program the control register of the TMP82C37A.

  During the DMA cycle, the most significant 8 bits of address are output on the data bus and latched by ADSTB signal externally. During the Memory-to-Memory transfer, the data of the source memory location are loaded into the temporary register of the TMP82C37A by the read operation and the contents of the temporary register are output to the destination memory location by the write operation.

- $\overline{IOR}$ (I/O Read, Input/Output)

  I/O read is a bidirectional, low active and 3-state signal. During the idle cycle, this signal serves as an input control signal used by MPU to read the control registers of the TMP82C37A. During the active cycle, this signal serves as an output control signal used by the TMP82C37A to access data from the peripheral circuit during the DMA read and transfer.

- $\overline{IOW}$ (I/O Write, Input/Output)

  I/O write is a bidirectional, low active, 3-state signal. During the idle cycle, this signal serves as an input control signal used by MPU to load the information to the TMP82C37A. During the active cycle, this signal served as an output control signal used by TMP82C37A to load the data to the peripheral. For write to the TMP82C37A by MPU, the leading edge of the write signal ($\overline{IOW}$) is required for every data transfer. It is not possible to write more than two data by toggling $\overline{CS}$ while holding the $\overline{IOW}$ pin at low level.

- $\overline{EOP}$ (End of Process, Input/Output)

  $\overline{EOP}$ (End of Process) is a signal relative to the end of DMA service, and is a low active, bidirectional and open drain signal. When the channel word count reaches from 0000H to FFFFH, the TMP82C37A outputs low pulse of $\overline{EOP}$ to peripheral devices as the end signal.

  In addition, it is also possible to pull $\overline{EOP}$ to the low level by peripheral device in order to cause the end of process.

  When $\overline{EOP}$ is received (internally or externally), the channel which is presently active terminates the service, sets that TC bit of the status register and resets that request bit.

  If that channel is programmed for auto initialization, that current register is updated from the base register. In all other cases, mask bit is set and the content of that register remains unchanged.

  During the Memory-to-Memory transfer, $\overline{EOP}$ is output when TC of channel 1 is produced. $\overline{EOP}$ is always used for channels with active DACK and external $\overline{EOP}$ has no connection when $DACK_0$-$DACK_3$ are all inactive.

  $\overline{EOP}$ is an open drain signal and therefore, requires an external pull-up resistor.

- A_0-A_3 (Address, Input/Output)

  The four least significant address lines are the bidirectional 3-state signals. In the idle cycle, these lines serve as the input signals and used by MPU for write/read of the control register. In the active cycle, they serve as the output signals and become low order 4 bits of output address.

- A_4-A_7 (Address, Output)

  The four most significant address lines are 3-state output signals. These lines are enabled for the period of DMA service only.

- HRQ (Hold Request, Output)

  This is the hold request signal to MPU, and is used to request the system bus control. HRQ is output by the TMP82C37A according to a software request or unmasked DREQ.

- $DACK_0$-$DACK_3$ (DMA Acknowledge, Output)

  The DMA acknowledge lines indicate that channels are active. Normally, these are used for selecting peripheral devices. Only one DACK becomes active but it does not become active unless DMA is controlling the system bus. Polarity of these lines are programmable. After reset, they initialize low active.

- AEN (Address Enable, Output)

  Address Eanble is a high active signal and used to enable output of the external latch which holds high order byte of address and to disable the system bus during the DMA cycle.

  During the DMA transfer, HLDA and AEN are used to disable all I/O except programmed I/O. The TMP82C37A disables $\overline{CS}$ input for DMA transfer to prevent itself from being selected automatically.

- ADSTB (Address Strobe, Output)

  This signal is a strobe output to an external latch circuit and is used to latch high order 8-bit address from DB_0-DB_7.

- $\overline{MEMR}$ (Memory Read, Outputd)

  This is a low active 3-state output used for transferring data from a memory to a peripheral device or for data accessing from a selected memory during the Memory-to-Memory transfer.

- $\overline{MEMW}$ (Memory Write, Output)

  This is a low active 3-state output used for transferring data from a peripheral device to a memory or for writing data into a selected memory during the Memory-to-Memory transfer.

5. OPERATIONAL DESCRIPTION

5.1 DMA OPERARTION

The TMP82C37A has two operations; idle cycle and active cycle. Each of these cycles consists of several states.

On the TMP82C37A, it is possible to consider 7 states each of which consists of one clock cycle. State I (SI) is an idle state. This is such a state as there is no valid DMA request pending. SI is a program condition state which is programmable by MPU.

State 0 (S0) is the first DMA service state. This is a state that the TMP82C37A made a hold request to MPU but not yet received the acknowledge signal from MPU. When the acknowledge signal is recieved from MPU, the transfer is started.

S1, S2, S3 and S4 are the DMA service states. If more time is required by the transfer, it is possible to insert the wait state (SW) before S4 by READY input to the TMP82C37A.

In the Memory-to-Memory transfer, in order to assure complete transfer, read from the memory and write to the memory are required. 8 states are necessary for one transfer. The first four states (S11, S12, S13 and S14) are read from the memory and the latter four state (S21, S22, S23 and S24) are write to the memory.

The temporary data register is used as an intermediate storage area of memory bytes.

5.2 IDLE CYCLE

When DMA service is not requested by channels, the TMP82C37A enters into the idle cycle and is placed in SI state. In order to check if the channels request DMA service, the TMP82C37A samples DREQ for every clock.

The TMP82C37A also samples $\overline{CS}$ to check if MPU is requesting read or write of internal registers. When $\overline{CS}$ is low and HLDA is also low, the TMP82C37A is placed in the program condition.

At this time, MPU is able to change or check the content of any internal register through read or write from that register.

Address lines A_0-A_3 are input signals and used for selecting a register being read or written. $\overline{IOR}$ and $\overline{IOW}$ are used for selecting read or write and decide read/write timing.

The internal flip-flop is used for generating address extension bits according to number and size of internal registers. (First/Last flip-flop) This bit is used for deciding high or low order byte of 16-bit address and word count register.

The flip-flop is reset by the master clear or reset. In addition, this flip-flop also can be reset by an independent software command. On a special software command, the execution in the TMP82C37A program conditon is possible. These commands are decoded as in the address setting when both $\overline{CS}$ and $\overline{IOW}$ are active.

The data bus is not used for this command. This command is available in three types; clear First/Last flip-flop, master clear and clear mask register.

5.3 ACTIVE CYCLE

When the TMP82C37A is in idel cycle and the channels are requesting DMA service, the TMP82C37A outputs HRQ to MPU and goes into the active cycle. In this cycle, the DMA service for any one of 4 modes is executed.

5.3.1 Single Transfer Mode:

In this mode, the TMP82C37A performs a single byte transfer during each HRQ/HLDA handshake. When DREQ becomes active, HRQ becomes active. After MPU responds by driving HLDA active, a single byte transfer will take place. After the transfer HRQ becomes inactive, its word count is decreased, and address is increased or decreased. When word count changes from 0000H to FFFFH, a terminal signal is generated and if the channels are programmed, the auto initialization is made.

To execute the single byte transfer, it is necessary to hold DREQ until DACK corresponding each DREQ becomes active. If DREQ is continuously active, HRQ becomes inactive following each transfer and then, becomes active again, and the new single byte is executed following the leading edge of HLDA.

On the 8085A system, one machine cycle can be executed during the DMA transfer.

5.3.2 Block Transfer Mode:

In this mode the TMP82C37A continues the transfer until terminal count (TC) is generated or an external End of Process signal ($\overline{EOP}$) is encountered. Here, TC is produced when the word count changes from 0000H to FFFFH.

What is required for DREQ is to hold it in active state until DACK becomes active. Auto initialization (if so programmed) is taken place at the end of DMA service.

5.3.3 Demand Transfer Mode:

In this mode the TMP82C37A continues the transfer until TC is produced or $\overline{EOP}$ is or DREQ becomes inactive. Thus, it is possible for a device, which is requesting the DMA service, to suspend the transfer by making DREQ inactive. The service is resumed when DREQ is made active again. It is possible to read an intermediate value of address and word count from the current address and current word count register of the TMP82C37A while the system bus is returned to MPU during execution of the DMA service.

The auto initialization is taken place following TC or $\overline{EOP}$ at the end of DMA service. In order to perform a new DMA service following the auto intialization, the active edge of DREQ is necessary.

5.3.4 Cascade Mode:

This mode is used when the TMP82C37A is cascade connected for a simple system expansion. HRQ and HLDA of the additional TMP82C37A are connected to DREQ and DACK of the first TMP82C37A. DMA request to the TMP82C37A which is added for the purpose of system expansion is authorized by the priority circuit of the first TMP82C37A.

If the priority is already decided, the additional device must wait till the acknowledge request. The cascade channel of the first TMP82C37A is used only for deciding priority of the additional TMP82C37A and therefore, the channel itself does not output address nor control signal. This is to prevent the added device from colliding with output of the cascade channel. On the TMP82C37A, DACK respond to DREQ. However all other outputs except HRQ are disabled.

The state of cascade connection is shown in Figure 5.1. In Figure 5.1, two levels of DMA are formed. To further expand the TMP82C37A, it is possible to add it to the second level using the TMP82C37A, it is possible to add it to the second level using the remaining channel of the first TMP82C37A. To further add another TMP82C37A, the third level can be formed by cascade connecting it to the second level.

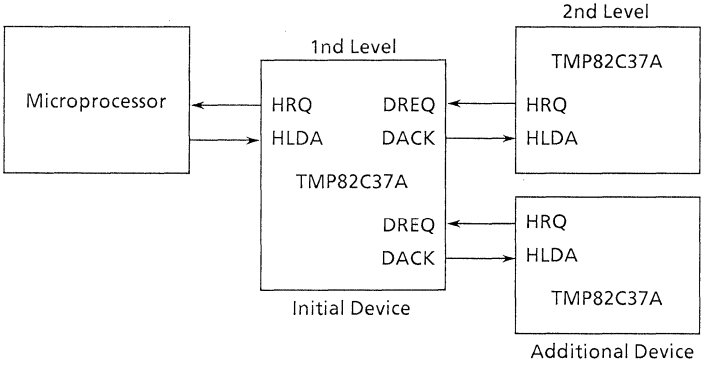


Figure 5.1 Example of Cascade Connection of TMP82C37A

5.4 TRANSFER FORMAT

3 different transfer format are available for 3 active transfer modes.

They are read, write and verify. In the write transfer, data is transferred from I/O device to memory by $\overline{\text{MEMW}}$ and $\overline{\text{IOR}}$. In the read transfer, data is transferred from memory to I/O device by $\overline{\text{MEMR}}$ and $\overline{\text{IOW}}$.

The verify transfer is a pseudo transfer. The TMP82C37A perform such operations as address generation for read or write transfer, answer to $\overline{\text{EOP}}$, etc. However, memory or I/O control line does not become active.

5.5 MEMORY-TO-MEMORY TRANSFER:

The TMP82C37A has the ability of block movement and is capable of transferring data block from one memory address location to another location. When Bit 0 of the command register is programmed at Logic 1, Channel 0 and 1 operate as the Memory-to-Memory transfer channels. Channel 0 serves as source address and Channel 1 as a destination address, and the word count of Channel 1 is used. The Memory-to-Memory transfer is executed when software DMA request is set for Channel 0.

The Memory-to-Memory transfer must use the block transfer mode.

When Channel 0 is programmed as a fixed source address, it is possible to write single source words into a memory block.

When the TMP82C37A is programmed for the Memory-to-Memory transfer, Channel 0 and Channel 1 must be masked. The same value as that is set for Channel 1 must be set for the word count of Channel 0. During the Memory-to-Memory transfer, AEN became active but DACK does not become active.

During the Memory-to-Memory transfer, the TMP82C37A respond to external $\overline{EOP}$ signal. In the block search, the data comparator uses this ($\overline{EOP}$) input to terminate the DMA service when match is found. The Memory-to-Memory transfer timing is shown in Timing Diagram 4.

5.6 AUTO INITIALIZATION:

When Bit 4 of the mode register is set to 1, the channels are set up for the auto initialization. During the auto initialization, data are loaded into the current address and current word count registers from the base address and base word count registers, respectively, following $\overline{EOP}$. The base registers are loaded by MPU simultaneously with the current registers and remain unchanged during the DMA service.

When the channels are under the auto initialization, mask bit is not set by $\overline{EOP}$.

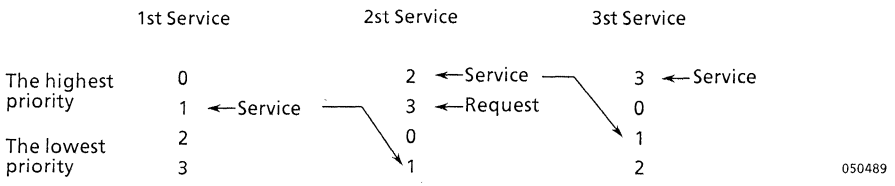
Following the auto initialization, that channel is prepared to execute the service without interposition of MPU.

5.7 PRIORITY:

The TMP82C37A has two types of priority which can be selected by software. The first type is the fixed priority. Channel priority is fixed by channel number. The lowest priority is channel 3, followed by 2, 1, and the highest priority is channel 0.

The second type is the rotating priority. In this type, an accepted channels is then given with the lowest priority .

On the rotating priority in the singel chip DMA system, the highest priority of any one channel comes after no more than three higher priority services have occurred. This rotating priority prevent a specific channel from occupying the system all the time. (See the following below diagram.)



The priority judging circuit selects a channel with the highest priority requesting the DMA service for every active edge of HLDA. Once the channel starts the service, that operation will not be suspended even when the service is demanded by another channel with higher priority. A channel with higher priority can get the control right only after a channel with lower priority relinquished HRQ. Whenever the control is transferred from a channel to another channel, MPU gets the system bus control right. This assures the leading edge of HLDA which is used for selecting a channel with the highest priority.

5.8 COMPRESSED TRANSFER TIMING:

In order to accomplish greater throughput allowed by system characteristics, the TMP82C37A is capable of compressing the transfer time to 2 clock cycles. As can be seen from Timing Diagram 3, State S3 is used to extend readout pulse access time. When State S3 is removed, readout pulse width becomes equal to write pulse width. Then, the transfer will consist of State S2 for changing address and State S4 for executing read/write. State S1 is produced when A_8 to A_{15} are updated (refer to Address Generation). Compressed transfer timing is shown in Timing Diagram 5.

During Memory-to-Memory transfer, compressed transfer is not available.

5.9 ADDRESS GENERATION:

To reduce number of pins, the TMP82C37A has the multiplexed address/data bus. State S1 is used to output high order address byte to the external latch. The trailing edge of ADSTB is used to load the address byte from the data line on the external latch circuit. AEN is used to enable latch outputs from 3 states. Low order address byte is ditectly output by the TMP82C37A.

A_0 to A_7 are connected to address bus. Timing Diagram 3 show the relationship among CLK, AEN, ADSTB, DB_0 to DB_7 and A_0 to A_7.

Address produced during the block and demand transfers are sequential. For many transfer the same address data will be held in the external address latch. This address data changes only when carry or borrow from A_7 to A_8 is produced in the normal sequence. To raise system throughput, on the TMP82C37A, S1 state is executed only for updating A_8 to A_{15} requiring the external latch.

6. DESCRIPTION OF REGISTERS

Register Name	Size	Number
Base address register	16-bit	4
Base word count register	16-bit	4
Curren address register	16-bit	4
Current word count register	16-bit	4
Temporary address register	16-bit	1
Temporary word count register	16-bit	1
Status register	8-bit	1
Command register	8-bit	1
Temporary register	8-bit	1
Mode register	6-bit	4
Mask register	4-bit	1
Request register	4-bit	1

050489

Figure 6.1 Internal Registers

6.1 CURRENT ADDRESS REGISTER:

Each channel has a 16-bit current address register. This register holds addresses that are used during the DMA transfer. After each transfer, this register is automatically incremented or decremented, and intermediate address values are stored in the current address register during the transfer. Write or read of this register is made by MPU. An original value is initialized again by the auto initialization.

The auto initialization is taken place only after $\overline{EOP}$.

6.2 CURRENT WORD COUNT REGISTER:

Each channel has a 16-bit current word count register. For this register, the number of words to be transferned that is one less than that to be tranferred must be programmed. The word counter is decremented after each transfer. Intermediate values of word count are stored in this register during the transfer. When the register value goes from 0000H to FFFFH, TC (Terminal Count) is produced.

When this register is in the program condition, load or read is made by MPU. Following the end of DMA service, this register is initialized to original values again by the auto initialization.

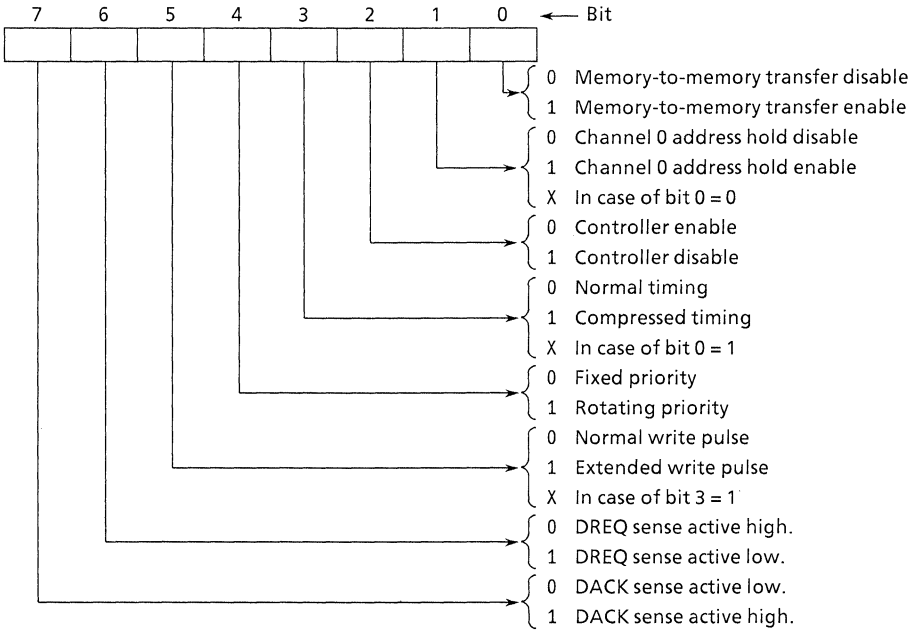
The auto initialization is taken place only when $\overline{EOP}$ is produced. Note that the content of the word count register becomes FFFFH following internally produced $\overline{EOP}$.

6.3 BASE ADDRESS REGISTER, BASE WORD COUNT REGISTER:

Each channel has a pair of registers; the base address register and base word count register. These 16-bit registers store original values of related current registers. These registers are used to store original values of current registers at time of the auto initialization. Write to the base register is made at the same time of write into equivalent current registers during the programming by MPU. Therefore, write into the current registers which store intermediate values are made over these intermediate values. The base register cannot be read out by MPU.

6.4 COMMAND REGISTER:

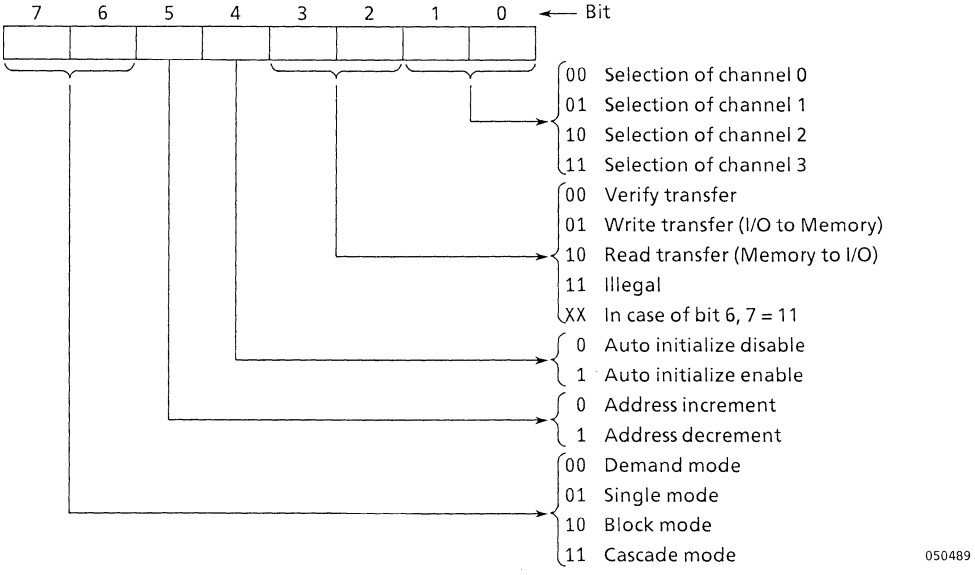
This 8-bit register controls the operation of TMP82C37A. This command register is programmed (clear or reset) by MPU when it is in the program condition. The figure below show the functions of command bits. For address codes, refer to Figure 6.2.



050489

6.5 MODE REGISTER:

All channels have a 6-bit mode register, respectively. This mode register is written by MPU when it is in the program condition, and Bit 0 and 1 select the channel to be programmed.

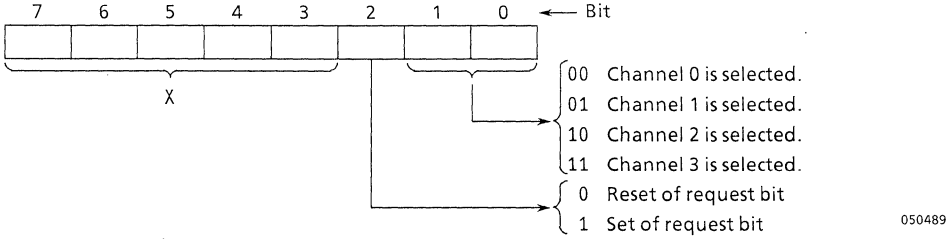


6.6 REQUEST REGISTER:

The TMP82C37A is capable of responding to DMA service request by software similar to DREQ. Each channel has a single bit request register which cannot be masked. Further, priority is given by the priority encode circuit.

Bit of each register is set or cleared by software and further, cleared by generation of TC or external $\overline{EOP}$. All registers are cleared by reset. In order to set or reset bit, a proper form of data word is loaded by software.

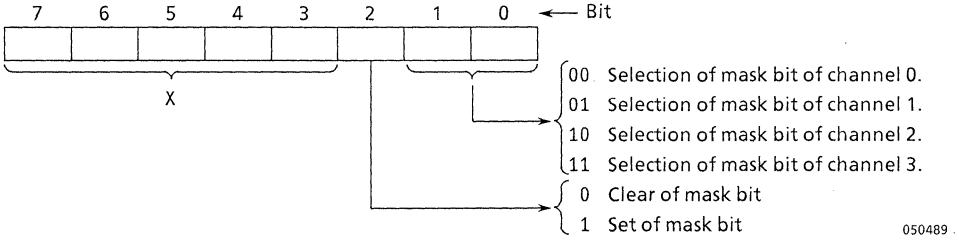
Address codes are shown in Figure 6.2. DMA service request by software is accepted only when the channels are in the block mode. In the Memory-to-Memory transfer, DMA service request only to Channel 0 by this software command becomes valid.



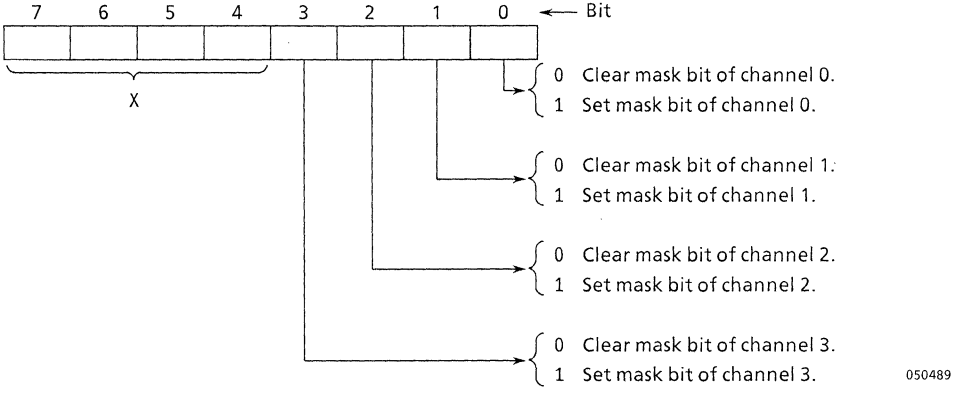
6.7 MASK REGISTER:

For each channel, mask bit are allocated to the mask register to disable DREQ input. If the auto initialization has not been programmed for the channels, the channel corresponding to a mask bit is set when $\overline{EOP}$ is produced. Each bit of the 4-bit mask register is also set or cleared by the software command. All bits are also set by reset. This will disable all DMA requests until the clear mask register command is enabled.

Command addressing is shown in Figure 6.2.



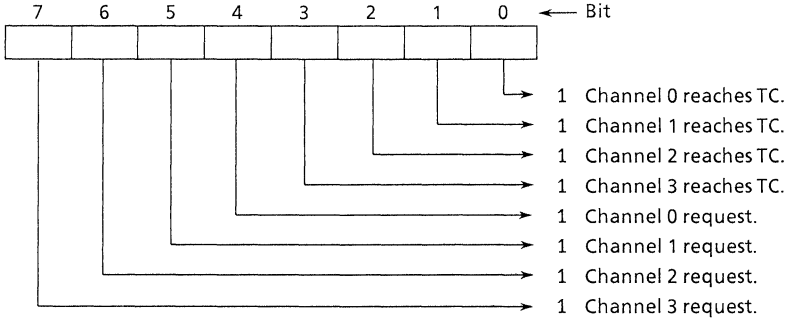
All four bits of the mask register can be written also by a single command.



6.8 STATUS REGISTER:

This register is read out by MPU through the TMP82C37A. Status information of the TMP82C37A at time of readout is included. Information as to which channel reaches the terminal count (TC) and which channel is pending the DMA request are included in this information. Bits 0 to 3 are set every time when a channel reaches TC including the auto initialization.

These bits are cleared by reset or when each status is read out. Bits 4 to 7 are always set when corresponding channels are requesting the DMA service



050489

6.9 TEMPORARY REGISTER:

This register is used for holding data during the Memory-to-Memory transfer. A last word transferred following the end of transfer is read out by MPU that is in the program condition. Unless cleared by reset, this register contains the last word transferred during the preceding Memory-to-Memory transfer.

6.10 SOFTWARE COMMANDS:

These commands are the special software commands which are executed in the program condition and do not depend upon the specified bit pattern on the data bus. These commands are available in following three commands:

6.10.1 Clear First/Last Flip-Flop

This command is executed prior to write or read of address information or word count information of the TMP82C37A. Furthermore, this command is used when low order or high order 8 bits of register are accessed.

6.10.2 Master Clear

This software command has the same effect as the hardware reset. The command, status, request, temporary, and internal First/Last flip-flop registers are all cleared by this command, and the mask register is set.

The TMP82C37A enters into the idle cycle.

6.10.3 Clear Mask Register

This command clears all mask bits of four channels, enabling acceptance of the DMA service requests. Address codes of the software command are shown in Figure 6.2.

Signal						Operation
A_3	A_2	A_1	A_0	$\overline{IOR}$	$\overline{IOW}$	
1	0	0	0	0	1	Read of status register
1	0	0	0	1	0	Write to command register
1	0	0	1	0	1	–
1	0	0	1	1	0	Write to request register
1	0	1	0	0	1	–
1	0	1	0	1	0	Bit set, reset of mask register
1	0	1	1	1	0	–
1	0	1	1	1	0	Write to mode register
1	1	0	0	0	1	–
1	1	0	0	1	0	Clear First/Last fip-flop
1	1	0	1	0	1	Read of temporary register
1	1	0	1	1	0	Master clear
1	1	1	0	0	1	–
1	1	1	0	1	0	Clear mask register
1	1	1	1	0	1	–
1	1	1	1	1	0	All bit write of mask register

050489

Note : The oblique lined codes denote illegal codes.

Figure 6.2 Register and Function Addressing

Channel	Register	Operation	Signal							(*)	Data Bus DB_0~DB_7
			$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A_3	A_2	A_1	A_0		
0	Base & Current Address	Write	0	1	0	0	0	0	0	0	A_0 ~ A_7
			0	1	0	0	0	0	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	0	0	0	0	A_0 ~ A_7
			0	0	1	0	0	0	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	0	0	1	0	W_0 ~ W_7
			0	1	0	0	0	0	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	0	0	1	0	W_0 ~ W_7
			0	0	1	0	0	0	1	1	W_8 ~ W_{15}
1	Base & Current Address	Write	0	1	0	0	0	1	0	0	A_0 ~ A_7
			0	1	0	0	0	1	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	0	1	0	0	A_0 ~ A_7
			0	0	1	0	0	1	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	0	1	1	0	W_0 ~ W_7
			0	1	0	0	0	1	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	0	1	1	0	W_0 ~ W_7
			0	0	1	0	0	1	1	1	W_8 ~ W_{15}
2	Base & Current Address	Write	0	1	0	0	1	0	0	0	A_0 ~ A_7
			0	1	0	0	1	0	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	1	0	0	0	A_0 ~ A_7
			0	0	1	0	1	0	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	1	0	1	0	W_0 ~ W_7
			0	1	0	0	1	0	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	1	0	1	0	W_0 ~ W_7
			0	0	1	0	1	0	1	1	W_8 ~ W_{15}
3	Base & Current Address	Write	0	1	0	0	1	1	0	0	A_0 ~ A_7
			0	1	0	0	1	1	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	1	1	0	0	A_0 ~ A_7
			0	0	1	0	1	1	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	1	1	1	0	W_0 ~ W_7
			0	1	0	0	1	1	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	1	1	1	0	W_0 ~ W_7
			0	0	1	0	1	1	1	1	W_8 ~ W_{15}

050489

(*) : Internal First/Last Flip-Flop

Figure 6.3 Word Count, Address Registers

7. PROGRAMMING

If HLDA of MPU is inactive it is possible to program the TMP82C37A by MPU even when HRQ is active.

However, it is necessary for MPU to take care that programming of the TMP82C37A and anser of HLDA are taken place simultaneously.

It requires care when the DMA service is requested to an unmasked channel during the programming of the TMP82C37A.

It is considered that an embarrassing trouble may be caused in this case.

For instance, if MPU is going to rewrite the address register of channel 2 and in addition, the TMP82C37A is enabled and channel 2 is not masked when channel 2 received a DMA request. The DMA service will be started after one byte of the address register is written. Such a problem as exampled above can be taken place.

To avoid such problems as this, it is better to disable the controller or mask unmasked channels before reprogramming any register.

It is better to enable the controller or clear the masking when the programming is completed.

Example of Program Set (CH2)

```
DI                  : Interrupt disable
OUT   MCLR          : Master clear
MVI   A,XXXXXXXXB
OUT   CMND          : Command register set-up
MVI   A,XXXXXX10B
OUT   MODE          : Mode register set-up
MVI   A,37H
OUT   ADR2          : CH2 Address Reg. (low order)
MVI   A,82H
OUT   ADR2          : CH2 Address Reg. (high order)
MVI   A,17H
OUT   WCNT2         : CH2 Word count register (low order)
MVI   A,95H
OUT   WCNT2         : CH2 Word count register (high order)
MVI   A,00000010B
OUT   MSKB          : CH2 Mask clear (signal bit)
EI                  : Interrupt enable
```

8. ELECTRIC CHARACTERISTICS

8.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEMS	TEST CONDITION	RATING	UNIT
V_{CC}	Supply Voltage	With Respect To GND.	− 0.5 to + 7.0	V
V_{IN}	Input Voltage		− 0.5 to V_{CC} + 0.5	V
V_{OUT}	Output Voltage		− 0.5 to V_{CC} + 0.5	V
PD	Power Dissipation	−	250	mW
T_{sol}	Soder Temperature	−	260 (10sec)	°C
T_{stg}	Strage Temperature	−	− 65 to + 150	°C
T_{opr}	Operating Temperature	−	− 40 to + 85	°C

050489

8.2 DC CHARACTERISTICS

Ta = − 40°C to + 85°C, VCC = 5V ± 10%, VSS (GND) = 0V

SYMBOL	ITEMS	CONDTION	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		− 0.5	−	0.8	V
V_{IH}	Input High Voltage		2.2	−	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 3.2mA	−	−	0.45	V
V_{OH1}	Output High Voltage	I_{OH1} = − 400μA	2.4	−	−	V
V_{OH2}	Output High Voltage	I_{OH2} = − 100μA	V_{CC} − 0.8	−	−	V
I_{IL}	Input Leak Current	$0V \leqq V_{IN} \leqq V_{CC}$	−	−	± 10	μA
I_{OFL}	Output Leak Current	$0.45V \leqq V_{OUT} \leqq V_{CC}$	−	−	± 10	μA
I_{CC1}	Operating Supply Current	CLK = 5MHz $V_{IH} = V_{CC} - 0.2V$ V_{IL} = 0.2V	−	−	10	mA
I_{CC2}	Stand-by Supply Current	CLK = DC $V_{IH} = V_{CC} - 0.2V$ V_{IL} = 0.2V	−	−	10	μA

050489

8.3 AC CHARACTERISTICS

8.3.1 Active Cycle (Notes : 2 and 9)

Ta = −40°C to +85°C, V_{CC} = 5V ± 10%, V_{SS} (GND) = 0V (1/2)

SYMBOL	PARAMETER		MIN.	MAX.	UNIT
T_{AEL}	AEN HIGH from CLK LOW (S1) Delay time		−	200	ns
T_{AET}	AEN LOW from CLK HIGH (SI) Delay time		−	130	ns
T_{AFAB}	ADR Active to Float Delay from CLK HIGH		−	90	ns
T_{AFC}	$\overline{READ}$ or $\overline{WRITE}$ Float Delay from CLK HIGH		−	120	ns
T_{AFDB}	DB Active to Float Delay from CLK HIGH		−	170	ns
T_{AHR}	ADR from $\overline{READ}$ HIGH Hold Time		TCY-100	−	ns
T_{AHS}	DB from ADSTB LOW Hold Time		30	−	ns
T_{AHW}	ADR from $\overline{WRITE}$ HIGH Hold Time		TCY-50	−	ns
T_{AK}	DACK Valid from CLK LOW Delay Time		−	170	ns
	$\overline{EOP}$ HIGH from CLK HIGH Delay Time		−	170	ns
	$\overline{EOP}$ LOW from CLK HIGH Delay Time		−	170	ns
T_{ASM}	ADR Stable from CLK HIGH		−	170	ns
T_{ASS}	DB to ADSTB LOW Setup Time		100	−	ns
T_{CH}	Clock HIGH Level Width		68	−	ns
* T_{CL}	Clock LOW Level Width		100	−	ns
T_{CY}	Clock Cycle Time		200	−	ns
T_{DCL}	CLK HIGH to $\overline{READ}$ or $\overline{WRITE}$ LOW Delay Time	(NOTE 3)	−	190	ns
T_{DCTR}	$\overline{READ}$ HIGH from CLK HIGH (S4) Delay Time	(NOTE 3)	−	190	ns
T_{DCTW}	$\overline{WRITE}$ HIGH from CLK HIGH (S4) Delay Time	(NOTE 3)	−	130	ns
T_{DQ1}	HRQ Valid from CLK HIGH Delay Time	(NOTE 4)	−	120	ns
* T_{DQ2}			−	140	ns
T_{EPS}	$\overline{EOP}$ Low from CLK LOW Setup Time		40	−	ns
T_{EPW}	$\overline{EOP}$ pulse width		220	−	ns
T_{FAAB}	DB Float to Active Dilay from CLK HIGH		−	170	ns
T_{FAC}	$\overline{READ}$ or $\overline{WRITE}$ Active from CLK HIGH		−	150	ns
T_{FADB}	DB Float to Active Dilay from CLK HIGH		−	200	ns
T_{HS}	HLDA Valid to CLK HIGH Setup Time		75	−	ns
T_{IDH}	Input Data from $\overline{MEMR}$ HIGH Hold Time		0	−	ns
T_{IDS}	Input Data to $\overline{MEMR}$ HIGH Setup Time		170	−	ns
T_{ODH}	Output Data from $\overline{MEMW}$ HIGH Hold Time		10	−	ns
T_{ODV}	Output Data Valid to $\overline{MEMW}$ HIGH		125	−	ns
* T_{QS}	DREQ to CLK LOW (SI, S4) Setup Time		30	−	ns
T_{RH}	CLK to READY LOW Hold Time		20	−	ns

050489

Ta = −40°C to +85°C, V_{CC} = 5V ± 10%, V_{SS} (GND) = 0V (2/2)

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
T_{RS}	READY to CLK LOW Setup Time	60	−	ns
T_{STL}	ADSTB HIGH from CLK HIGH Delay Time	−	130	ns
T_{STT}	ADSTB LOW from CLK HIGH Delay Time	−	90	ns

050489

Note 1 : TCL and TDQ2.
The following AC specification can be also guaranteed under the conditions
: Ta = −40°C to 50°C
Vcc = 5V + 5%
Vss = 0V

Note 2 : Value with * is different from AC specification of N-MOS part.

8.3.2 Program Condition (Idle Cycle) (Notes : 2, 8 and 9)

Ta = −40°C to +85°C, V_{CC} = 5V ± 10%, V_{SS} (GND) = 0V

SYMBOL	ITEM	MIN.	MAX.	UNIT
T_{AR}	ADR Valid or $\overline{CS}$ LOW to $\overline{READ}$ LOW	50	−	ns
T_{AW}	ADR Valid or $\overline{WRITE}$ HIGH Setup Time	130	−	ns
T_{CW}	$\overline{CS}$ LOW to $\overline{WRITE}$ HIGH Setup Time	130	−	ns
T_{DW}	Data Valid to $\overline{WRITE}$ HIGH Setup Time	130	−	ns
T_{RA}	ADR or $\overline{CS}$ Hold from $\overline{READ}$ HIGH	0	−	ns
T_{RDE}	Data Access from $\overline{READ}$ LOW (Note 7)	−	140	ns
T_{RDF}	Data Bus Float Delay from $\overline{READ}$ HIGH	0	70	ns
T_{RSTD}	Power Supply HIGH to RESET LOW Setup Time	500	−	ns
T_{RSTS}	RESET to First $\overline{IOWR}$	2	−	TCY
T_{RSTW}	RESET pulse width	300	−	ns
T_{RW}	$\overline{READ}$ pulse width	200	−	ns
T_{WA}	ADR from $\overline{WRITE}$ HIGH Hold Time	20	−	ns
T_{WC}	$\overline{CS}$ HIGH from $\overline{WRITE}$ HIGH Hold Time	20	−	ns
T_{WD}	Data from $\overline{WRITE}$ HIGH Hold Time	30	−	ns
T_{WWS}	$\overline{WRITE}$ pulse width	160	−	ns

050489

Capacity Ta = 25°C, V_{CC} = GND = 0V

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
C_O	Output Capacitance	f_C = 1.0MHz, Input = 0V	−	−	8	pF
C_I	Input Capacitance		−	−	15	
$C_{I/O}$	I/O Capacitance		−	−	20	

050489

Note 1. TYP. value is that when rated voltage is applied at Ta = 25°C.

Note 2. Test conditions; à) Unless otherwise specified, timing defining signal voltage are;
Input High level = 2.4V, Low level = 0.45V
Output High level = 2.2V, Low level = 0.8V
b) Input rising and falling times are below 20ns.
c) Unless otherwise spesified, 1×TTL gate and 150 pF load are provided to output.

Note 3. Normal write pulse width is TCY-100 ns. Extension write pulse width is 2TCY-100 ns. Read pulse width is 2TCY-50 ns, and compressed read pulse width is TCY-50 ns.

Note 4. TDQ is measured at two different high levels.
TDQ1 = 2.2V, TDQ2 = 3.3V.

Note 5. It is necessary to keep DREQ active until DACK is received.

Note 6. Both low active and high active level are available for DREQ and DACK.

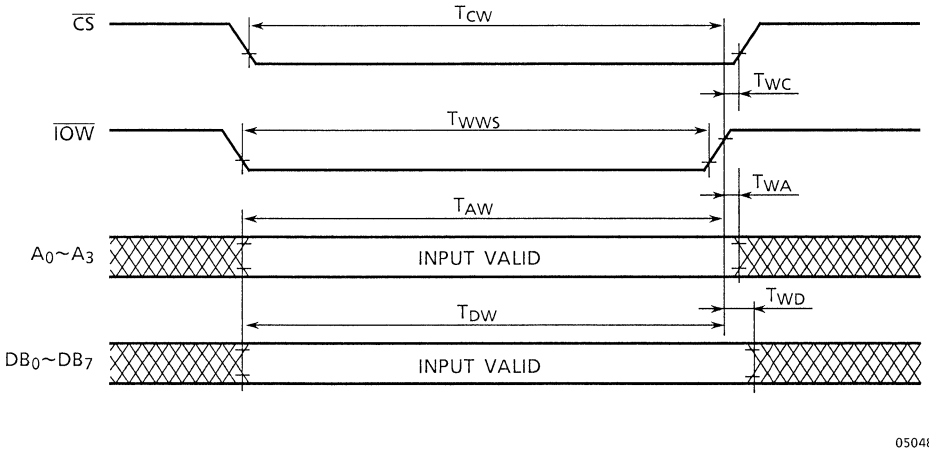
Note 7. Output load of the data bus are provided with 1 ×TTL gate and 15 pf as the minimum value, and 1×TTL gate and 150 pF as the maximum value.

Note 8. Sucessive read or/and write operations by the MPU to program must be timed to allow at least 400ns as recovery time between active read or write pulses.

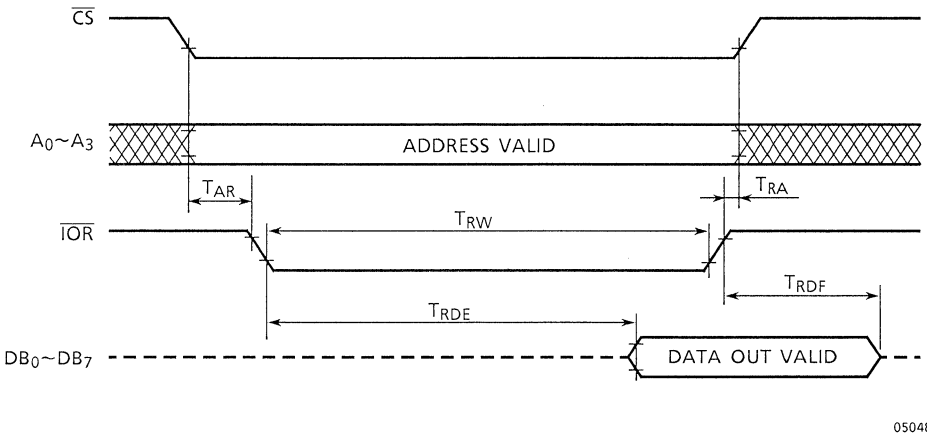
Note 9. Signal $\overline{\text{READ}}$ and $\overline{\text{WRITE}}$ are $\overline{\text{IOR}}$ and $\overline{\text{MEMW}}$ for the DMA operations from peripheral devices to the memory. In the DMA operations from the memory to peripheral devices, they are $\overline{\text{MEMR}}$ and $\overline{\text{IOW}}$.

Note 10. When N state wait is added at time of write to memory in the latter half memory-to-memory transfer, this parameter increases by N (TCY) at a time.

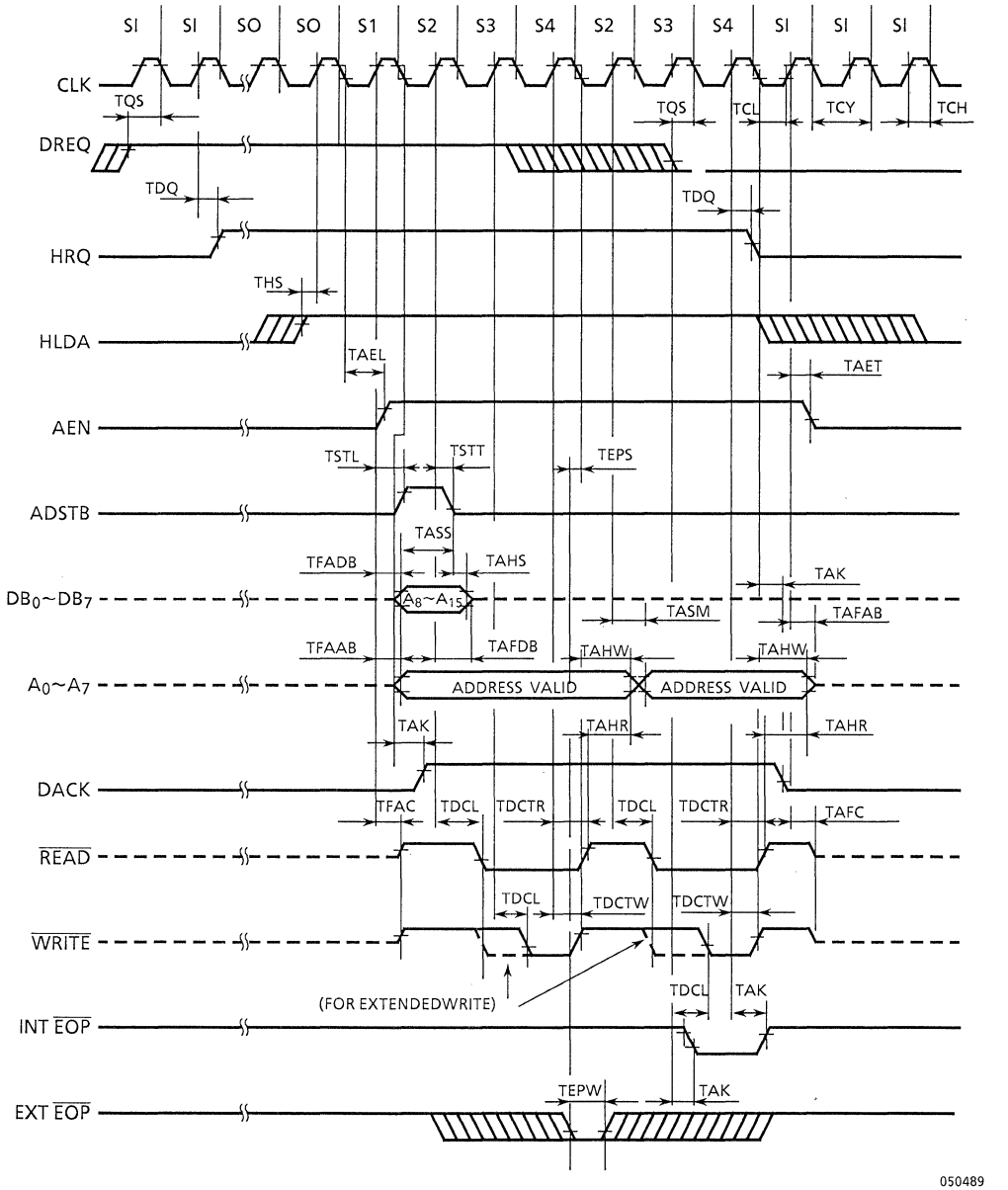
9. TIMING DIAGRAM



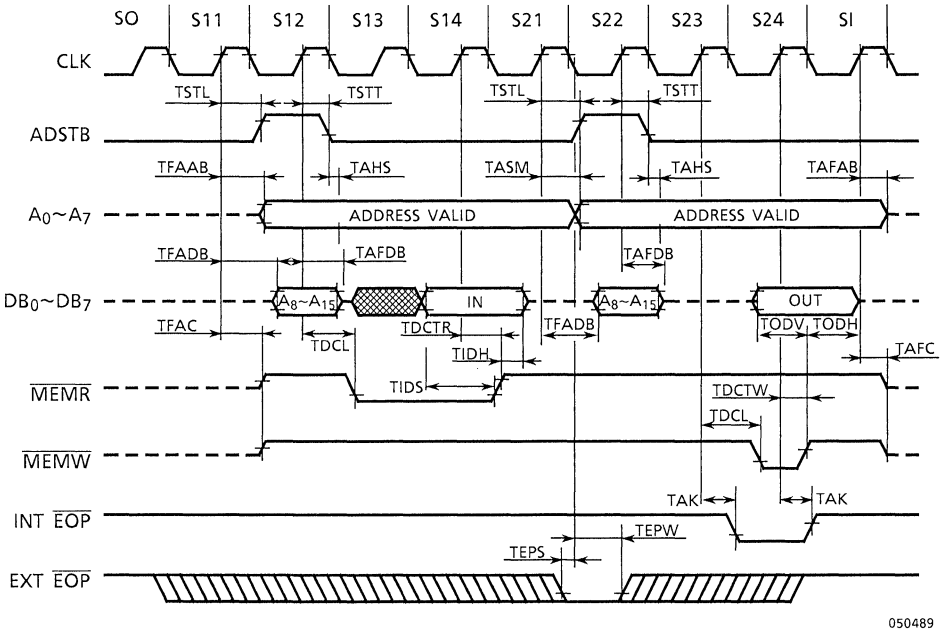
Timing Diagram 1 Program Condition Write Timing



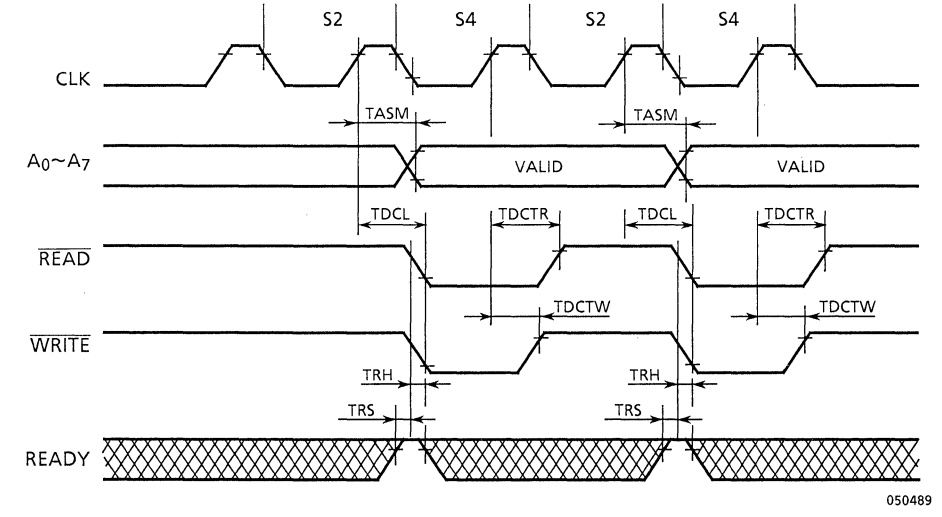
Timing Diagram 2 Program Condition Read Cycle



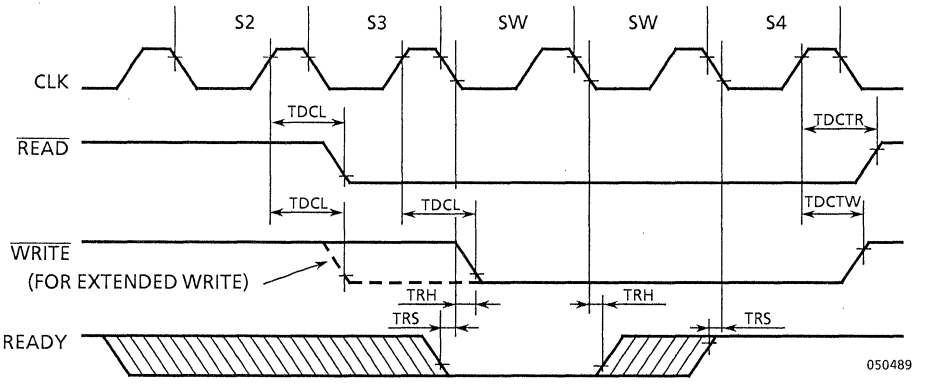
Timing Diagram 3 Active Cycle



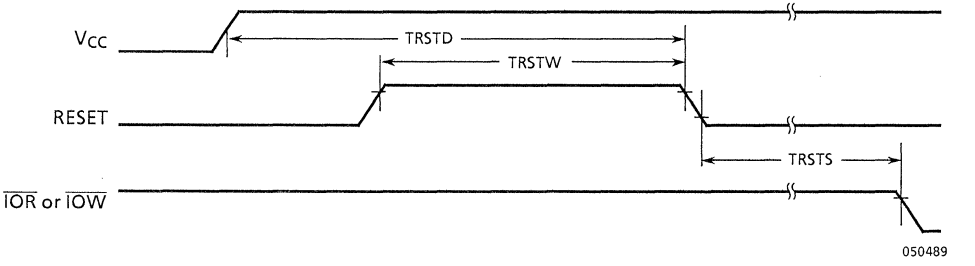
Timing Diagram 4 Memory-to-Memory Transfer



Timing Diagram 5 Compressed Transfer



Timing Diagram 6 Ready Timing



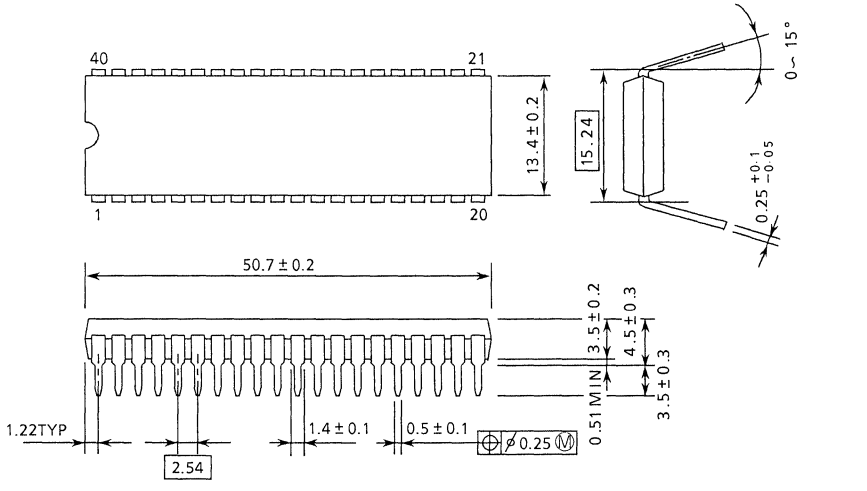
Timing Diagram 7 Reset Timing

10. EXTERNAL DIMENSION

10.1 40PIN DIP EXTERNAL DIMENSION

DIP40-P-600

Unit : mm



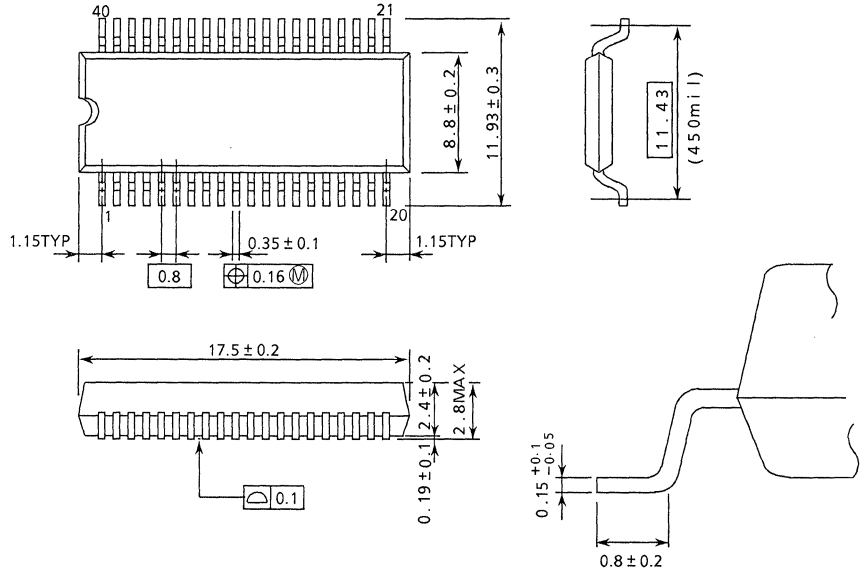
270289

Note : Each pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No. 1 and No. 40 leads.

10.2 40PIN SOP EXTERNAL DEMENSION

SSOP40-P-450

Unit : mm



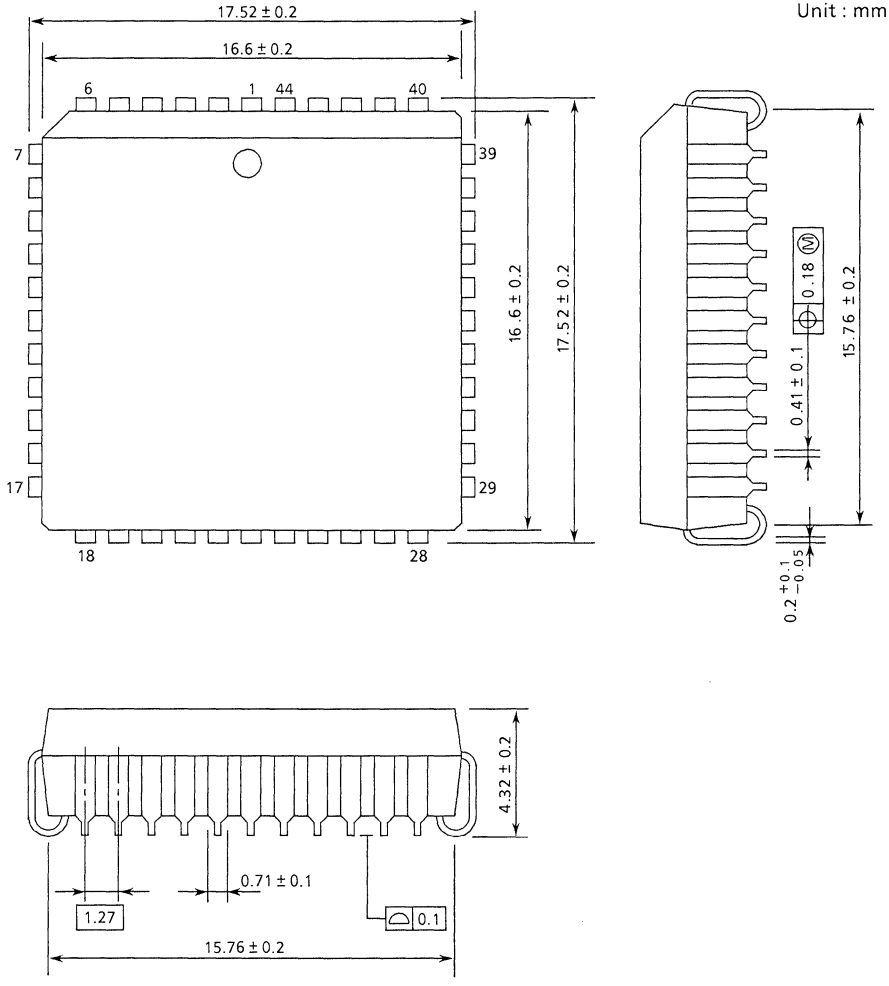
270289

Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

10.3 44PIN PLCC EXTERNAL DEMENSION

QFJ44-P-S650

Unit : mm



270289

11. EXAMPLE OF APPLICATION CIRCUIT

The connecting method of the TMP82C37A and MPU is shown in Figure 11.1.

The multimode DMA controller outputs a hold request whenever valid DMA request is produced from peripheral device. When MPU answers by the hold acknowledge signal, the TMP82C37A receives the control right of the address bus, data bus, and control bus. In the first transfer, address (the least significant 8 bits of the address bits and the most significant 8 bits on the data bus) is output.

The content of the data bus is latched by the 8-bit latch (TC74HC373P) to make the address bus complete. After execution of the first transfer, that latched data is updated only when carry or borrow is produced on the least significant address byte. When one TMP82C37A is used, four DMA channels are provided.

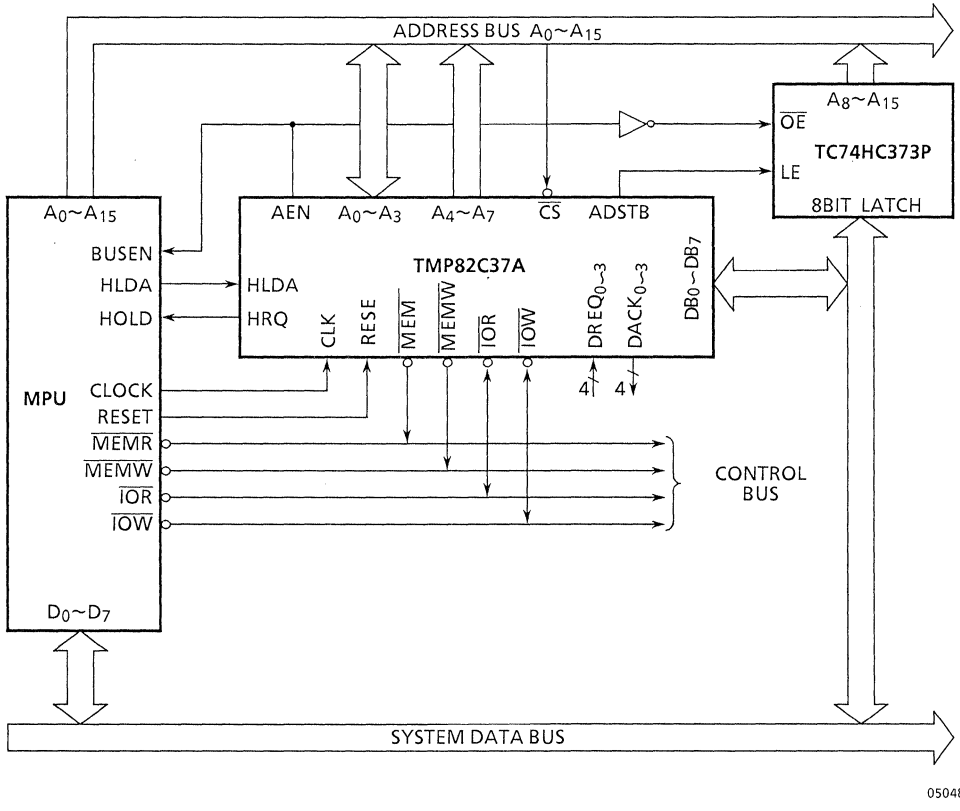


Figure 11.1 Basic System Connection Diagram

Figure 11.2 shows the expansion method for number of DMA channels. It is possible to realize net 7 DMA channels by connecting the second TMP82C37A to one of the DMA channels of the first TMP82C37A.

Two DMA chips commonly use the same 8-bit latch. Thus, any channel is used for expansion.

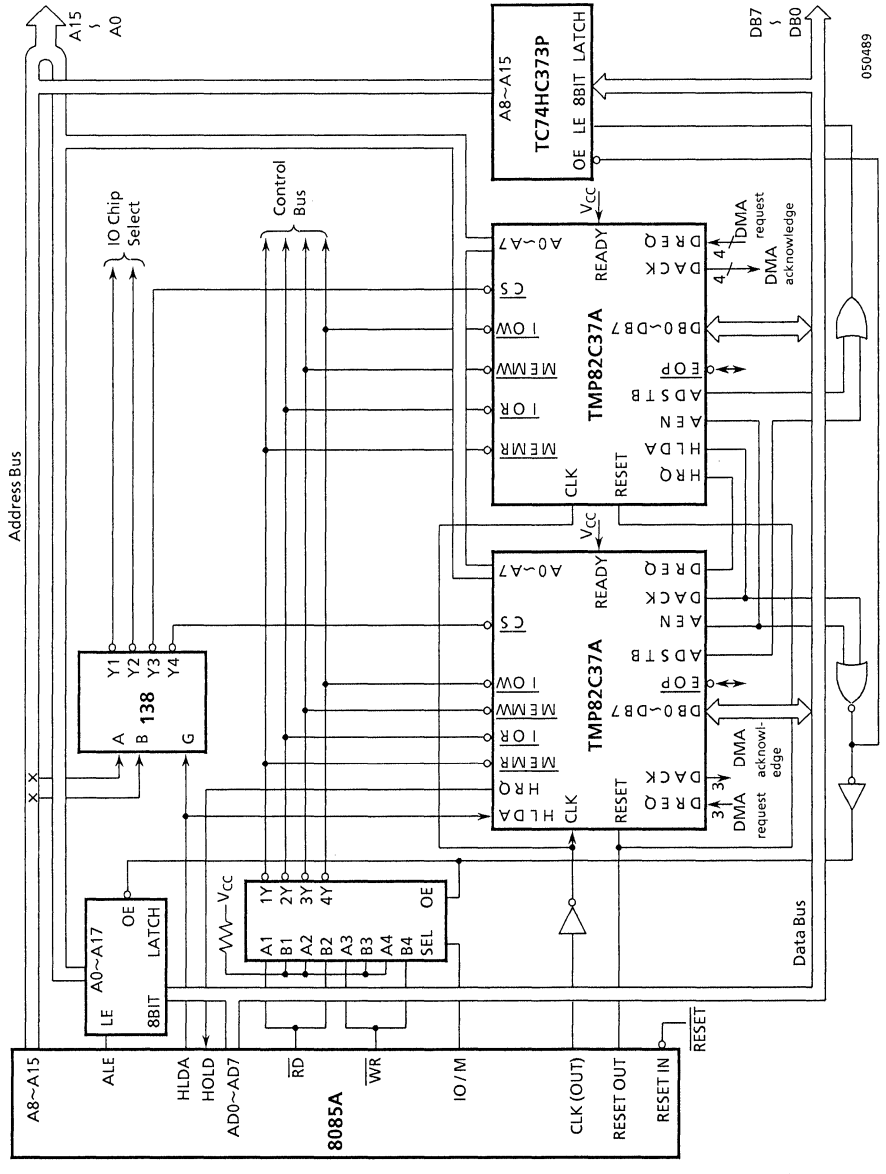


Figure 11.2 Expansion of TMP82C37A

MULTIMODE DMA CONTROLLER
TMP8237AP-5

1. GENERAL DESCRIPTION

The TMP8237AP-5 (hereinafter referred to as TMP8237A) is a multimode direct memory access (DMA) controller. The TMP8237A improves the system function by directly transferring information between the system memory and external devices. Memory-to-Memory data transfer capability is also provided.

The TMP8237A is provided with versatile programmable control functions in order to improve data throughput.

The TMP8237A is used with an 8-bit address register connected externally.
The TMP8237A has four built-in independent channels and it is possible to expand channels through cascade connection.

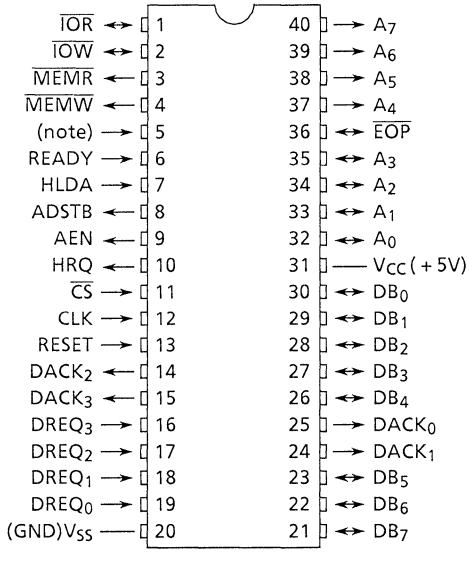
There are three basic data transfer modes which are programmable by the user. Each channel is programmable individually and autoinitialization is possible by End of Process ($\overline{EOP}$) signal.

Each channel has the maximum 64K capability for both address and word count. $\overline{EOP}$ signal is capable of terminating data transfer between DMA and memories. $\overline{EOP}$ signal is useful for block search or verify or for terminating erroneous service.

2. FEATURES

- Four independent DMA channels
- Three transfer modes; block, demand, and single transfer modes
- Independent auto initialize function provided to each of all channels
- Memory-to-Memory transfer
- Address increment or decrement
- All DMA request disabled by disabling the master system
- Individual DMA request enable/disable control
- Unrestricted channel expansion by cascade connection
- End of Process ($\overline{EOP}$) input for terminating transfer
- Software DMA Request
- Polarity control provided for DREQ signal and DACK signal
- Option for increasing transfer speed up to 2.5M word/sec
- Single +5V power supply

3. PIN CONNECTIONS (TOP VIEW)



Note : PIN 5 must be connected to VCC or opened

050489

Figure 3.1 Pin Connections

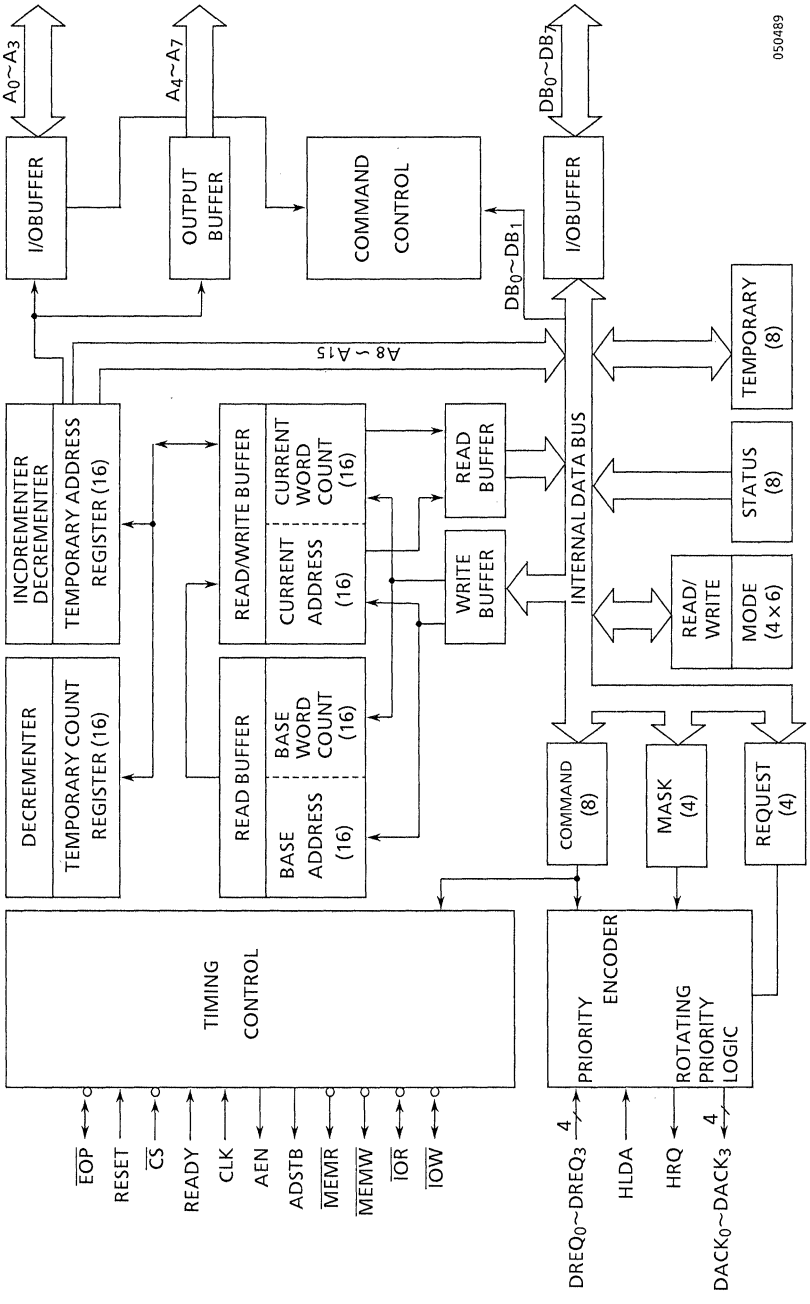


Figure 3.2 Block Diagram of TMP8237A

4. PIN NAME & FUNCTION

- VCC

  +5V power supply

- VSS

  Ground

- CLK (Clock, Input)

  This input controls the internal operation and data transfer rate of the TMP8237A.

- $\overline{CS}$ (Chip Select, Input)

  This input is low active and used to select the TMP8237A as an I/O device during an I/O read or I/O write by the host MPU. If $\overline{IOR}$ or $\overline{IOW}$ is toggled following each transfer when a host MPU and the TMP8237A are transferring data mutually, $\overline{CS}$ may be kept at low.

- RESET (Reset, Input)

  This input is asynchronous input to clear the command, status, request and temporary registers. In addition, this input is used to clear First/Last flip-flops and set the mask register. Following the reset, the TMP8237A is placed in the idle cycle.

- READY (Ready, Input)

  This input is used to extend the memory or I/O read and write pulses in DMA cycle in order to adapt to low speed memories or I/O peripheral devices.

- HLDA (Hold Acknowledge, Input)

  By this signal, the TMP8237A knows that the system bus control is turned over from MPU.

- $DREQ_0$-$DREQ_3$ (DMA Request, Input)

  DMA request signals are input from peripheral circuits. If priority is fixed, the highest priority is given to $DREQ_0$ and the lowest priority to $DREQ_3$. Polarity of DREQ is programmable. DREQ becomes high active by RESET.

- DB_0-DB_7 (Data Bus, Input/Output)

  The data bus are bidirectional 3-state lines connected to the system data bus. During MPU is in I/O read state, output is enabled and contents of the registers (address, status, temporary and word count) are output to MPU. During MPU is in I/O read state, the data bus serves as input and it becomes possible to program the control register of the TMP8237A.

  During the DMA cycle, the most significant 8 bits of address are output on the data bus and latched by ADSTB signal externally. During the Memory-to-Memory transfer, the data of the source memory location are loaded into the temporary register of the TMP8237A by the read operation and the contents of the temporary register are output to the destination memory location by the write operation.

- $\overline{IOR}$ (I/O Read, Input/Output)

  I/O read is a bidirectional, low active and 3-state signal. During the idle cycle, this signal serves as an input control signal used by MPU to read the control registers of the TMP8237A. During the active cycle, this signal serves as an output control signal used by the TMP8237A to access data from the peripheral circuit during the DMA read and transfer.

- $\overline{IOW}$ (I/O Write, Input/Output)

  I/O write is a bidirectional, low active, 3-state signal. During the idle cycle, this signal serves as an input control signal used by MPU to load the information to the TMP8237A. During the active cycle, this signal served as an output control signal used by TMP8237A to load the data to the peripheral. For write to the TMP8237A by MPU, the leading edge of the write signal ($\overline{IOW}$) is required for every data transfer. It is not possible to write more than two data by toggling $\overline{CS}$ while holding the $\overline{IOW}$ pin at low level.

- $\overline{EOP}$ (End of Process, Input/Ouptut)

  $\overline{EOP}$ (End of Process) is a signal relative to the end of DMA service, and is a low active, bidirectional and open drain signal. When the channel word count reaches from 0000H to FFFFH, the TMP8237A output low pulse of $\overline{EOP}$ to peripheral devices as the end signal.

  In addition, it is also possible to pull $\overline{EOP}$ to the low level by peripheral device in order to cause the end of process.

  When $\overline{EOP}$ is received (internally or externally), the channel which is presently active terminates the service, sets that TC bit of the status register and resets that request bit.

  If that channel is programmed for auto initialization, that current register is updated from the base register. In all other cases, mask bit is set and the content of that register remains unchanged.

During the Memory-to-Memory transfer, $\overline{EOP}$ is output when TC of channel 1 is produced. $\overline{EOP}$ is always used for channels with active DACK and external $\overline{EOP}$ has no connection when $DACK_0$-$DACK_3$ are all inactive.

$\overline{EOP}$ is an open drain signal and therefore, requires an external pull-up resistor.

- A_0-A_3 (Address, Input/Output)

  The four least significant address lines are the bidirectional 3-state signals. In the idle cycle, these lines serve as the input signals and used by MPU for write/read of the control register. In the active cycle, they serve as the output signals and become low order 4 bits of output address.

- A_4-A_7 (Address, Output)

  The four most significant address lines are 3-state output signals.
  These lines are enabled for the period of DMA service only.

- HRQ (Hold Request, Output)

  This is the hold request signal to MPU, and is used to request the system bus control. HRQ is output by the TMP8237A according to a software request or unmasked DREQ.

- $DACK_0$-$DACK_3$ (DMA Acknowledge, Output)

  The DMA acknowledge lines indicate that channels are active. Normally, these are used for selecting peripheral devices.

  Only one DACK becomes active but it does not become active unless DMA is controlling the system bus. Polarity of these lines are programmable. After reset, they initialize low active.

- AEN (Address Enable, Output)

  Address Enable is a high active signal and used to enable output of the external latch which holds high order byte of address and to disable the system bus during the DMA cycle.

  During the DMA transfer, HLDA and AEN are used to disable all I/O except programmed I/O. The TMP8237A disables $\overline{CS}$ input for DMA transfer to prevent itself from being selected automatically.

- ADSTB (Address Strobe, Output)

  This signal is a strobe output to an external latch circuit and is used to latch high order 8-bit address from DB_0-DB_7.

- $\overline{\text{MEMR}}$ (Memory Read, Output)

  This is a low active 3-state output used for transferring data from a memory to a peripheral device or for data accessing from a selected memory during the Memory-to-Memory transfer.

- $\overline{\text{MEMW}}$ (Memory Write, Output)

  This is a low active 3-state output used for transferring data from a peripheral device to a memory or for writing data into a selected memory during the Memory-to-Memory transfer.

5. OPERATIONAL DESCRIPTION

5.1 DMA OPERATION

The TMP8237A has two operations; idle cycle and active cycle. Each of these cycles consists of several states.

On the TMP8237A, it is possible to consider 7 states each of which consists of one clock cycle. State I (SI) is an idle state. This is such a state as there is no valid DMA request pending. SI is a program condition state which is programmable by MPU.

State 0 (S0) is the first DMA service state. This is a state that the TMP8237A made a hold request to MPU but not yet received the acknowledge signal from MPU. When the acknowledge signal is received from MPU, the transfer is started.

S1, S2, S3 and S4 are the DMA service states. If more time is required by the transfer, it is possible to insert the wait state (SW) before S4 by READY input to the TMP8237A.

In the Memory-to-Memory transfer, in order to assure complete transfer, read from the memory and write to the memory are required. 8 states are necessary for one transfer. The first four status (S11, S12, S13 and S14) are read from the memory and the latter four state (S21, S22, S23 and S24) are write to the memory.

The temporary data register is used as an intermediate storage area of memory bytes.

5.2 IDLE CYCLE

When DMA service is not requested by channels, the TMP8237A enters into the idle cycle and is placed in SI state. In order to check if the channels request DMA service, the TMP8237A samples DREQ for every clock.

The TMP8237A also samples $\overline{CS}$ to check if MPU is requesting read or write of internal registers. When $\overline{CS}$ is low and HLDA is also low, the TMP8237A is placed in the program condition.

At this time, MPU is able to change or check the content of any internal register through read or write from that register.

Address lines A_0-A_3 are input signals and used for selecting a register being read or written. $\overline{IOR}$ and $\overline{IOW}$ are used for selecting read or write and decide read/write timing.

The internal flip-flop is used for generating address extension bits according to number and size of internal registers. (First/Last flip-flop) This bit is used for deciding high or low order byte of 16-bit address and word count register.

The flip-flop is reset by the master clear or reset. In addition, this flip-flop also can be reset by an idependent software command.

On a special software command, the execution in the TMP8237A program condition is possible. These commands are decoded as in the address setting when both $\overline{CS}$ and $\overline{IOW}$ are active.

The data bus is not used for this command. This command is available in three types; clear First/Last flip-flop , master clear and clear mask register.

5.3 ACTIVE CYCLE

When the TMP8237A is in the idle cycle and the channels are requesting DMA service, the TMP8237A outputs HRQ to MPU and goes into the active cycle. In this cycle, the DMA service for any one of 4 modes is executed.

5.3.1 Single Transfer Mode:

In this mode, the TMP8237A performs a single byte transfer during each HRQ/HLDA handshake. When DREQ becomes active, HRQ becomes active.

After MPU responds by driving HLDA active, a single byte transfer will take place. After the transfer HRQ becomes inactive, its word count is decreased, and address is increased or decreased. When word count changes from 0000H to FFFFH, a terminal signal is generated and if the channels are programmed, the auto initialization is made.

To execute the single byte transfer, it is necessary to hold DREQ until DACK corresponding each DREQ becomes active. If DREQ is continuously active, HRQ becomes inactive following each transfer and then, becomes active again, and the new single byte is executed following the leading edge of HLDA.

On the 8085A system, one machine cycle can be executed during the DMA transfer.

5.3.2 Block Transfer Mode:

In this mode the TMP8237A continues the transfer until terminal count (TC) is generated or an external End of Process signal ($\overline{EOP}$) is encountered. Here, TC is produced when the word count changes from 0000H to FFFFH.

What is requied for DREQ is to hold it in active state until DACK becomes active. Auto initialization (if so programmed) is taken place at the end of DMA service.

5.3.3 Demand Transfer Mode:

In this mode the TMP8237A continues the transfer until TC is produced or $\overline{EOP}$ is or DREQ becomes inactive. Thus, it is possible for a device, which is requesting the DMA service, to suspend the transfer by making DREQ inactive. The service is resumed when DREQ is made active again. It is possible to read an intermediate value of address and word count from the current address and current word count register of the TMP8237A while the system bus is returned to MPU during execution of the DMA service.

The auto initialization is taken place following TC or $\overline{EOP}$ at the end of DMA service. In order to perform a new DMA service following the auto initialization, the active edge of DREQ is necessary.

5.3.4 Cascade Mode:

This mode is used when the TMP8237A is cascade connected for a simple system expansion. HRQ and HLDA of the additional TMP8237A are connected to DREQ and DACK of the first TMP8237A. DMA request to the TMP8237A which is added for the purpose of system expansion is authorized by the priority circuit of the first TMP8237A.

If the priority is already decided, the additional device must wait till the acknowledge request. The cascade channel of the first TMP8237A is used only for deciding priority of the additional TMP8237A and therefore, the channel itself does not output address nor control signal. This is to prevent the added device from colliding with output of the cascade channel. On the TMP8237A, DACK respond to DREQ.

However all other outputs except HRQ are disabled.

The state of cascade connection is shown in Figure 5.1. In Figure 5.1, two levels of DMA are formed. To further expand the TMP8237A, it is possible to add it to the second level using the remaining channel of the first TMP8237A. To further add another TMP8237A, the third level can be formed by cascade connecting it to the second level.

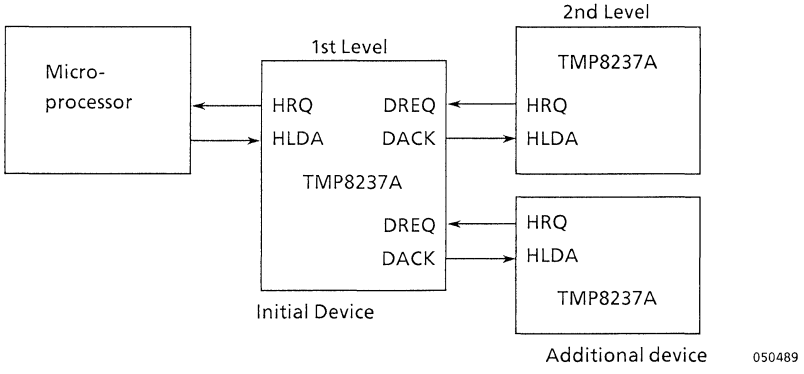


Figure 5.1 Example of Cascade Connection of TMP8237A

5.4 TRANSFER FORMAT

3 different transfer format are available for 3 active transfer modes.

They are read, write and verify. In the write transfer, data is transferred from I/O device to memory by $\overline{\text{MEMW}}$ and $\overline{\text{IOR}}$. In the read transfer, data is transferred from memory to I/O device by $\overline{\text{MEMR}}$ and $\overline{\text{IOW}}$.

The verify transfer is a pseudo transfer. The TMP82C37A perform such operations as address generation for read or write transfer, answer to $\overline{\text{EOP}}$, etc. However, memory or I/O control line does not become active.

5.5 MEMORY-TO-MEMORY TRANSFER:

The TMP8237A has the ability of block movement and is capable of transferring data block from one memory address location to another location. When Bit 0 of the command register is programmed at Logic 1, Channel 0 and 1 operate as the Memory-to-Memory transfer channels.

Channel 0 serves as a source address and Channel 1 as a destination address, and the word count of Channel 1 is used. The Memory-to-Memory transfer is executed when software DMA request is set for Channel 0.

The Memory-to-Memory transfer must use the block transfer mode.

When Channel 0 is programmed as a fixed source address, it is possible to write single source words into a memory block.

When the TMP8237A is programmed for the Memory-to-Memory transfer, Channel 0 and Channel 1 must be masked. The same value as that is set for Channel 1 must be set for the word count of Channel 0. During the Memory-to-Memory transfer, AEN became active but DACK does not become active.

During the Memory-to-Memory trnasfer, the TMP8237A respond to external $\overline{EOP}$ signal. In the block search, the data comparator uses this ($\overline{EOP}$) input to terminate the DMA service when match is found. The Memory-to-Memory transfer timing is shown in Timing Diagram 4.

5.6 AUTO INITIALIZATION:

When Bit 4 of the mode register is set to 1, the channels are set up for the auto initialization. During the auto initialization, data are loaded into the current address and current word count registers from the base address and base word count registers, respectively, following $\overline{EOP}$. The base registers are loaded by MPU simultaneously with the current registers and remain unchanged during the DMA service.

When the channels are under the auto initialization, mask bit is not set by $\overline{EOP}$.

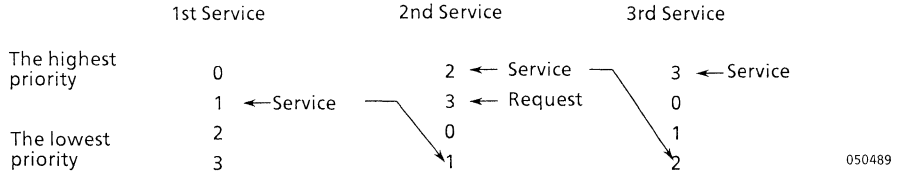
Following the auto initialization, that channel is prepared to execute the service without interposition of MPU.

5.7 PRIORITY:

The TMP8237A has two types of priority which can be selected by software. The first type is the fixed priority. Channel priority is fixed by channel number. The lowest priority is channel 3, followed by 2, 1, and the highest priority is channel 0.

The second type is the rotating priority. In this type, an accepted channels is then given with the lowest priority .

On the rotating priority in the single chip DMA system, the highest priority of any one channel comes after no more than three higher priority services have occurred. This rotating priority prevent a specfic channel from occupying the system all the time.(See the following next page diagram.)



The priority judging circuit selects a channel with the highest priority requesting the DMA service for every active edge of HLDA

Once the channel starts the service, that operation will not be suspended even when the service is demanded by another channel with higher priority. A channel with higher priority can get the control right only after a channel with lower priority relinquished HRQ.

Whenever the control is transferred from a channel to another channel, MPU gets the system bus control right. This assures the leading edge of HLDA which is used for selecting a channel with the highest priority.

5.8 COMPRESSED TRANSFER TIMING:

In order to accomplish greater throughout allowed by system characteristics, the TMP8237A is capable of compressing the transfer time to 2 clock cycles. As can be seen from Timing Diagram 3, State S3 is used to extend readout pulse access time. When State S3 is removed, readout pulse width becomes equal to write pulse width. Then, the transfer will consist of State S2 for changing address and State S4 for executing read/write. State S1 is produced when A_8 to A_{15} are updated (refer to Address Generation). Compressed transfer timing is shown in Timing Diagram 5.

During Memory-to Memory transfer, compressed transfer is not available.

5.9 ADDRESS GENERATION:

To reduce number of pins, the TMP8237A has the multiplexed address/data bus. State S1 is used to output high order address byte to the external latch. The trailing edge of ADSTB is used to load the address byte from the data line on the external latch circuit.

AEN is used to enable latch outputs from 3 states. Low order address byte is directly output by the TMP8237A.

A_0 to A_7 are connected to address bus. Timing Diagram 3 show the relationship among CLK, AEN, ADSTB, DB_0 to DB_7 and A_0 to A_7.

Addresses produced during the block and demand transfers are sequential. For many transfer the same address data will be held in the external address latch. This address data changes only when carry or borrow from A_7 to A_8 is produced in the normal sequence. To raise system through put, on the TMP8237A, S1 state is executed only for updating A_8 to A_{15} requiring the external latch.

6. DESCRIPTION OF REGISTERS

Register Name	Size	Number
Base address register	16-bit	4
Base word count register	16-bit	4
Count address register	16-bit	4
Current word count register	16-bit	4
Temporary address register	16-bit	1
Temporary word count register	16-bit	1
Status register	8-bit	1
Command register	8-bit	1
Temporary register	8-bit	1
Mode register	6-bit	4
Mask register	4-bit	1
Request register	4-bit	1

050489

Figure 6.1 Internal Registers

6.1 CURRENT ADDRESS REGISTER:

Each channel has a 16-bit current address register. This register holds addreses that are used during the DMA trnasfer. After each transfer, this register is automatically incremented or decremented, and intermediate address values are stored in the current address register during the transfer. Write or read of this register is made by MPU. An original value is initialized again by the auto initialization.

The auto initialization is taken place only after $\overline{EOP}$.

6.2 CURRENT WORD COUNT REGISTER:

Each channel has a 16-bit current word count register. For this register, the number of words to be transferred that is one less than that to be transferred must be programmed. The word counter is decremented after each transfer. Intermediate values of word count are stored in this register during the transfer. When the register value goes from 0000H to FFFFH, TC (Terminal Count) is produced.

When this register is in the program condition, load or read is made by MPU. Following the end of DMA service, this register is initialized to original values again by the auto initialization.

The auto initialization is taken place only when $\overline{EOP}$ is produced.

Note that the content of the word count register becomes FFFFH following internally produced $\overline{EOP}$.

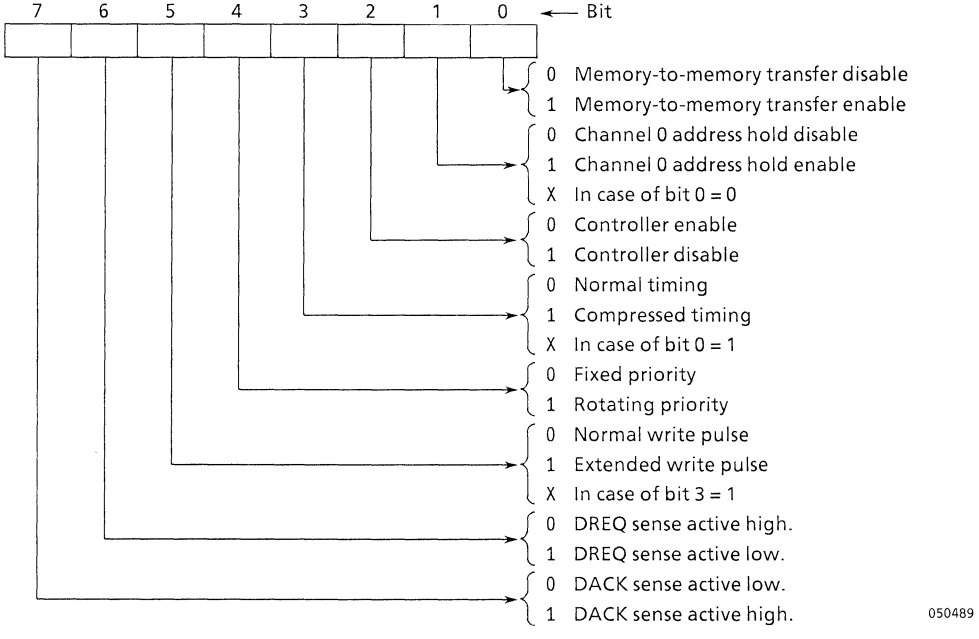
6.3 BASE ADDRESS REGISTER, BASE WORD COUNT REGISTER:

Each channel has a pair of registers; the base address register and base word count register. These 16-bit registers store original values of related current registers. These registers are used to store original values of current registers at time of the auto initialization. Write to the base register is made at the same time of write into equivalent current registers during the programming by MPU.

Therefore, write into the current registers which store intermediate values are made over these intermediate values. The base register cannot be read out by MPU.

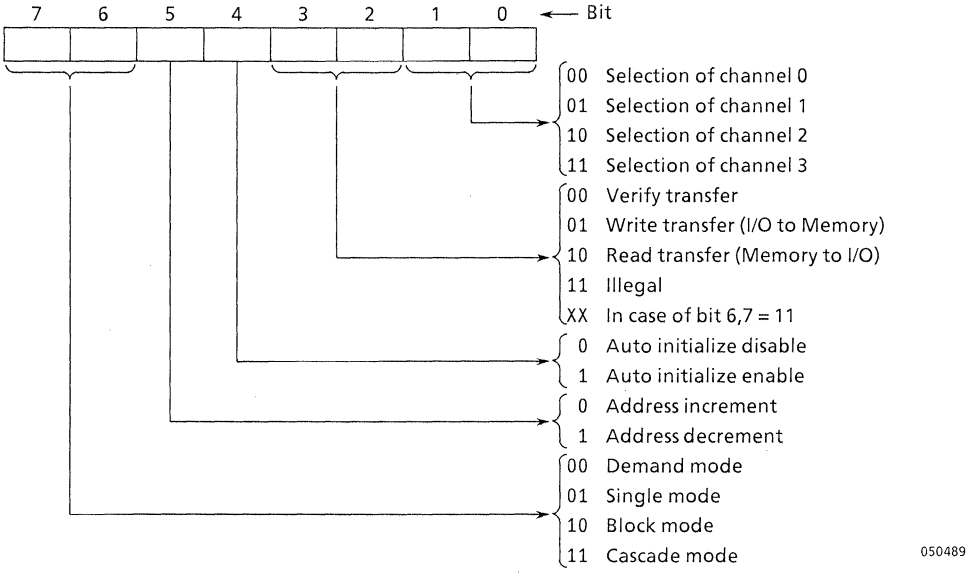
6.4 COMMAND REGISTER:

This 8-bit register controls the operation of TMP8237A. This command register is programmed (clear or reset) by MPU when it is in the program condition. The figure presented below show the functions of command bits. For address codes, refer to Figure 6.2.



6.5 MODE REGISTER:

All channels have a 6-bit mode register, respectively. This mode register is written by MPU when it is in the program condition, and Bit 0 and 1 select the channel to be programmed is to be written.

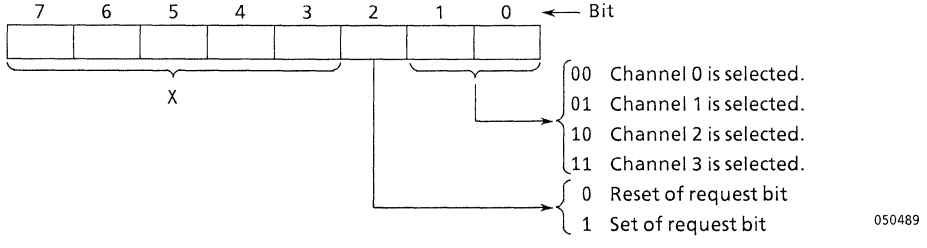


6.6 REQUEST REGISTER:

The TMP8237A is capable of responding to DMA service request by software similar to DREQ. Each channel has a single bit request register which cannot be masked. Further, priority is given by the priority encode circuit.

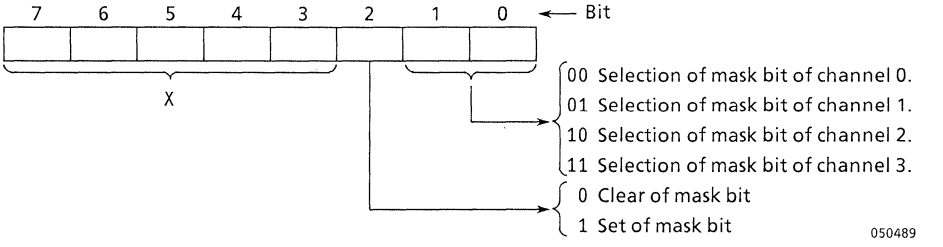
Bit of each register is set or cleared by software and further, cleared by generation of TC or external $\overline{EOP}$. All registers are cleared by reset. In order to set or reset bit, a proper form of data word is loaded by software.

Address codes are shown in Figure 6.2. DMA service request by software is accepted only when the channels are in the block mode. In the Memory-to-Memory transfer, DMA service request to only channel 0 by this software command becomes valid.

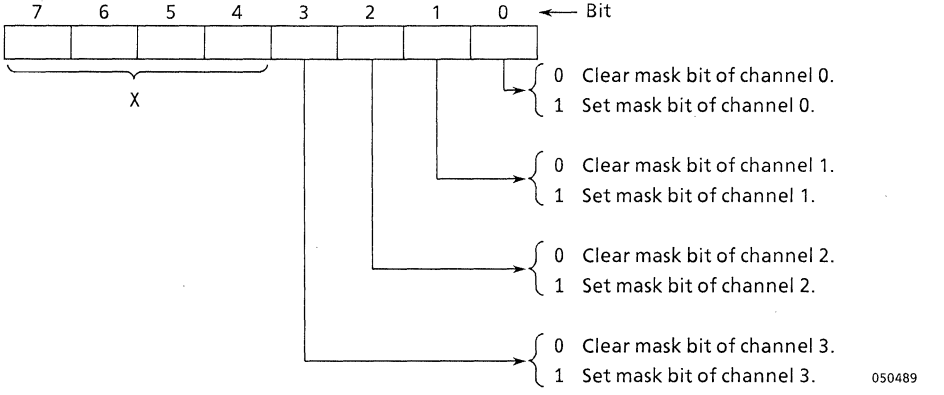


6.7 MASK REGISTER:

For each channel, mask bit are allocated to the mask register to disable DREQ input. If the auto initialization has not been programmed for the channels, the channel corresponding to a mask bit is set when $\overline{EOP}$ is produced. Each bit of the 4-bit mask register is also set or cleared by the software command. All bits are also set by reset. This will disable all DMA requests until the clear mask register command is enabled. Command addressing is shown in Figure 6.2.



All four bits of the mask register can be written also by a single command.



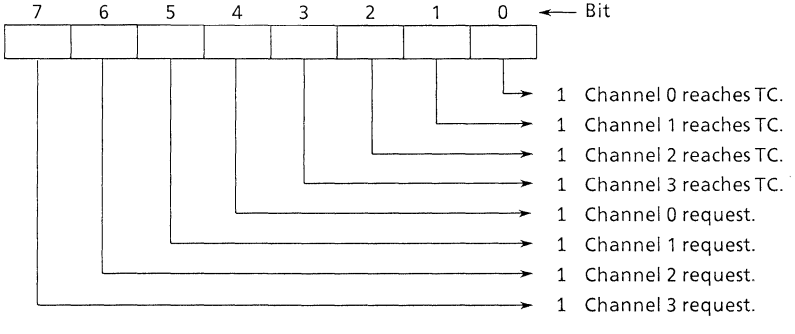
6.8 STATUS REGISTER

This register is read out by MPU through the TMP8237A. Status information of the TMP8237A at time of readout is included.

Information as to which channel reaches the terminal count (TC) and which channel is pending the DMA request are included in this information. Bits 0 to 3 are set every time when a channel reaches TC including the auto initialization.

These bits are cleared by reset or when each status is read out.

Bits 4 to 7 are always set when corresponding channels are requesting the DMA service



050489

6.9 TEMPORARY REGISTER:

This register is used for holding data during the Memory-to-Memory transfer. A last word transferred following the end of transfer is read out by MPU that is in the program condition. Unless cleared by reset, this register contains the last word transferred during the preceding Memory-to-Memory transfer.

6.10 SOFTWARE COMMANDS:

These commands are special software commands which are executed in the program condition and do not depend upon the specified bit pattern on the data bus. These commands are available in following third commands:

6.10.1 Clear First/Last flip-flop

This command is executed prior to write or read of address information or word count information of the TMP8237A. Furthermore, this command is used when low order or high order 8 bits of register are accessed.

6.10.2 Master Clear

This software command has the same effect as the hardware reset. The command, status, request, temporary, and internal First/Last flip-flop registers are all cleared by this command, and the mask register is set.

The TMP8237A enters into the idle cycle.

6.10.3 Clear Mask Register

This command clears all mask bits of four channels, enabling acceptance of the DMA service requests.

Address codes of the software commands are shown in Figure 6.2.

Signal						Operation
A_3	A_2	A_1	A_0	$\overline{IOR}$	$\overline{IOW}$	
1	0	0	0	0	1	Read of status register
1	0	0	0	1	0	Write to command register
1	0	0	1	0	1	–
1	0	0	1	1	0	Write to request register
1	0	1	0	0	1	–
1	0	1	0	1	0	Bit set, reset of mask register
1	0	1	1	0	1	–
1	0	1	1	1	0	Write to mode register
1	1	0	0	0	1	–
1	1	0	0	1	0	Clear First/Last flip-flop
1	1	0	1	0	1	Read of temporary register
1	1	0	1	1	0	Master clear
1	1	1	0	0	1	–
1	1	1	0	1	0	Clear mask register
1	1	1	1	0	1	–
1	1	1	1	1	0	All bit write of mask register

Note : The oblique lined codes denote illegal codes. 050489

Figure 6.2 Register and Function Addressing

Channel	Register	Operation	Signal							Internal F/L F/F	Data Bus DB_0~DB_7
			$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A_3	A_2	A_1	A_0		
0	Base & Current Address	Write	0	1	0	0	0	0	0	0	A_0 ~ A_7
			0	1	0	0	0	0	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	0	0	0	0	A_0 ~ A_7
			0	0	1	0	0	0	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	0	0	1	0	W_0 ~ W_7
			0	1	0	0	0	0	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	0	0	1	0	W_0 ~ W_7
			0	0	1	0	0	0	1	1	W_8 ~ W_{15}
1	Base & Current Address	Write	0	1	0	0	0	1	0	0	A_0 ~ A_7
			0	1	0	0	0	1	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	0	1	0	0	A_0 ~ A_7
			0	0	1	0	0	1	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	0	1	1	0	W_0 ~ W_7
			0	1	0	0	0	1	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	0	1	1	0	W_0 ~ W_7
			0	0	1	0	0	1	1	1	W_8 ~ W_{15}
2	Base & Current Address	Write	0	1	0	0	1	0	0	0	A_0 ~ A_7
			0	1	0	0	1	0	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	1	0	0	0	A_0 ~ A_7
			0	0	1	0	1	0	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	1	0	1	0	W_0 ~ W_7
			0	1	0	0	1	0	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	1	0	1	0	W_0 ~ W_7
			0	0	1	0	1	0	1	1	W_8 ~ W_{15}
3	Base & Current Address	Write	0	1	0	0	1	1	0	0	A_0 ~ A_7
			0	1	0	0	1	1	0	1	A_8 ~ A_{15}
	Current Address	Read	0	0	1	0	1	1	0	0	A_0 ~ A_7
			0	0	1	0	1	1	0	1	A_8 ~ A_{15}
	Base & Current Address	Write	0	1	0	0	1	1	1	0	W_0 ~ W_7
			0	1	0	0	1	1	1	1	W_8 ~ W_{15}
	Current Address	Read	0	0	1	0	1	1	1	0	W_0 ~ W_7
			0	0	1	0	1	1	1	1	W_8 ~ W_{15}

050489

Figure 6.3 Word Count, Address Registers

7. PROGRAMMING

If HLDA of MPU is inactive it is possible to program the TMP8237A by MPU even when HRQ is active.

However, it is necessary for MPU to take care that programming of the TMP8237A and answer of HLDA are taken place simultaneously.

It requires care when the DMA service is requested to an unmasked channel during the programming of the TMP8237A.

It is considered that an embarrassing trouble may be caused in this case.

For instance, if MPU is going to rewrite the address register of channel 2 and in addition, the TMP8237A is enabled and channel 2 is not masked when channel 2 received a DMA request. The DMA service will be started after one byte of the address register is written. Such a problem as exampled above can be taken place.

To avoid such problems as this, it is better to disable the controller or mask unmasked channels before reprogramming any register.

It is better to enable the controller or clear the masking when the programming is completed.

Example of Program Set (CH2)

```
DI                          :  Interrupt disable
OUT   MCLR                  :  Master clear
MVI   A, XXXXXXXXB
OUT   CMND                  :  Command register set-up
MVI   A, XXXXXX10B
OUT   MODE                  :  Mode register set-up
MVI   A, 37H
OUT   ADR2                  :  CH2 Address Reg. (low order)
MVI   A, 82H
OUT   ADR2                  :  CH2 Address Reg. (high order)
MVI   A, 17H
OUT   WCNT2                 :  CH2 Word count register (low order)
MVI   A, 95H
OUT   WCNT2                 :  CH2 Word count register (high order)
MVI   A, 00000010B
OUT   MSKB2                 :  CH2 Mask clear (single bit)
EI                          :  Interrupt enable
```

8. ELECTRIC CHARACTERISTICS

8.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	PARAMETER	TEST CONDITIONS	RATING	UNIT
V_{CC}	Supply Voltage	With Respect to GND.	− 0.5 ~ + 7.0	V
V_{IN}	Input Voltage		− 0.5 ~ + 7.0	V
V_{OUT}	Output Voltage		− 0.5 ~ + 7.0	V
P_D	Power Dissipation	−	1.5	W
T_{sol}	Solder Temperature	−	260 (10 sec)	°C
T_{stg}	Strage Temperature	−	− 65 ~ + 150	°C
T_{opr}	Operating Temperature	−	0 ~ + 70	°C

050489

8.2 DC CHARACTERISTICS

Ta = 0°C to 70°C, V_{CC} = 5V ± 5%, V_{SS} (GND) = 0V

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{IL}	Input Low Voltage		− 0.5	−	0.8	V
V_{IH}	Input High Voltage		2.2	−	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 3.2mA	−	−	0.45	V
V_{OH1}	Output High Voltage	I_{OH1} = − 200μA	2.4	−	−	V
V_{OH2}	Output High Voltage	I_{OH2} = − 100μA (HRQ ONLY)	3.3	−	−	V
I_{IL}	Input Leakage Current	$0V \leqq V_{IN} \leqq V_{CC}$	−	−	± 10	μA
I_{OFL}	Output Leakage Current	$0.45V \leqq V_{OUT} \leqq V_{CC}$	−	−	± 10	μA
I_{CC}	Supply Current		−	−	150	mA

050489

8.3 AC CHARACTERISTICS

8.3.1 Active Cycle (1/3) (Notes : 2 and 9)

Ta = 0°C to 70°C, V_{CC} = 5V ± 5%, V_{SS} = 0V

SYMBOL	PARAMETER	MIN.	MAX.	Unit
T_{AEL}	AEN HIGH from CLK LOW (S1) Delay Time	−	200	ns
T_{AET}	AEN LOW from CLK HIGH (SI) Delay Time	−	130	ns
T_{AFAB}	ADR Active to FLoat Delay from CLK HIGH	−	90	ns
T_{AFC}	READ or $\overline{\text{WRITE}}$ Float Delay from CLK HIGH	−	120	ns

050489

Active Cycle (2/3)

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
T_{AFDB}	DB Active to Float Delay from CLK HIGH	—	170	ns
T_{AHR}	ADR from $\overline{READ}$ HIGH Hold Time	TCY – 100	—	ns
T_{AHS}	DB from ADSTB LOW Hold Time	30	—	ns
T_{AHW}	ADR from $\overline{WRITE}$ HIGH Hold Time	TCY – 50	—	ns
T_{AK}	DACK Valid from CLK LOW Delay Time	—	170	ns
	$\overline{EOP}$ HIGH from CLK HIGH Delay Time	—	170	ns
	$\overline{EOP}$ LOW to CLK HIGH Delay Time	—	170	ns
T_{ASM}	ADR Stable from CLK HIGH	—	170	ns
T_{ASS}	DB to ADSTB LOW Setup Time	100	—	ns
T_{CH}	Clock HIGH Level Width	80	—	ns
T_{CL}	Clock LOW Level Width	68	—	ns
T_{CY}	Clock Cycle Time	200	—	ns
T_{DCL}	CLK HIGH to $\overline{READ}$ or $\overline{WRITE}$ LOW Delay Time (Note 3)	—	190	ns
T_{DCTR}	$\overline{READ}$ HIGH from CLK HIGH (S4) Delay Time (Note 3)	—	190	ns
T_{DCTW}	$\overline{WRITE}$ HIGH from CLK HIGH (S4) Delay (Note 3)	—	130	ns
T_{DQ1}	HRQ Valid from CLK HIGH Delay TIme (Note 4)	—	120	ns
T_{DQ2}		—	120	ns
T_{EPS}	$\overline{EOP}$ Low from CLK LOW Setup Time	40	—	ns
T_{EPW}	$\overline{EOP}$ pulse width	220	—	ns
T_{FAAB}	DB Float to Active Delay from CLK HIGH	—	170	ns
T_{FAC}	$\overline{READ}$ or $\overline{WRITE}$ Active from CLK HIGH	—	150	ns
T_{FADB}	DB Float to Active Delay from CLK HIGH	—	200	ns
T_{HS}	HLDA Valid to CLK HIGH Setup Time	75	—	ns
T_{IDH}	Input Data from $\overline{MEMR}$ HIGH Hold Time	0	—	ns
T_{IDS}	Input Data to $\overline{MEMR}$ HIGH Setup Time	170	—	ns
T_{ODH}	Output Data to $\overline{MEMR}$ HIGH Hold Time	10	—	ns
T_{ODV}	Output Data Valid to $\overline{MEMW}$ HIGH	125	—	ns
T_{QS}	DREQ to CLK LOW (SI, S4) Setup Time	0	—	ns
T_{RH}	CLK to READY LOW Hold Time	20	—	ns

050489

Active Cycle (3/3)

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
T_{RS}	READY to CLK LOW Setup Time	60	—	ns
T_{STL}	ADSTB HIGH from CLK HIGH Delay TIme	—	130	ns
T_{STT}	ADSTB LOW from CLK HIGH Delay TIme	—	90	ns

050489

8.3.2 Program Condition (Idle Cycle) (Notes : 2, 8 and 9)

Ta = 0°C to 70°C, V_{CC} = 5V ± 5%, V_{SS} = 0V

SYMBOL	PARAMETER	MIN.	MAX.	UNIT
T_{AR}	ADR Valid or $\overline{CS}$ LOW to $\overline{READ}$ LOW	50	—	ns
T_{AW}	ADR Valid or $\overline{WRITE}$ HIGH Setup Time	130	—	ns
T_{CW}	$\overline{CS}$ LOW to $\overline{WRITE}$ HIGH Setup Time	130	—	ns
T_{DW}	Data Valid to $\overline{WRITE}$ HIGH Setup Time	130	—	ns
T_{RA}	ADR or CS Hold from $\overline{READ}$ HIGH	0	—	ns
T_{RDE}	Data Access from $\overline{READ}$ LOW	—	140	ns
T_{RDF}	Data Bus Float Delay from $\overline{READ}$ HIGH	0	70	ns
T_{RSTD}	Power Supply HIGH to RESET LOW Setup Time	500	—	ns
T_{RSTS}	RESET to First $\overline{IOWR}$	2	—	TCY
T_{RSTW}	RESET pulse width	300	—	ns
T_{RW}	$\overline{READ}$ pulse width	200	—	ns
T_{WA}	ADR from $\overline{WRITE}$ HIGH Hold Time	20	—	ns
T_{WC}	$\overline{CS}$ HIGH from $\overline{WRITE}$ HIGH Hold Time	20	—	ns
T_{WD}	Data from $\overline{WRITE}$ HIGH Hold Time	30	—	ns
T_{WWS}	$\overline{WRITE}$ pulse width	160	—	ns

050489

8.4 CAPACITY

Ta = 25°C, V_{CC} = GND = 0V

SYMBOL	PARAMETER	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
C_O	Output Capacitance	f_C = 1.0MHz, Input = 0V	—	—	8	pF
C_I	Input Capacitance		—	—	15	
$C_{I/O}$	I/O Capacitance		—	—	20	

050489

Note 1 : Typ. value is that when rated voltage is applied at Ta = 25°C.

Note 2 : Test conditions; a) Unless otherwise specified, timing defining signal voltages are;
Input High level = 2.4V, Low level = 0.45V
Output High level = 2.0V, Low level = 0.8V

b) Unless otherwise specified, 1 × TTL gate and 150pF load are provided to output.

Note 3 : Normal write pulse width is TCY − 100 ns. Extension write pulse width is 2TCY − 100 ns. Read pulse width is 2TCY-50 ns, and compressed read pulse width is TCY − 50 ns.

Note 4 : TDQ is measured at two different high levels.
TDQ1 = 2.0V, TDQ2 = 3.3V

Note 5 : It is necessary to keep DREQ active until DACK is received.

Note 6 : Both low active and high active level ae available for DREQ and DACK.

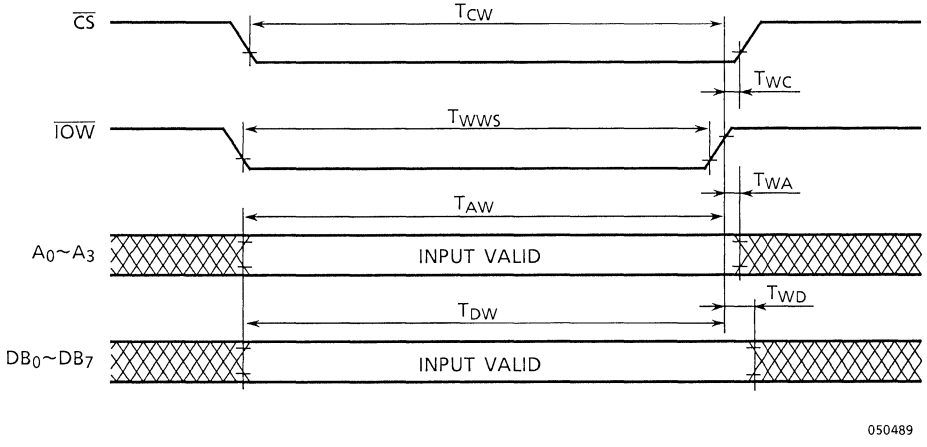
Note 7 : Output load of the data bus are provided with 1 × TTL gate and 15 pF as the minimum value, and 1 × TTL gate and 150 pF as the maximum value.

Note 8 : Sucessive read or/and write operations by the MPU to program must be timed to allow at least 600ns for the TMP8237AP and at least 400ns for the TMP8237AP-5 as recovery time between active read or write pulses.

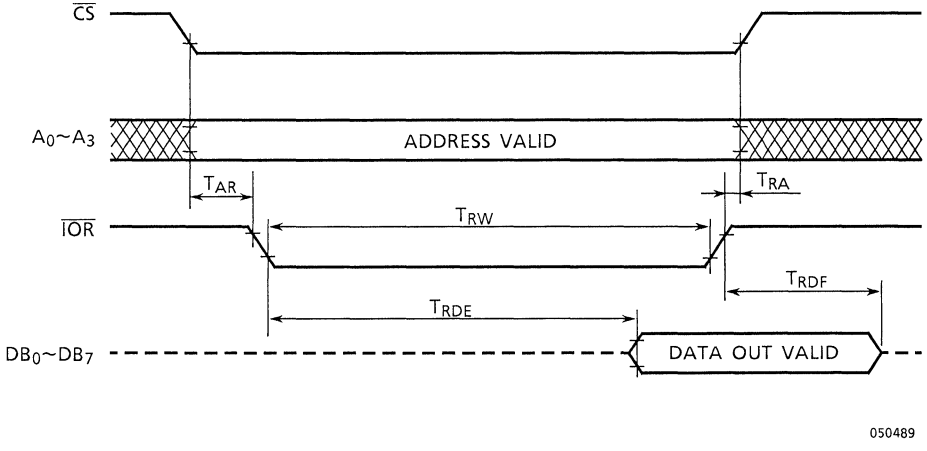
Note 9 : Signal $\overline{\text{READ}}$ and $\overline{\text{WRITE}}$ are $\overline{\text{IOR}}$ and $\overline{\text{MEMW}}$ for the DMA operations from peripheral devices to the memory. In the DMA operations from the memory to peripheral devices, they are $\overline{\text{MEMR}}$ and $\overline{\text{IOW}}$.

Note10 : When N state wait is added at time of write to memory in the latter half of memory-to-memory transfer, this parameter increases by N (TCY) at a time.

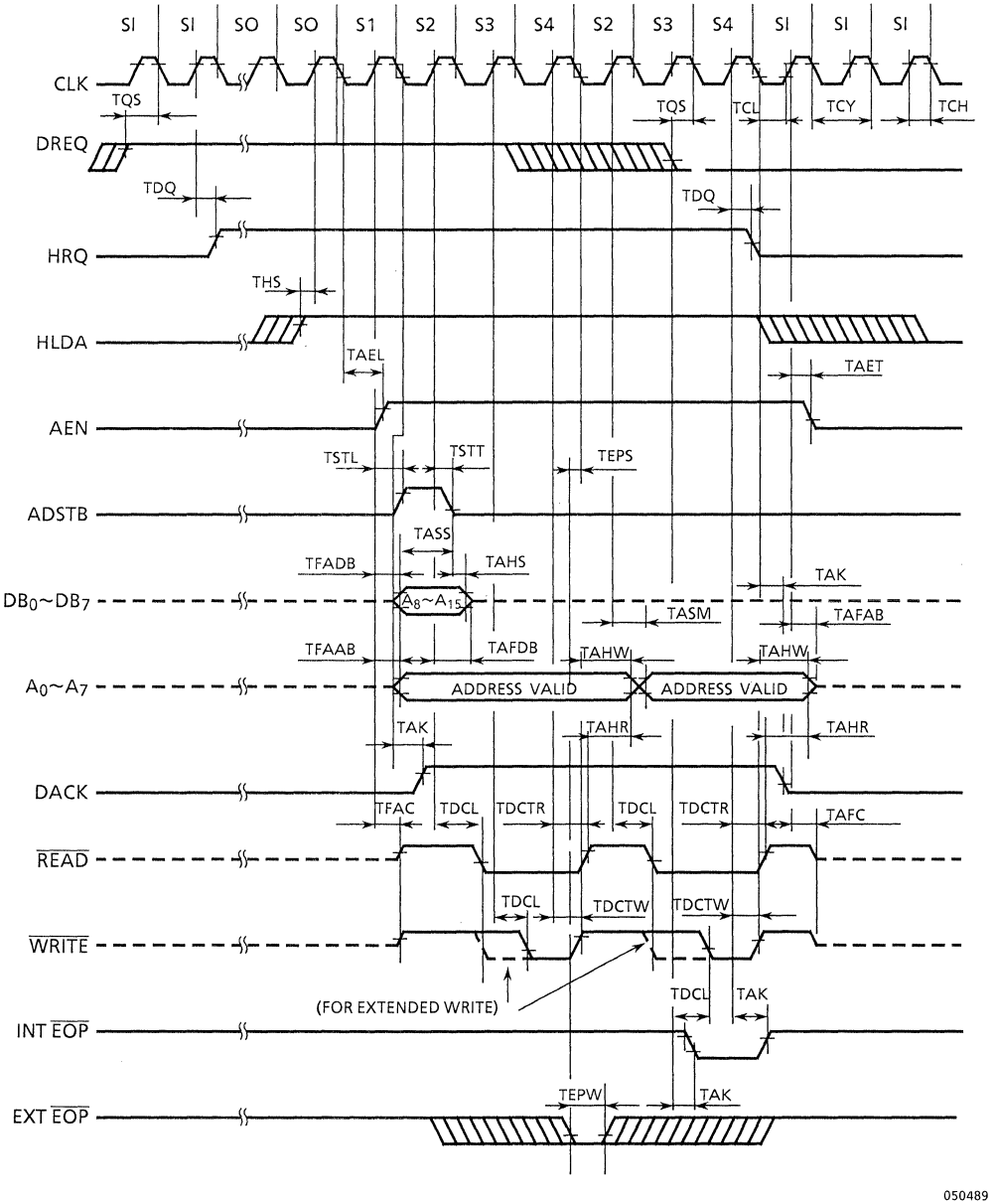
9. TIMING DIAGRAM



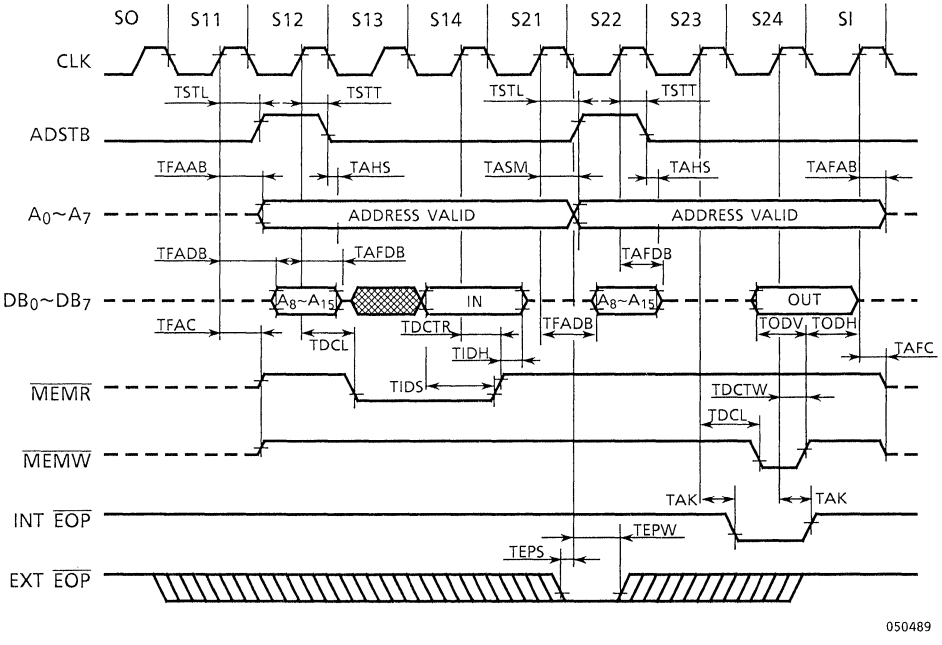
Timing Diagram 1 Program Condition Write Timing



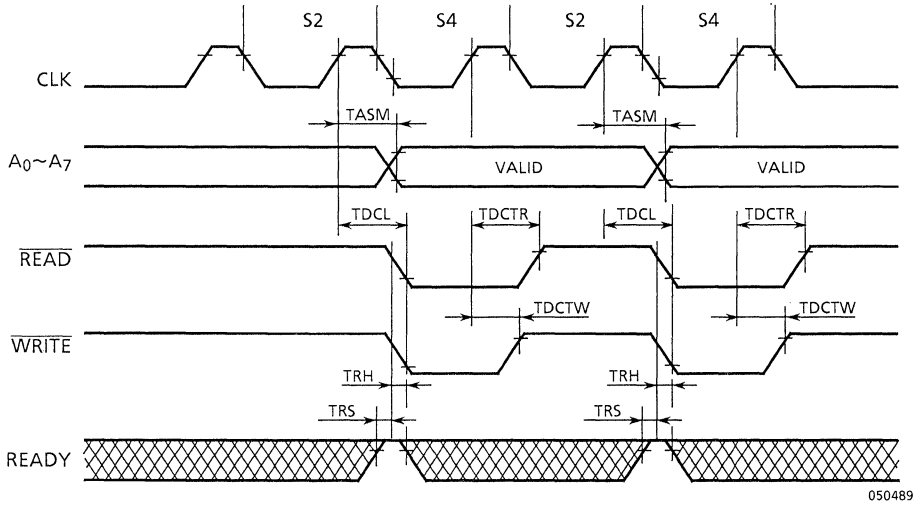
Timing Diagram 2 Program Condition Read Cycle



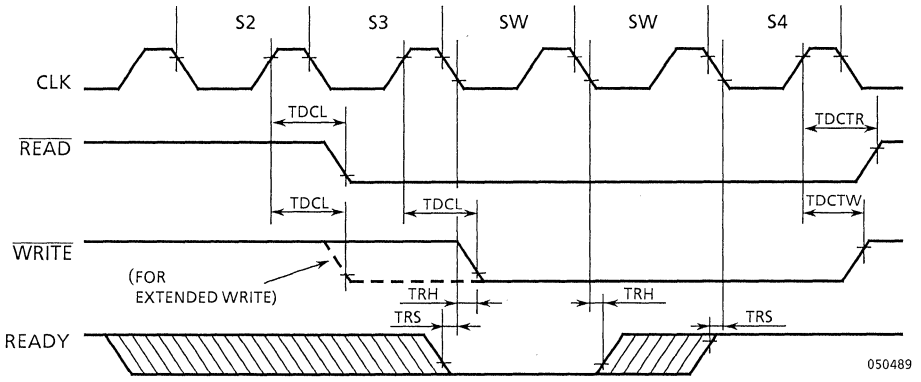
Timing Diagram 3 Active Cycle



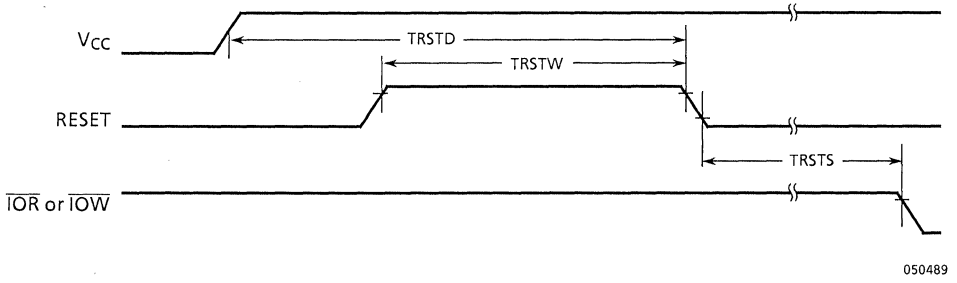
Timing Diagram 4 Memory-to-Memory Transfer



Timing Diagram 5 Compressed Transfer



Timing Diagram 6 Ready Timing

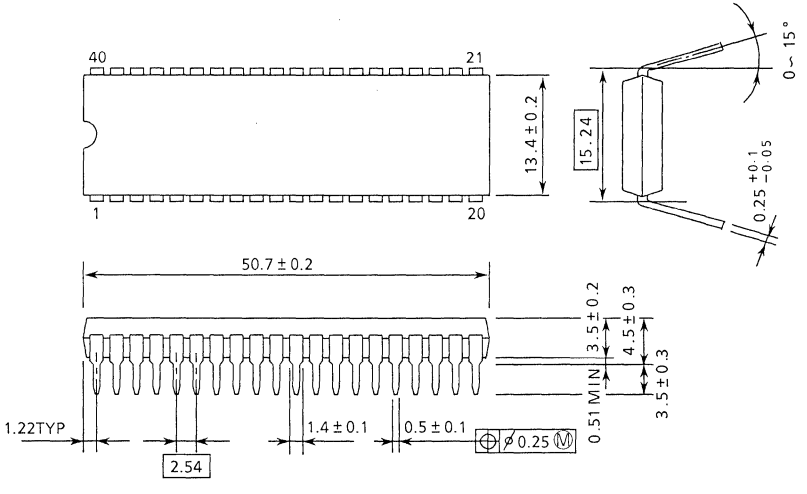


Timing Diagram 7 Reset Timing

10. EXTERNAL DIMENSION VIEW (PLASTIC PACKAGE)

DIP40-P-600

Unit : mm



270289

Note : Each pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoretical positions with respect to No.1 and No.40 leads.

11. EXAMPLE OF APPLICATION CIRCUIT

The connecting method of the TMP8237A and MPU is shown in Figure 11.1.

The multimode DMA controller outputs a hold request whenever valid DMA request is produced from peripheral device. When MPU answers by the hold acknowledge signal, the TMP8237A receives the contorl right of the address bus, data bus, and control bus. In the first transfer, address (the least significant 8 bits of the address bits and the most significant 8 bits on the data bus) is output.

The content of the data bus is latched by the 8-bit latch (TC74HC373P) to make the address bus complete. After execution of the first transfer, that latched data is updated only when carry or borrow is produced on the least significant address byte.

When one TMP8237A is used, four DMA channels are provided.

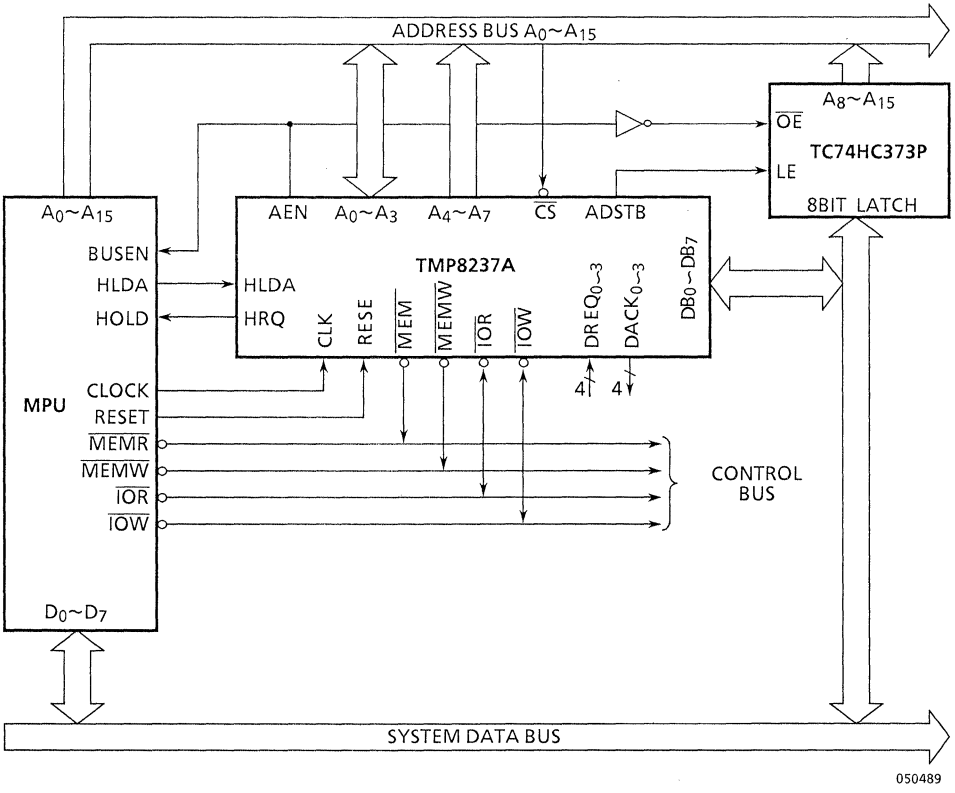


Figure 11.1 Basic System Connection Diagram

Figure 11.2 shows the expansion method for number of DMA channels. It is possible to realize net 7 DMA channels by connecting the second TMP8237A to one of the DMA channels of the first TMP8237A.

Two DMA chips commonly use the same 8-bit latch. Thus, any channel is used for expansion.

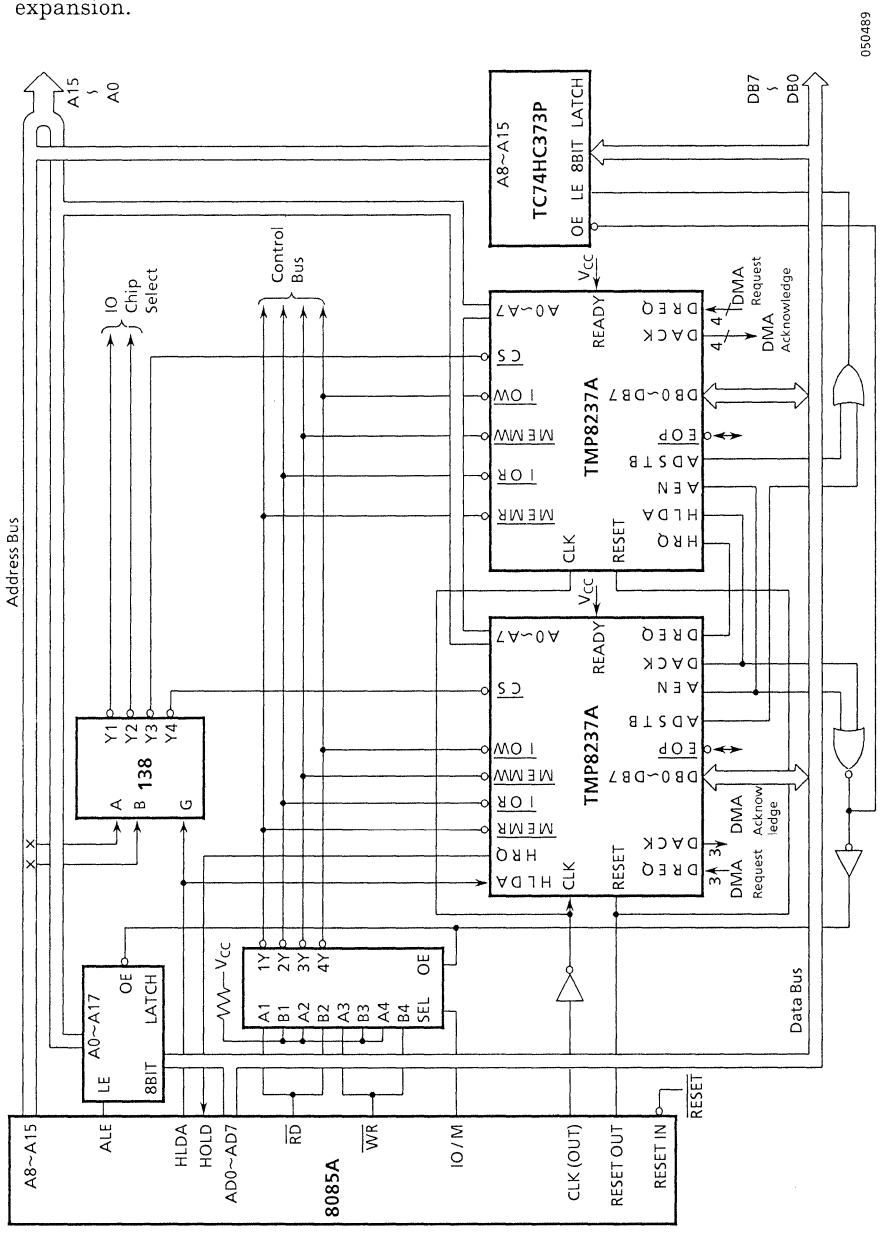


Figure 11.2 Expansion of TMP8237A

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

TMP82C79P-2/TMP82C79M-2

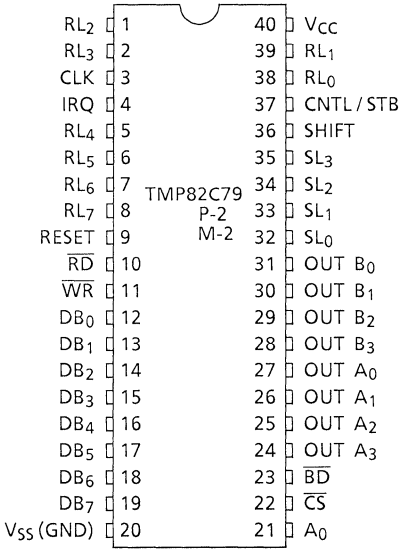
1. GENERAL DESCRIPTION

The TMP82C79P-2/M-2 (hereinafter referred to as TMP82C79) is a programmable keyboard/display interface. The keyboard portion can provide a scanned interface up to 64-contact key matrix. Also, the keyboard portion can interface to an array of sensors or a strobed interface keyboard. Key depressions can be 2-key lockout or N-key rollover. The display portion has 16×8 bits display RAM which can be treated as dual 16×4 bits. Both right entry and left entry display formats are possible.

2. FEATURES

- Simultaneous Keyboard Display operation.
- Scanned Keyboard mode.
- Scanned Sensor matrix mode.
- Strobed Input Entry mode.
- Built-in 8-Character FIFO or 64 bit Sensor RAM.
- Programmable 2-key Lockout or N-key Rollover with contact debounce.
- Built-in 16×8bit Display RAM
- Programmable scan timing.
- Extend operating temperature range −40 °C to +85 °C.

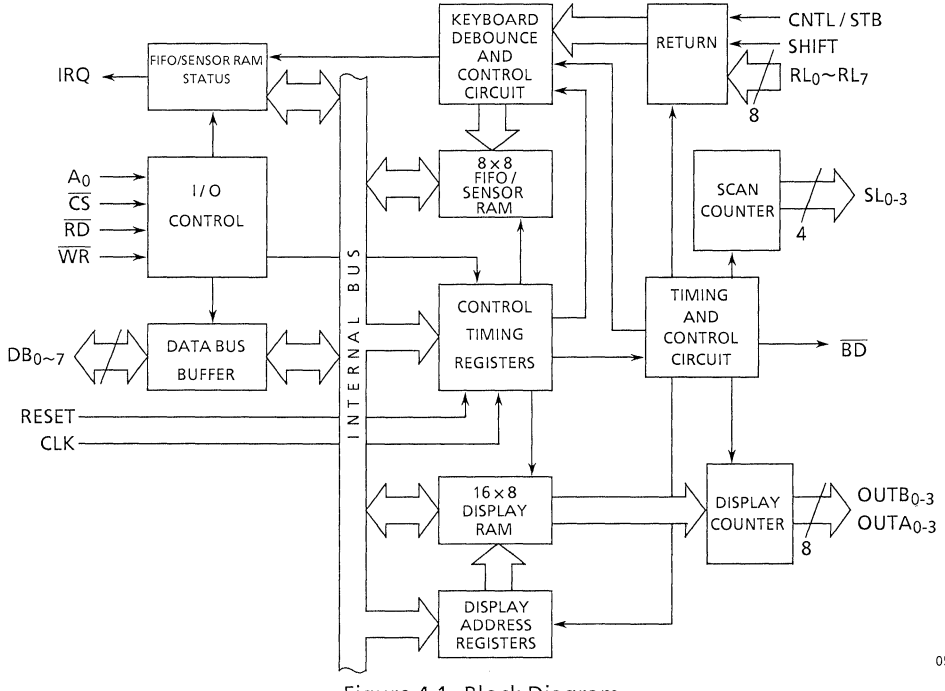
3. PIN CONNECTION



050489

Figure 3.1 Pin Connection

4. BLOCK DIAGRAM



050489

Figure 4.1 Block Diagram

5. PIN NAMES AND PIN DESCRIPTION

- VSS (Power Supply)

  Gound

- VCC (Power Supply)

  +5V during operation

- DB_0-DB_7 (Input/Output)

  Bidirectional data bus. All data and commands are transferred via this data bus.

- CLK (Input)

  System Clock used to generate the TMP82C79 internal timing.

- RESET (Input)

  A high level signal on this pin resets the TMP82C79. After being reset the TMP82C79 is placed in the following state.

  (1)16×8 bit character display, left entry.

  (2)Encode scan keyboard, 2 key lockout, clock pre-scale value is set to 31.

- $\overline{CS}$ (Input)

  A low level input on this pin enables $\overline{RD}$ and $\overline{WR}$ communication between the MPU and the TMP82C79.

- A_0 (Input)

  This input acts in conjunction with the $\overline{CS}$, $\overline{WR}$ and $\overline{RD}$ pins. A high level on this pin indicates the signals on data bus are interpreted as command or status. A low level indicates they are data in the RAMs.

- $\overline{WR}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP82C79 to accept command or data from the MPU.

- $\overline{RD}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP82C79 to output data or status onto the data bus.

- IRQ (Output)

  Interrupt request output. In keyboard mode, the interrupt line is high when the FIFO/Sensor RAM has effective data. The interrupt line goes low when each FIFO/Sensor RAM is read and returns high if the RAM still has effective data. In sensor matrix mode, the interrupt line goes high whenever any change in the sensor matrix is detected.

- SL_0-SL_3 (Output)

  Scan lines which are used to scan the key switch or sensor matrix and the display digits. These lines can be either encoded (1 of 16) or decoded (1 of 4).

- RL_0-RL_7 (Input)

  Return lines which are connected to the scan lines through the key or sensor switches. Each line has an internal pullup to keep it high until a switch closure pulls it low. They also serve as an 8-bit input in Strobed Input mode.

- SHIFT (Input)

  This input status is stored in the FIFO RAM in addition to the information of the key position on key closure in Scanned key board modes. It has an internal pullup to keep it high until a switch closure pulls it low.

- CNTL/STB (Input)

  For Keyboard modes this line is used as a control input and stored like status on a key closure. This can be also programmed as the strobe line that enters the data into FIFO in Strobed Input mode (Rising Edge). It has an internal pullup to keep it high until a switch closure pulls it low.

  $OUTA_0$-$OUTA_3$ (Output)

  $OUTB_0$-$OUTB_3$ (Output)

  These two ports are the outputs for the 16×4 display refresh registers. The data from these outputs is updated synchronized with the scan lines (SL_0-SL_3) for multiplexed digit displays. The two 4 bit ports may be blanked independently. These two ports may also be treated as one 8-bit port.

- $\overline{BD}$ (Output)

  This output is used to blank the display during digit switching or by a display blanking command.

6. FUNCTIONAL DESCRIPTION

6.1 I/O CONTROL AND DATA BUS BUFFER

The I/O control section uses the $\overline{CS}$, A_0, $\overline{RD}$ and $\overline{WR}$ lines and controls the flow of data to and from the various internal registers and buffers in the TMP82C79. $\overline{CS}$ input enables the all data flow to and from the TMP82C79. The character of the information given by the MPU, is identified by A_0.

$\overline{RD}$ and $\overline{WR}$ decide the direction of data flow through the data bus buffer. The data bus buffer is bidirectional buffer which is used for connecting the internal bus and a system bus. When $\overline{CS}$ is high, the buffer is in a high impedance state.

6.2 CONTROL REGISTER, TIMING REGISTER AND TIMING CONTROL CIRCUIT

Keyboard and display modes and the other operating conditions are programmed by the MPU. These modes are latched at the rising edge of $\overline{WR}$ when A_0 is high. The timing control contains the basic counter chains. The first counter is the 1/N prescaler that can be programmed to yield an basic internal frequncy. In case of 100kHz basic internal frequency, it gives a 5.1 ms keyboard scan time and a 10.3 ms debounce time. The other counters divide down the basic internal frequency to provide the proper key scan, row scan, keyboard matrix scan and display scan timings.

6.3 SCAN COUNTER

Two modes are available for the scan counter. In the encode mode, the counter provides a binary count that must be externally decoded to provide the scan lines for the key board and display. In the decode mode, the scan counter decodes the least significant 2 bits internally and provides a decoded 1 of 4 scan. Note that the only first 4 characters in the Display RAM are outputted from $OUTA_{0\_3}$ and $OUTB_{0\_3}$ in the decode mode.

6.4 RETURN BUFFER AND KEYBOARD DEVOUNCE CONTROL CIRCUIT

The 8 return lines are latched onto the return line buffer. In the Keyboard mode, these lines are scanned to look for key closures in a row. If the debounce circuit detects a closed switch, it waits about *10 ms, and checks if the switch remains closed. If it does so, the address of the switch and the status of SHIFT and CNTL lines is transferred to the FIFO.

In the scanned Sensor Matrix Modes, the contents of the return lines are directly transferred to the corresponding row of the sensor RAM (FIFO) each scan time.

In the Strobed Input Mode, the contents of the return lines are transferred to the FIFO on the rising edge of the CNTL/STB line pulse.

6.5 FIFO/SENSOR RAM AND FIFO/SENSER RAM STATUS

The FIFO/Sensor RAM is a dual function RAM. In the keyboard mode or In the Strobe Input mode, this RAM serves as a FIFO. The FIFO status shows whether the FIFO is empty or full and keeps the number of characters in the FIFO. In addition, there is a flag to show an error in the case where too many reads or writes is recognized. The FIFO status can be read at $\overline{CS}=\overline{RD}=0$, $A_0=1$. The FIFO status logic provides an IRQ signal when the FIFO is not empty. In the scanned sensor matrix mode, the RAM serves as a sensor RAM. IRQ becomes high when a change in the sensor is detected.

6.6 DISPLAY ADDRESS REGISTERS AND DISPLAY RAM

The display address registers hold the address of the word currently being written or read by the MPU and the two 4 bit nibbles being displayed. The Display RAM stores data for display outputs. The read/write addresses are programmed by the MPU command. They also can be programmed to autoincrement after read or write. The Display RAM can be directry read out by the MPU after mode and address is set. The A and B nibbles of the Display RAM are outputted to the Display Output A and B synchronously with scan signals (SL_0-SL_3). The A and B nibbles can be entered independently or as one word by the MPU command.

* In case of 100kHz basic internal frequency

7. COMMAND DESCRIPTION

7.1 KEYBOARD/DISPLAY MODE SET

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	0	0	D	D	K	K	K

Display Mode

0	0	8×8 bit character display-Left entry	
* 0	1	16×8 bit character display-Left entry	
1	0	8×8 bit character display-Right entry	
1	1	16×8 bit character display-Right entry	

* Default after Reset

*	0	0	0	Encoded Scan Keyboard-2-Key Lockout
	0	0	1	Decoded Scan Keyboard-2-Key Lockout
	0	1	0	Encoded Scan Keyboard-N-Key Rollover
	0	1	1	Decoded Scan Keyboard-N-Key Rollover
	1	0	0	Encoded Scan Sensor Matrix
	1	0	1	Decoded Scan Sensor Matrix
	1	1	0	Strobed Input, Encoded Display Scan
	1	1	1	Strobed Input, Decoded Display Scan

050489

7.2 PROGRAM CLOCK

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	0	1	P	P	P	P	P

050489

The TMP82C79 generates all timing and multiplexing signals by means of the internal prescaler. The prescaler generates internal reference clocks by dividing an external supply clock by a programmable value PPPPP. Any number from 2 to 31 can be set as a prescaler value. When this value is set to 0 or 1, it is interpreted to be 2. If the internal reference clock is set to 100kHz, it is possible to obtain 5.1ms keyboard scan time and 10.3ms debounce time. The value PPPPP is set to 31 after reset, but cannot be changed by the Clear command.

7.3 READ FIFO/SENSOR RAM

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	1	0	AI	×	A	A	A

× = don't care 050489

If this command is written, the subsequent data reads are set up for the FIFO/Sensor RAM. Auto-increment flag (AI) and the RAM address bits AAA are valid only in Senser Matrix Mode. The address bits AAA select one of the 8 rows of the Sensor RAM. If AI = 1, the RAM address is incremented after each successive read. The Auto-increment flag does not affect the auto-increment of the Display RAM.

7.4 READ DISPLAY RAM

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	1	1	AI	A	A	A	A

050489

If this command is written, the subsequent data reads are set up for the Display RAM. The address bits AAAA select one of the 16 rows of the Display RAM. If AI = 1, the address is incremented after each read or write to the Display RAM. Since the same counter is used for both reading and writing, this command sets the next read or wirte address and the sense of the Auto-increment for both operation.

7.5 WRITE DISPLAY RAM

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	0	0	AI	A	A	A	A

050489

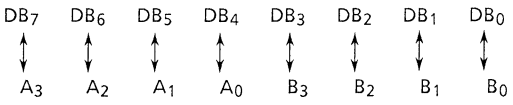
If this command is written, the subsequent data writes are set up for the Display RAM. Note that writing this command does not switch the source of the subsequent data reads. The address register of the Display RAM is same for read/write operations. The addressing and Auto-increment function are identical to those for the Read Display RAM.

7.6 DISPLAY WRITE INHIBIT/BLANKING

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	0	1	×	IWA	IWB	BLA	BLB

× = don't care 050489

The IWA or IWB bit can be used to mask A nibble or B nibble for entering the Display data independently. The BLA of BLB flag is available for the nibble A or B to blank the display. In the case where the Display Outputs are used as separate 4-bit display ports, the IWA or IWB bit is useful so as not to affect the other display port when the MPU writes a word to the display RAM. The BLA or BLB bit is used for blanking the display independently without giving any affect to the other 4-bit display port. The blank code is determined by the last Clear command that has been programmed after reset. If the Display Output is used as an 8-bit port, it is necessary to set both BLA and BLB bits for blnaking the display. Then $\overline{BD}$ signal becomes low.



Correspondence between Display Output and Data Bus

050489

7.7 CLEAR

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	1	0	CD	CD	CD	CF	CA

050489

The CD bits are used to clear all rows of the Display RAM to the following code shown below.

(DB_4) CD	(DB_3) CD	(DB_2) CD	
1	0	×	···· All Zeros (X = Don't Care)
1	1	0	···· All Hex 20H (0010 0000)
1	1	1	···· All Ones
0	×	×	···· not clear display if CA = 0

Enable clear display when CD = 1 (or by CA = 1)

050489

While the Display RAM is being cleared, it may not write to the Display RAM. The MSB bit of the FIFO status word is set during this time. If the CF bit is set to "1", the FIFO status is cleared and the interrupt request output (IRQ) is reset. Also, the Senser RAM pointer is set to the row 0.

The CA bit has the combined effect of the CD bit and CF bit. It enables clear display code to the Display RAM and also clears the FIFO status. Furthermore, it re-synchronizes the internal timing chains.

7.8 END INTERRUPT/ERROR MODE SET

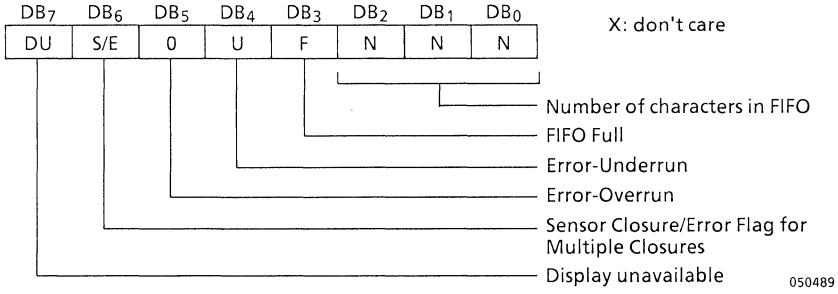
DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	1	1	E	×	×	×	×

× = don't care

050489

In the Sensor Matrix mode, this command loweres the IRQ line and enables writing to the sensor RAM. This means that a write to the Sensor RAM is inhibited when IRQ line is high. If the E bit is set to "1", the S/E bit of the FIFO status becomes "1" when any one of the sensor switches is closed. If E = 0, the S/E bit is always "0". In the N-Key Rollover mode, if the E bit is programmed to "1", the Special Error mode will be resulted.

7.9 FIFO STATUS



Du : Indicates that the Display RAM was unavailable because a Clear Display or Clear All command has not completed its clearing operation.

S/E : In a Sensor Matrix mode, if the E bit of End Interrupt/error mode set is programmed to "0", this S/E bit is set to indicate that at least one sensor closure indication is contained in the Sensor RAM.
In Special Error Mode, this S/E bit is showing the error flag and serves as an indication to whether a simultaneous multiple closure error has occurred.

O : indicates that the entry of another character into a full FIFO was attempted.

U : indicates that the MPU tried to read an empty FIFO.

F : indicates that the FIFO is full of the eight characters.

NNN : indicates number of characters in the FIFO when in the Keyboard Mode or in the Strobe Input Mode.

Table 7.1 ADDRESSING

$\overline{CS}$	A_0	$\overline{RD}$	$\overline{WR}$	Functions
0	0	0	1	Read Data
0	0	1	0	Write Data
0	1	0	1	Read Status word
0	1	1	0	Write Command word
1	×	×	×	High-impedance state

050489

8. INTERFACE WITH KEYBOARD

8.1 SCANNED KEYBOARD, 2-KEY LOCKOUT

In this mode, if one key only is kept depressed during one debounce cycle (2 times of the key scan cycle), the key is recognized. When a key is depressed, the debounce logic is set and the other depressed keys are checked during the next two scan cycle. If none are encountered, it is a single key depression and the key position is entered into the FIFO along with the status of CNTL and SHIFT lines. If another depressed key are encountered, operates as follows.

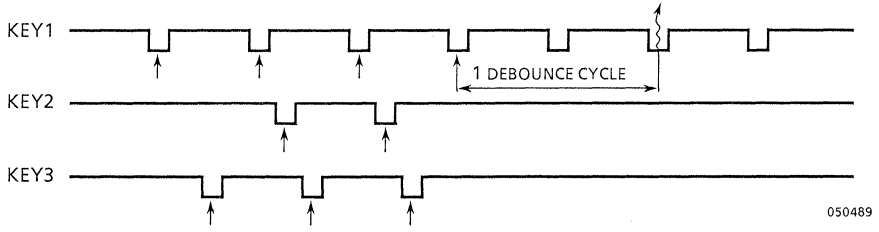


Figure 8.1 Example of a Case Where a First Depressed Key is Continuously Kept to the Last

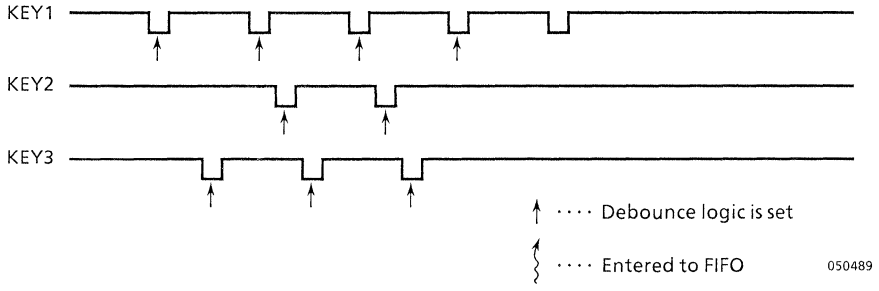


Figure 8.2 Example of a Case Where all Depressed Keys are Ignored

As shown in Figure 8.1 if all the other keys are released before the first depressed key, the first depressed key is recognized. As shown in Figure 8.2 if the first depressed key is released within one debounce cycle after the other keys was released, then all keys are ignored.

8.2 SCANNED KEYBOARD, N-KEY ROLLOVER

In this mode, each key depression is independently treated from all others. In the 2-key lockout mode, if a key is depressed, the debounce logic is set. If the other keys are depressed within one debounce cycle after it, the debounce logic is set again. The first depressed key is ignored. In the N-key Rollover mode, if a key is depressed waits one debounce cycle and then checks of the key is still down. If it is, the key is entered into the FIFO even of other keys are depressed.

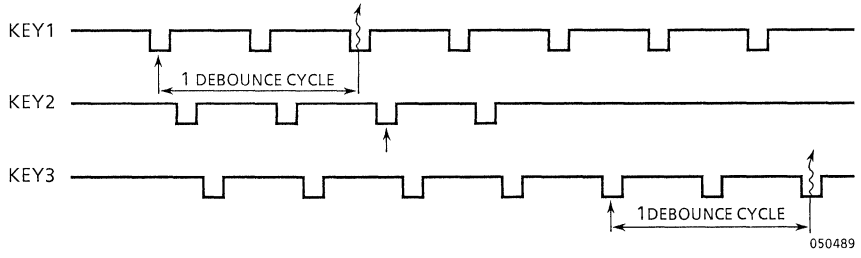


Figure 8.3 Example of 3 Keys Being Pushed Simultaneously

In the example as shown in Figure 8.3 the debounce circuit starts by Key 1, and checks if the key is still down after one debounce cycle. If it is, Key 1 is recognized and Key 2 is ignored not to be depressed for one debounce cycle.

8.3 SPECIAL ERROR MODE (N-KEY ROLLOVER)

This mode is set if the E bit of the End interrupt/error mode set command is programmed to "1". In the normal N-Key Rollover Mode, the key information is entered to the FIFO according to the key scan timing even if a simultaneous multiple depression occures during one debounce cycle. In the Special Error Mode, if a simultaneous multiple depression occurs during one debounce cycle, sets the error falg (the S/E bit of the FIFO status word) to "1". This flag prevents any further writing into the FIFO and will set interrupt request (IRQ). The S/E bit is cleared if the normal Clear command is written with CF = 1.

8.4 SENSOR MATRIX MODE

In Sensor Matrix Mode, the debounce circuit does not operate. The status of the sensor switch is inputted directly to the Sensor RAM. The MPU can only know a validated closure in the keyboard mode, however this mode has such advantage that the CPU knows how long the sensor was closed and when it was released. If there is any change in the sensor value at the end of the sensor matrix scan, the IRQ line goes high. The IRQ line is cleared by the first data read if the Auto-increment flag is "0" or by the End Interrupt/error mode set command if AI = 1.

8.5 STROBE INPUT MODE

In Strobe Input Mode, the debounce circuit does not operate. The data is inputted into the FIFO from the return lines at the rising edge of CNTL/STB Signal. When the data is entered into the FIFO, the IRQ line goes high. The functions of the FIFO and the FIFO status in this mode are same as those in the keyboard mode.

9. DATA FORMAT

9.1 KEYBOARD MODE

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
CNTL	SHIFT	SCAN			RETURN		

050489

In this mode, the Data Format of the character entered into the FIFO is as follows. The MSB is the status of CNTL/STB line and the next MSB shows the status of SHIFT line. The next three bits are from the scan counter and indicate the row the key was found in. The last three bits are from the column counter and indicate to which return line the key was connected.

9.2 SENSOR MATRIX MODE

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
RL_7	RL_6	RL_5	RL_4	RL_3	RL_2	RL_1	RL_0

050489

In this mode, the data on return lines is inputted in the row of the Sensor RAM in order according to the scan. The data is entered even if there is no change in the status of the sensor matrix switches. Each switch position maps to a Sensor RAM position. CNTL and SHIFT signals are ignored.

9.3 STROBE INPUT MODE

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
RL_7	RL_6	RL_5	RL_4	RL_3	RL_2	RL_1	RL_0

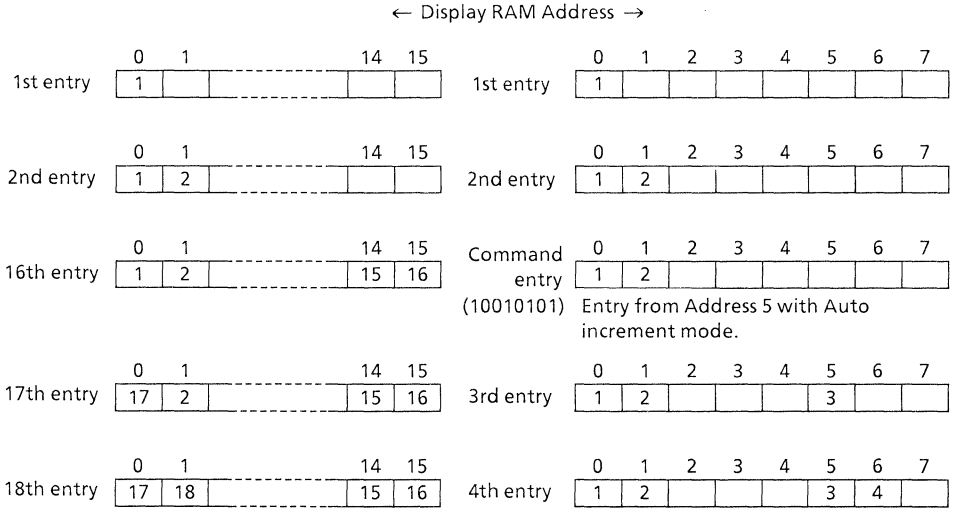
050489

In this mode, the data on return lines is entered into FIFO at the rising edge of CNTL/STB signal.

10. INTERFACE WITH DISPLAY

10.1 LEFT ENTRY

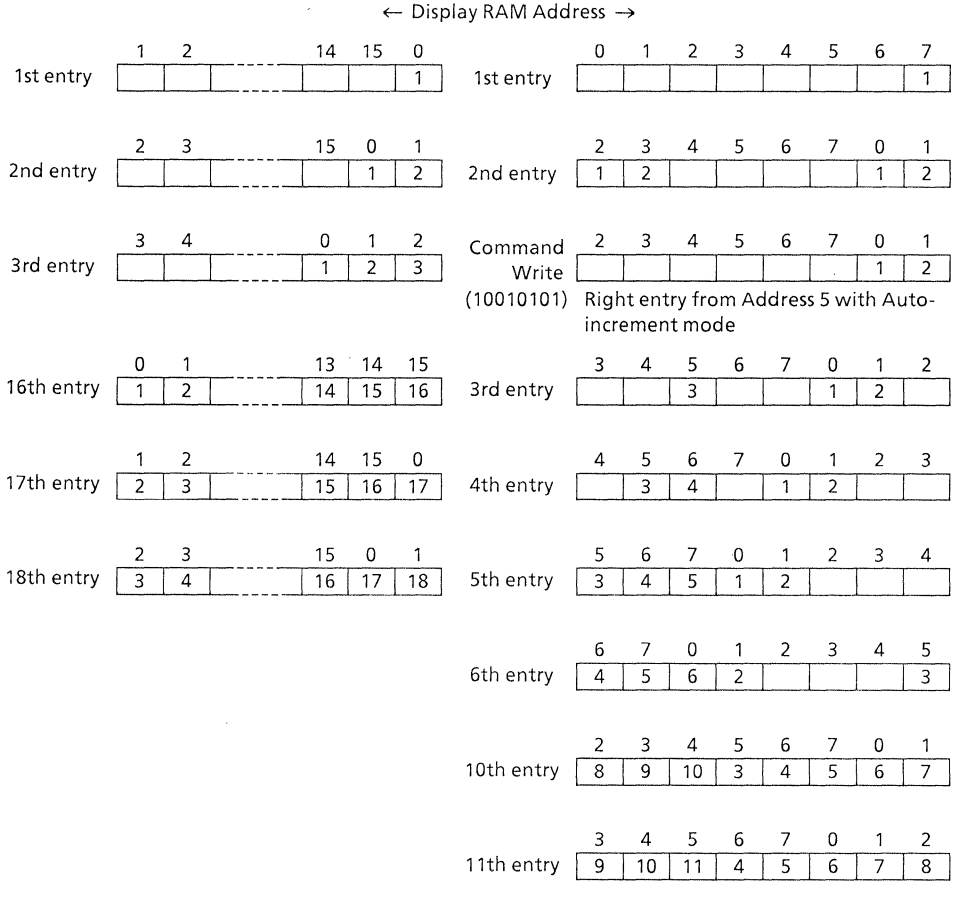
In Left Entry mode, address 0 of the Display RAM is the left-most side of the display and address 15 (address 7 in the case of 8-character display) is the right-most side. when characters are inputted onto the display RAM with the auto increment mode from address 0 of the display RAM, Characters are filled from the left-most position of the display. The 17th (or 9th) character is placed in the left-most position again. Address of the display RAM corresponds directly to each display positon of the display, and so its position does not change every entry.



050489

10.2 RIGHT ENTRY

In Right Entry, the first entry is from the right-most position. Address of the Display RAM does not correspond to the display position.



050489

11. ELECTRIC CHARACTERISTICS

11.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING	UNIT
V_{CC}	VCC Supply Voltage (with respect to VSS (GND))	− 0.5 to + 7.0	V
V_{IN}	Input Voltage (with respect to VSS (GND))	− 0.5 to VCC + 0.5	V
V_{OUT}	Output Voltage (with respect to VSS (GND))	− 0.5 to VCC + 0.5	V
P_D	Power Dissipation	250	mW
T_{sol}	Soldering Temperature (soldering time 10 sec)	260	°C
T_{stg}	Storage Temperature	− 65 to + 150	°C
T_{opr}	Operating Temperature	− 40 to + 85	°C

050489

11.2 D.C. ELECTRICAL CHARACTERISTICS

(Ta = − 40 to + 85 °C, V_{CC} = 5V ± 10%, V_{SS} (GND) = 0V)

SYMBOL	PARAMETER	CONDITION	MIN.	TYP.	MAX.	UNIT
V_{IL1}	Input Low Voltage (RL_0-RL_7)		− 0.5	—	1.4	V
V_{IL2}	Input Low Voltage (Others)		− 0.5	—	0.8	V
V_{IH1}	Input High Voltage (RL_0-RL_7)		2.2	—	V_{CC} + 0.5	V
V_{IH2}	Input High Voltage (Others)		2.2	—	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	IOL = 2.2mA	—	—	0.45	V
V_{OH1}	Output High Voltage	IOH = -400μA	2.4	—	—	V
V_{OH2}	Output High Voltage	IOH = -100μA	Vcc − 0.8	—	—	V
I_{IL1}	Input 1LEAK Current (SHIFT, CNTL, RL_0-RL_7)	VIN = VCC	—	—	+ 10	μA
		VIN = 2.4V	− 10	− 30	—	
		VIN = 0V	—	—	− 100	
I_{IL2}	Input Leak Current (Others)	0V ≦ VIN ≦ VCC	—	—	± 10	μA
I_{OFL}	Output Leak Current	0.45V ≦ VOUT ≦ VCC	—	—	± 10	μA
I_{CC}	Operating Supply Current	VIH = VCC-0.2V VIL = 0.2V, fc = 5MHz	—	—	5	mA

050489

11.3 INPUT CAPACITY

SYMBOL	PARAMETER	CONDITION	MIN.	TYP.	MAX.	UNIT
C_{IN}	Input Capcity	fc = 1MHz Unmeasured Pins returned to VSS.	—	5	10	pF
C_{OUT}	Output Capcity		—	10	20	pF

050489

11.4 A.C. ELECTRICAL CHARACTERISTICS

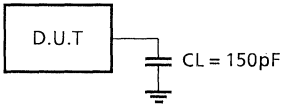
Ta = −40 to +85° C, VCC = 5.0V ± 10%, VSS (GND) = 0V

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t_{AR}	Address Set up Time ($\overline{RD}$)	10	—	—	ns
t_{RA}	Address Hold Time ($\overline{RD}$)	10	—	—	ns
t_{RR}	$\overline{RD}$ Pulse Width	160	—	—	ns
*t_{RD}	Valid Data ($\overline{RD}$)	—	—	120	ns
*t_{AD}	Address to Valid Data	—	—	185	ns
t_{DF}	Data Floating ($\overline{RD}$)	10	—	85	ns
t_{RCY}	Read cycle Time	200	—	—	ns
t_{AW}	Address Set up Time ($\overline{WR}$)	0	—	—	ns
t_{WA}	Address Hold Time ($\overline{WR}$)	0	—	—	ns
t_{WW}	$\overline{WR}$ Pulse Width	160	—	—	ns
t_{DW}	Data Set up Time ($\overline{WR}$)	120	—	—	ns
t_{WD}	Data Hold Time ($\overline{WR}$)	30	—	—	ns
t_{WCY}	Write Cycle Time	200	—	—	ns
t_{OWH}	CLK Pulse Width of High Level	80	—	—	ns
t_{OWL}	CLK Pulse Width of Low Level	50	—	—	ns
t_{CY}	Clock Period	200	—	—	ns

050489

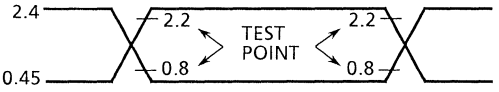
*TEST CONDITION CL = 150pF

11.4.1 AC TEST CONDITION



050489

11.4.2 AC TEST INPUT WAVEFORM



050489

12. TIMING DIAGRAM

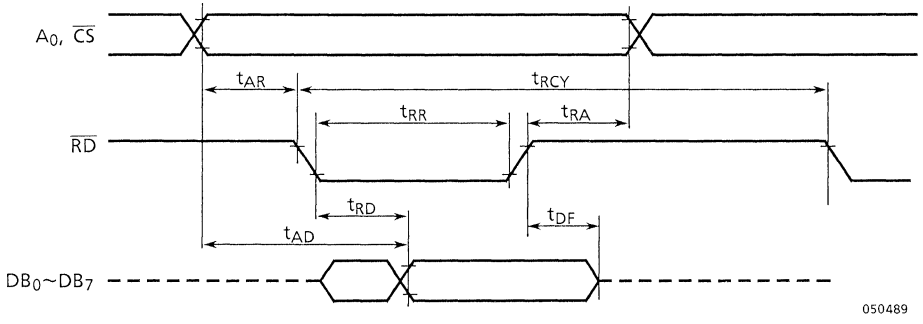


Figure 12.1 READ TIMING

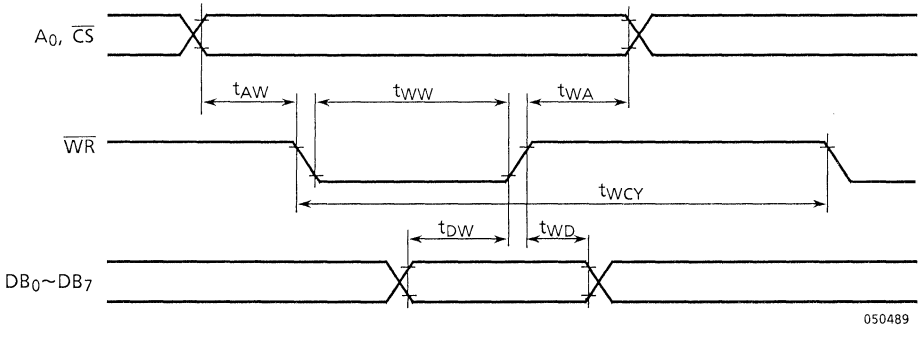


Figure 12.2 WRITE TIMING

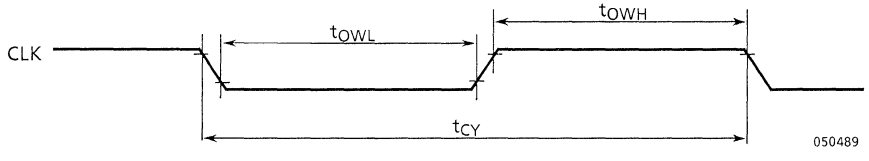


Figure 12.3 CLOCK TIMING

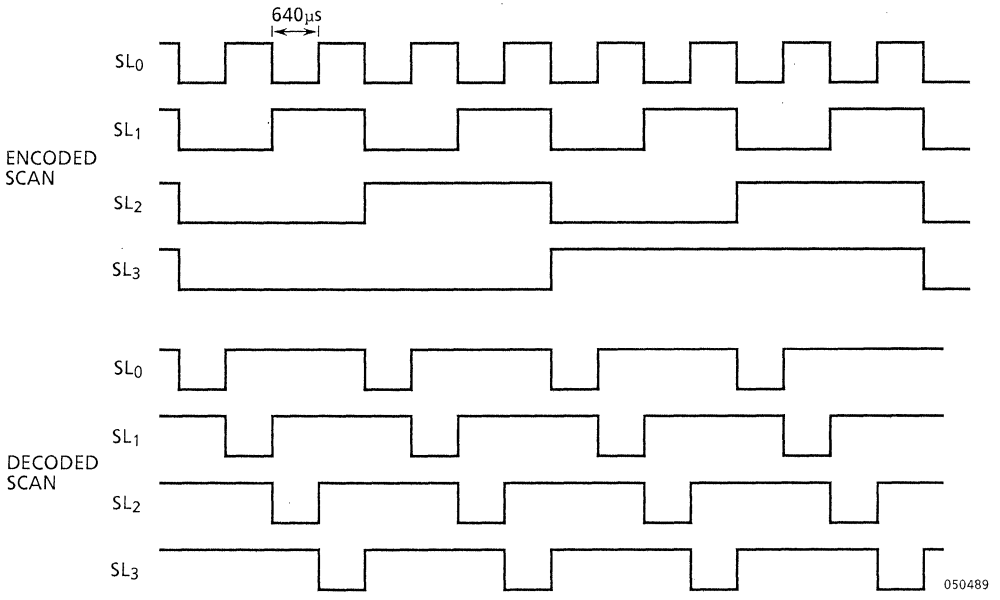


figure 12.4 scan timing

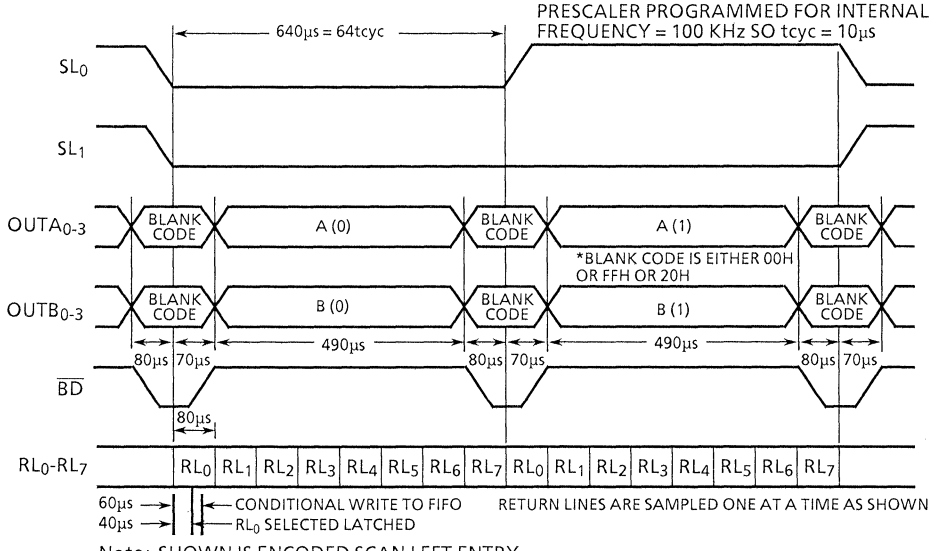


figure 12.5 example of display timing

13. EXAMPLE OF APPLICATION CIRCUIT

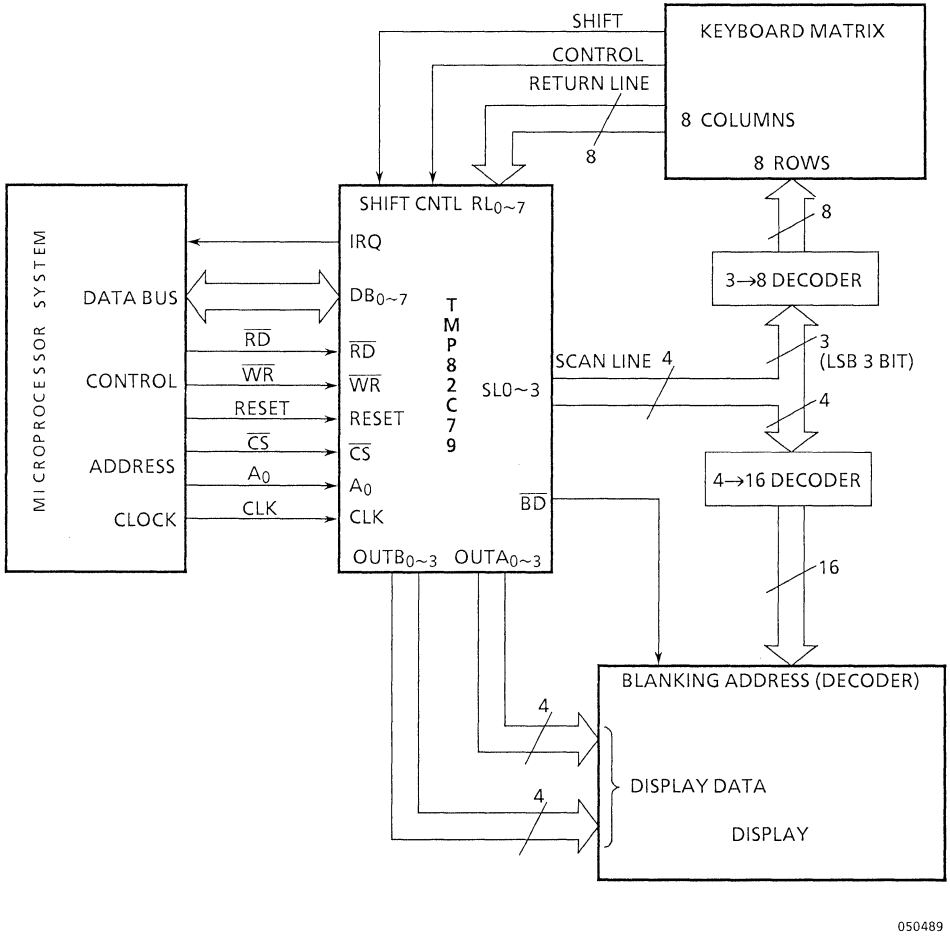


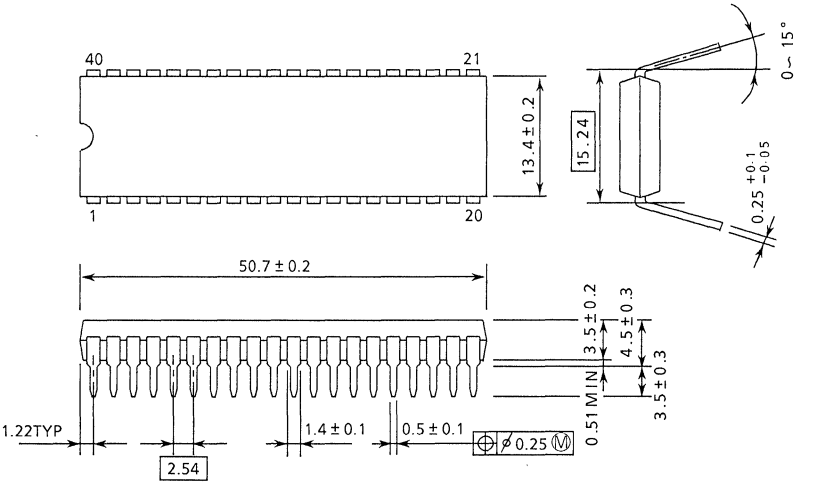
Figure 13.1 EXAMPLE OF APPLICATION CIRCUIT

14. EXTERNAL DIMENSION

14.1 40PIN DIP EXTERNAL DIMENSION

DIP40-P-600

Unit : mm



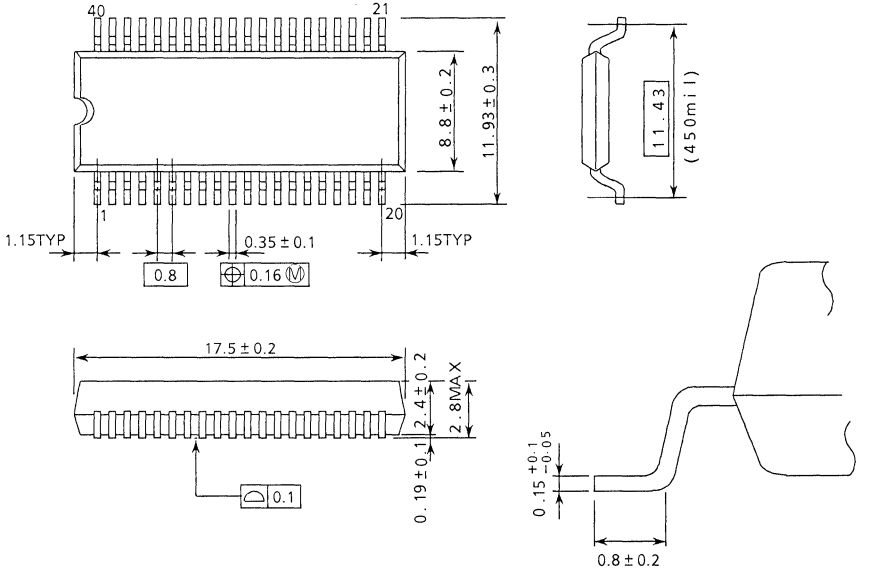
050489

Note : Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoritical positions with respect to No.1 and No.40 leads.

14.2 40PIN SOP EXTERNAL DIMENSION

SSOP40-P-450

Unit : mm



050489

Note : Package Width and Length do not include Mold Protrusions.
Allowable Mold Protrusion is 0.15mm.

PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

TMP8279P-5

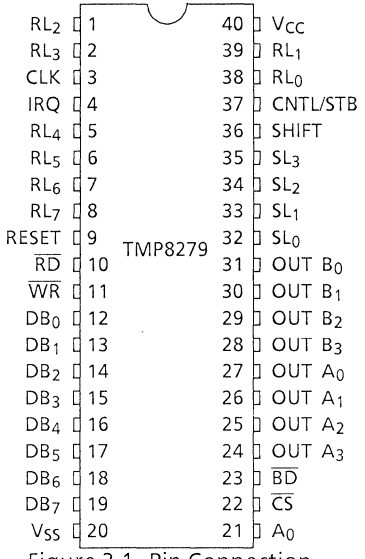
1. GENERAL DESCRIPTION

The TMP8279P-5 (hereinafter referred to as TMP8279) is a programmable keyboard/display interface designed for use as the TLCS-85A microcomputer peripheral. The keyboard portion can provide a scanned interface to a 64-contact key matrix. Also, the keyboard portion can interface to an array of sensors or a strobed interface keyboard. Key depressions can be 2-key lockout or N-key rollover. The display portion has 16×8 bits display RAM which can be treated as dual 16×4 bits. Both right entry and left entry display formats are possible.

2. FEATURES

- Simulataneous Keyboard/Display operation is possible
- Scanned Keyboard mode.
- Scanned Sensor Matrix mode.
- Strobed Input Entry mode.
- Buil in 8-character FIFO or 64 bit Sensor RAM
- Programmable 2 Key Lockout or N-key Rollover with contact Debounce.
- Built in 16×8 bit display RAM.
- Programmable scan timing
- Compatible with INTEL 8279-5.

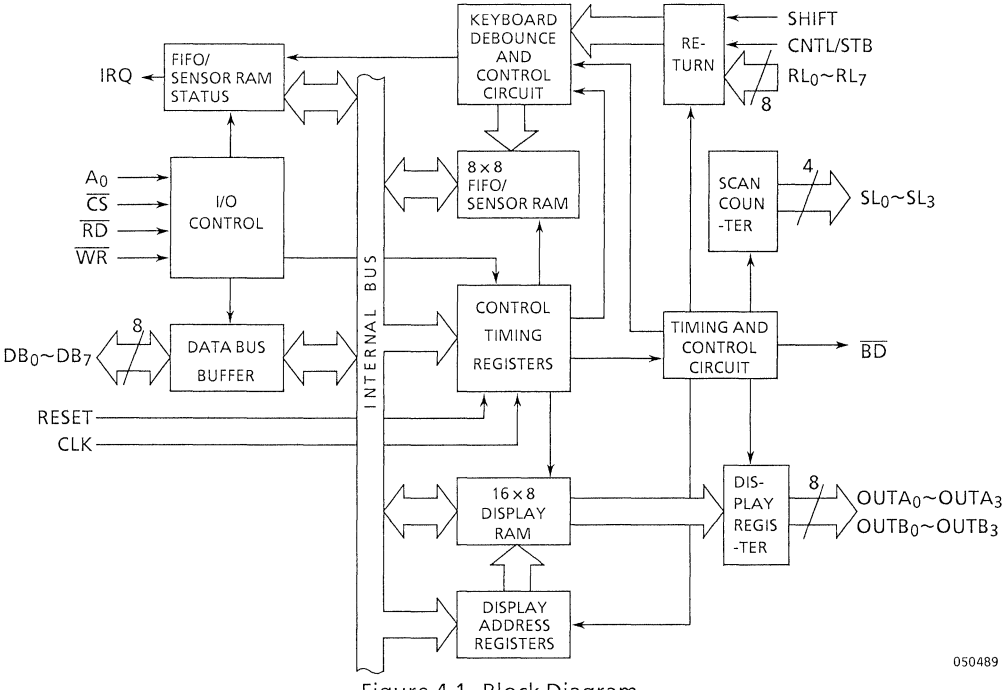
3. PIN CONNECTION



050489

Figure 3.1 Pin Connection

4. BLOCK DIAGRAM



050489

Figure 4.1 Block Diagram

5. PIN NAME AND PIN DESCRIPTION

- VSS (Power Supply)

  Ground

- VCC (Power Supply)

  +5V during operaiton

- DB_0~DB_7 (Input/Output)

  Bidirecitonal Data Bus. All data and commands are transfered via this data bus.

- CLK (Input)

  System clock used to generate the TMP8279 internal timing.

- RESET (Input)

  A high level signal on this pin resets the TMP8279. After being reset the TMP8279 is placed in the following state.

  (1) 16×8 bit character display, left entry.

  (2) Encode scan keyboard, 2 key lockout, clock pre-scale value is set to 31.

- $\overline{CS}$ (Input)

  A low level input on this pin enables $\overline{RD}$ and $\overline{WR}$ communication between the MPU and the TMP8279.

- A_0 (Input)

  This inputs acts in conjunction with the $\overline{CS}$, $\overline{WR}$ and $\overline{RD}$ pins. A high level input on this pin indicates the signals on data bus are interpreted as command or status. A low level input indicates they are data in the RAMs.

- $\overline{WR}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP8279 to accept command or data from the MPU.

- $\overline{RD}$ (Input)

  A low level input on this pin when $\overline{CS}$ is low enables the TMP8279 to output data or status onto the bus.

- IRQ (Output)

  Interrupt request output. In a keyboard mode, the interrupt line is high when the FIFO/Sensor RAM has effective data. The interrupt line goes low when each FIFO/Sensor RAM read and returns high if the RAM still has effective data. In sensor matrix mode, the interrupt line goes high whenever any change in the sensor matrix is detected.

- SL_0~SL_3 (Output)

  Scan lines which are used to scan the key switch or the sensor matrix and the display digits. These lines can be either encoded (1 of 16) of decode (1 of 4).

- RL_0~RL_7 (Input)

  Return lines which are connected to the scan lines through the keys or sensor switches. Each line has an internal pullup to keep it high until a switch closure pulls it low. They also serve as an 8-bit input in Strobed Input mode.

- SHIFT (Input)

  This input status is stored in the FIFO RAM in addition to information of the key position on key closure in Scannned key board modes. It has an internal pullup to keep it high until a switch closure pulls it low.

- CNTL/STB (Input)

  For Keyboard modes this line is used as a control input and stored like status on a key closure. This can be programmed as the strobe line that enters the data into FIFO in Strobed Input mode (Rising Edge). It has an internal pullup to keep it high until a switch closure pulls it low.

- $OUTA_0$~$OUTA_3$ (Output)
- $OUTB_0$~$OUTB_3$ (Output)

  These two ports are the outputs for the 16×4 display refresh registers. The data from these outputs is updated synchronized with the scan lines (SL_0~SL_3) for multiplexed digit displays. The two 4 bit ports may be blanked independently. These two ports may also be treated as one 8-bit port.

- $\overline{BD}$ (Output)

  This output is used to blank the display during digit switching or by a display blanking command.

6. FUNCTIONAL DESCRIPTION

6.1 I/O CONTROL AND DATA BUS BUFFER

The I/O control section uses the $\overline{CS}$, A_0, $\overline{RD}$ and $\overline{WR}$ lines and controls the flow of data to and from the various internal registers and buffers in the TMP8279. $\overline{CS}$ input enables the all data flow to and from the TMP8279. The character of the information given by the MPU, is identified by A_0. $\overline{RD}$ and $\overline{WR}$ decide the direction of data flow through the data bus buffer. The data bus buffer is bidirectional buffer which is used for connecting the internal bus and a system bus. When $\overline{CS}$ is high, the buffer is in a high impedance state.

6.2 CONTORL REGISTER, TIMING REGISTER AND TIMING CONTROL CIRCUIT

The keyboard and display modes or the other operating conditions are programmed by the MPU. These modes are latched at the rising edge of $\overline{WR}$ when A_0 is high. The timing control contains the basic counter chains. The first counter is the 1/N prescaler that can be programmed to yield an basic internal frequency which gives a 5.1ms keyboard scan time and a 10.3ms debounce time. The other counters divide down the basic internal frequency to provide the proper keyboard matrix scan and display scan timings.

6.3 SCAN COUNTER

Two modes are available for the scan counter. In the encode mode, the counter provides a binary count that must be externally decoded to provide the scan lines for the key board and display. In the decode mode, the scan counter decodes the least significant 2 bits internally and provides a decoded 1 of 4 scan. Note that the only first 4 characters in the Display RAM are outputted from $OUTA_{0\text{-}3}$ and $OUTB_{0\text{-}3}$ in the decode mode.

6.4 RETURN BUFFER AND KEYBOARD DEVOUNCE CONTROL CIRCUITS

The 8 return lines are latched onto the return line buffer. In the Keyboard mode, these lines are scanned to look for key closures in a row. If the debounce circuit detects a closed switch, it waits about 10ms*, and checks if the switch remains closed. If it does so, the address of the switch and the status of SHIFT and CNTL lines is transferred to the FIFO .

6.5 FIFO/SENSOR RAM AND FIFO/SENSOR RAM STATUS

The FIFO/Sensor RAM is a dual funciton RAM. In the keyboard mode or in the Strobe Input mode, this RAM serves as a FIFO. The FIFO status shows whether the FIFO is empty or full and keeps the number of characters in the FIFO. In addiiton, there is a flag to show an error in the case where too many reads or writes is recognized. The FIFO status can be read at $\overline{CS}=\overline{RD}=0$, $A_0=1$. The FIFO status logic provides an IRQ signal when the FIFO is not empty. In the scanned sensor matrix mode, the RAM serves as a Sensor RAM. IRQ becomes high when a change in the sensor is detected.

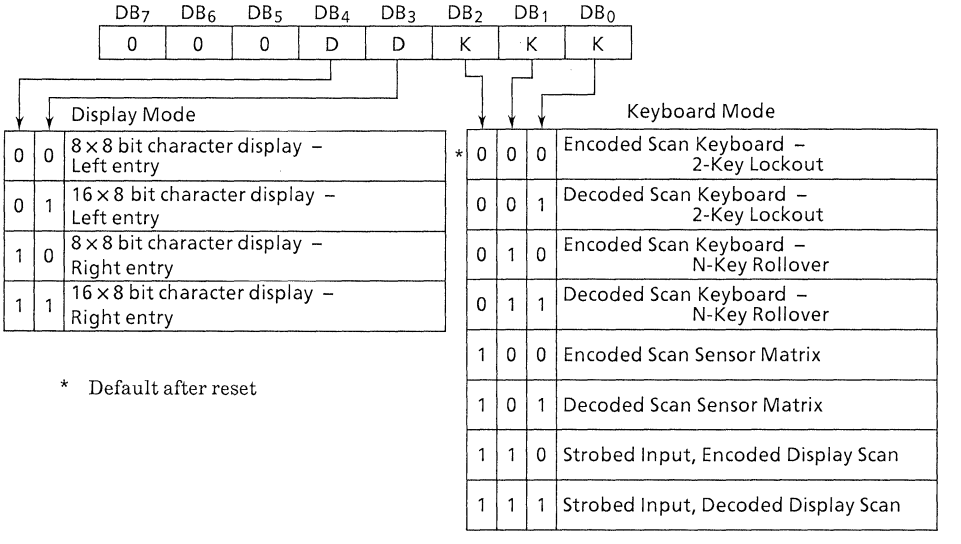
6.6 DISPLAY ADDRESS REGISTERS AND DISPLAY RAM

The display address registers hold the address of the word currently being written or read by the MPU and the two 4 bit nibbles being displayed. The Display RAM stores data for display outputs. The read/write addresses are programmed by the MPU command. They also can be programmed to auto-increment after read or wirte. The Display RAM can be directy read out by the MPU after mode and address is set. The A and B nibbles of the Display RAM are outputted to the Display Outputs A and B syncronously with scan signals (SL_0~SL_3). The A and B nibbles can be entered independently or as one word by the MPU command.

* In case of 100KHz basic internal frequency.

7. COMMAND DESCRIPTION

7.1 KEYBOARD/DISPLAY MODE SET



* Default after reset

050489

7.2 PROGRAM CLOCK

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	0	1	P	P	P	P	P

050489

The TMP8279 generates all timing and multiplexing signals by means of the internal prescaler. The prescaler generates internal reference clocks by dividing an external supply clock by a programmable value PPPPP. Any number from 2 to 31 can be set as a prescaler value. When this value is set to 0 or 1, it is interpreted to be 2. If the internal reference clock is set to 100kHz, it is possible to obtain 5.1ms keyboard scan time and 10.3ms debounce time. The value PPPPP is set to 31 after reset, but cannot be changed by the Clear command.

7.3 READ FIFO/SENSOR RAM

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	1	0	AI	X	A	A	A

X = don't care

050489

If this command is writen, the subsequent data reads are set up for the FIFO/Sensor RAM. Auto-increment flag (AI) and the RAM address bits AAA are valid only in Sensor Matrix Mode. The address bits AAA select one of the 8 rows of the Sensor RAM. If AI=1, the RAM address is incremented after each successive read. The Auto-incremented flag does not affect the auto-increment of the Display RAM.

7.4 READ DISPLAY RAM

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
0	1	1	AI	A	A	A	A

050489

If this command is written, the subsequent data reads are set up for the Display RAM. The address bits AAAA select one of the 16 rows of the Display RAM. If AI=1, the address is incremented after each read or write to the Display RAM. This command sets the next read or write address and the sense of the Auto-increment.

7.5 WRITE DISPLAY RAM

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	0	0	AI	A	A	A	A

050489

If this command is written, the subsequent data writes are set up for the Display RAM. Note that writing this command does not switch the source of the subsequent data reads. The address register of the Display RAM is same for read/write operations. The addressing and Auto-increment funciton are identical to those for the Read Display RAM.

7.6 DISPLAY WRITE INHIBIT/BLANKING

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	0	1	X	IWA	IWB	BLA	BLB

X = don't care

050489

The IWA or IWB bit can be used to mask A nibble or B nibble for entering the Display data independently. The BLA or BLB flag is available for the nibble A or B to blank the display. In the case where the Display Outputs are used as separate 4-bit display ports, the IWA or IWB bit is useful so as not to affect the other display port when the MPU writes a word to the display RAM. The BLA or BLB bit is used for blanking the display independently without giving any affect to the other 4-bit display port. The blank code is determined by the last Clear command that has been programmed after reset. If the Display Output is used as an 8-bit port, it is necessary to set both BLA and BLB bits for blanking the display. The $\overline{BD}$ signal becomes low.

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
↕	↕	↕	↕	↕	↕	↕	↕
A_3	A_2	A_1	A_0	B_3	B_2	B_1	B_0

Correspondence between Display Output and Data Bus

050489

7.7 CLEAR

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	1	0	C_D	C_D	C_D	C_F	C_A

050489

The C_D bits are used to clear all rows of the Dislay RAM to the following code shown below.

(DB_4) C_D	(DB_3) C_D	(DB_2) C_D	
1	0	×	···· All Zeros (X = Don't Care)
1	1	0	···· All Hex 20H (0010 0000)
1	1	1	···· All Ones
0	×	×	···· not clear display if $C_A = 0$

↑ Enable clear display when $C_D = 1$ (or by $C_A = 1$)

050489

While the Display RAM is being cleared, it may not write to the Display RAM. The MSB bit of the FIFO status word is set during this time. If the C_F bit is set to "1", the FIFO status is cleared and the interrupt request output (IRQ) is reset. Also, the Sensor RAM pointer is set to the row 0.

The C_A bit has the combined effect of the C_D bit and C_F bit. It enables clear display code to the Display RAM and also clears the FIFO status. Furthermore, it re-synchronizeds the inernal timing chains.

7.8 END INTERRUPT/ERROR MODE SET

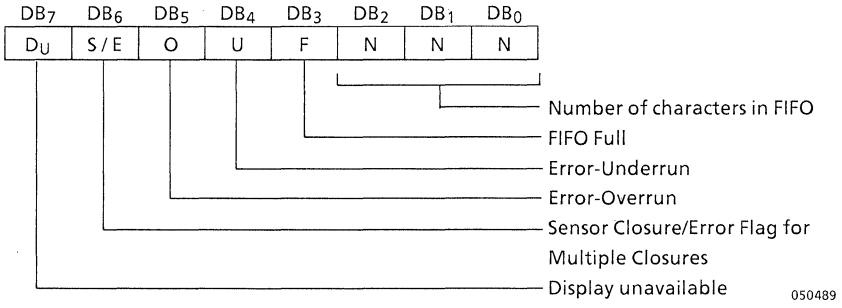
DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
1	1	1	E	×	×	×	×

X: don't care

050489

In the Sensor Matrix mode, this command loweres the IRQ line and enables writing to the Sensor RAM. This means that a write to the Sensor RAM is inhibited when IRQ line is high. If the E bit is set to "1", the S/E bit of the FIFO status becomes "1" when any one of the sensor switches is closed. If E = 0, the S/E bit is always "0". In the N-Key Rollover, if the E bit is progrmamed to "1", the Special Error mode will be resulted.

7.9 FIFO STATUS



Du : indicates that the Display RAM is unavailable because a Clear Display or Clear All command has not completed its clearing operation.

S/E : in a Sensor Matrix mode, if the E bit of End Interrupt/Error Mode Set is programmed to "1", this S/E bit is set to indicate that at least one sensor closure indication is contained in the Sensor RAM.
In Special Error Mode, this S/E bit is showing the error flag and serves as an indication to whether a simulataneous multiple closure error has occured.

O : indicates that the entry of another character into a full FIFO was attempted.

U : indicates that the MPU tried to read an empty FIFO.

F : indicates that the FIFO is full of the eight characters.

NNN : indicate number of characters in the FIFO when in the Keyboard Mode or in the Strobe Input Mode.

Table 7.1 Addressing

$\overline{CS}$	A_0	$\overline{RD}$	$\overline{WR}$	Functions
0	0	0	1	Read Data
0	0	1	0	Write Data
0	1	0	1	Read Status word
0	1	1	0	Write Command word
1	×	×	×	High-impedance state

050489

8. INTERFACE WITH KEYBOARD

8.1 SCANNED KEYBOARD, 2-KEY LOCKOUT

In this mode, if one key only is kept depressed during one debounce cycle (2 times of the key scan cycle), the key is recognized. When a key is depressed, the debounce logic is set and the other depressed keys are checked during the next two scan cycle. If none are encountered, it is a single key depressing and the key position is entered into the FIFO along with the status of CNTL and SHIFT lines. If another depressed key are encountered, operates as follows.

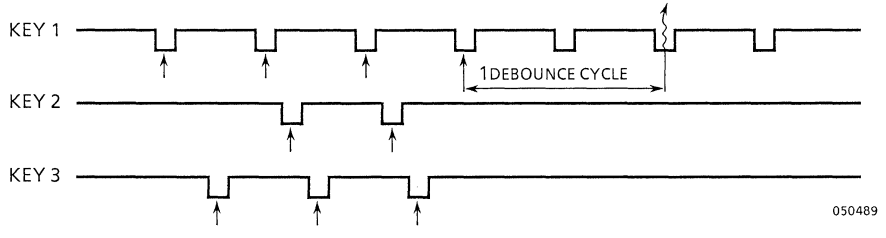


Figure 8.1 Example of a case where a first depressed key is continuously kept to the last

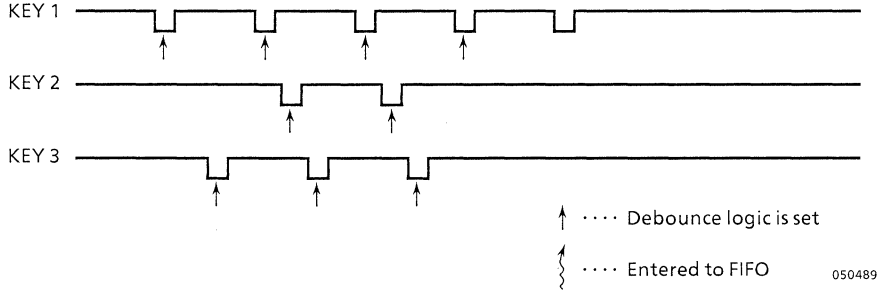


Figure 8.2 Example of a case where all depressed keys are ignored

As shown in Figure 8.1, if all the other keys are released before the first depressed key, the first depressed key is recognized. As shown in Figure 8.2, if the first depressed key is released within one debounce cycle after the other keys was released, than all keys are ignored.

8.2 SCANNED KEYBOARD, N-KEY ROLLOVER

In this mode, each key depression is independently treated from all others. In the 2-Key lockout mode, if a key is depressed, the debounce logic is set. If the other keys are depressed within one debounce cycle after it, the debounce logic is set again. The first

depressed key is ignored. In the N-key Rollover mode, if a key is depressed waits one debounce cycle and then checks if the key is still down. If it is, the key is entered into the FIFO even if other keys are depressed.

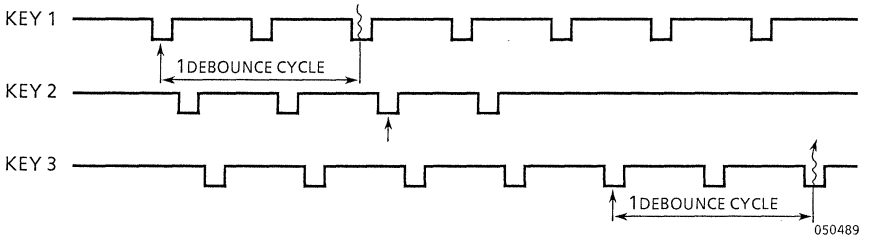


Figure 8.3 Example of 3 keys being pushed simultaneously

In the example as shown in Figure 8.3, the debounce circuit starts by Key 1, and checks if the key is still down after one debounce cycle. If it is, Key 1 is recognized and Key 2 is ignored not to be depressed for one debounce cycle.

8.3 SPECIAL ERROR MODE (N-KEY ROLLOVER)

This mode is set if the E bit of the End Interrupt/error Mode Set command is programmed to "1". In the normal N-Key Rollover Mode, the key information is entered to the FIFO according to the key scan timing even if a simulataneous multiple depression occures during one debounce cycle. In the Special Error Mode, if a simulataneous multiple depression occurs during one debounce cycle, sets the error flag (the S/E bit of the FIFO status word) to "1". This flag prevents any further writing into the FIFO and will set interrupt request (IRQ). The S/E bit is cleared if the normal Clear command is written with $C_F = 1$.

8.4 SENSOR MATRIX MODE

In Sensor Matrix Mode, the debounce circuit does not operate. The status of the sensor switch is inputted directly to the Sensor RAM. The MPU can know a validated closure in the keyboard, however this mode has such advantage that the MPU knows how long the sensor was closed and when it was released.

If there is any change in the sensor value at the end of the sensor matrix scan, the IRQ line goes high. The IRQ line is cleared by the first data read if the Auto-increment flag is "0" or by the End Interrupt/Error Mode Set command if AI = 1.

8.5 STROBE INPUT MODE

In Strobe Input Mode, the debounce circuit does not operate. The data is inputted into the FIFO from the return lines at the rising edge of CNTL/STB Signal. When the data is entered into the FIFO, the IRQ line goes high.

The functions of the FIFO and the FIFO status in this mode are same as those in the keyboard mode.

9. DATA FORMAT

9.1 KEYBOARD MODE

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
CNTL	SHIFT	SCAN			RETURN		

050489

In this mode, the Data Format of the character entered into the FIFO is as follows. The MSB is the status of CNTL/STB line and the next MSB shows the status of SHIFT line. The next three bits are from the scan counter and indicate the row the key was found in. The last three bits are from the column counter and indicate to which return line the key was connected.

9.2 SENSOR MATRIX MODE

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
RL_7	RL_6	RL_5	RL_4	RL_3	RL_2	RL_1	RL_0

050489

In this mode, the data on return lines is inputted in the row of the Sensor RAM in order according to the scan. The data is entered even if there is no change in the status of the sensor matrix switches. Each switch position maps to a Sensor RAM position. CNTL and SHIFT signals are ignored.

9.3 STROBE INPUT MODE

DB_7	DB_6	DB_5	DB_4	DB_3	DB_2	DB_1	DB_0
RL_7	RL_6	RL_5	RL_4	RL_3	RL_2	RL_1	RL_0

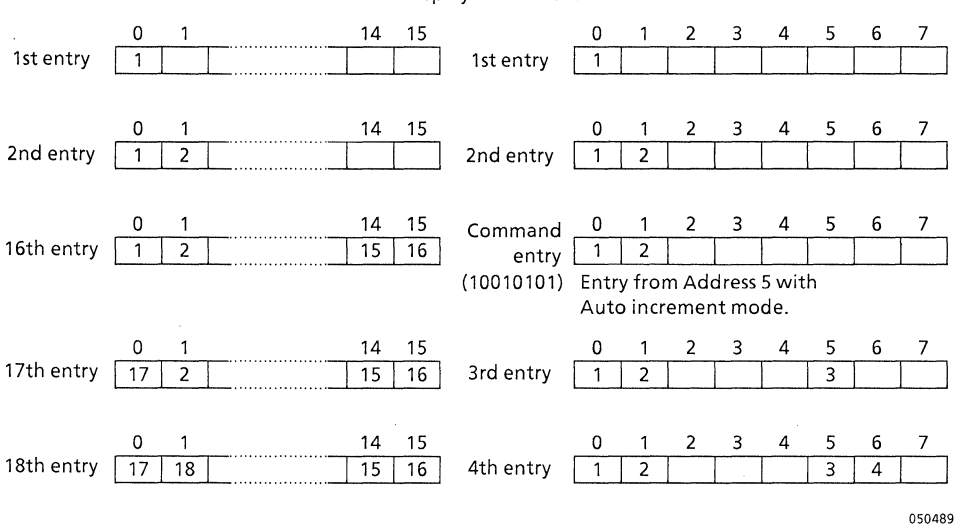
050489

In this mode, the data on the return line is entered into the FIFO at the rising edge of CNTL/STB signal.

10. INTERFACE WITH DISPLAY

10.1 LEFT ENTRY

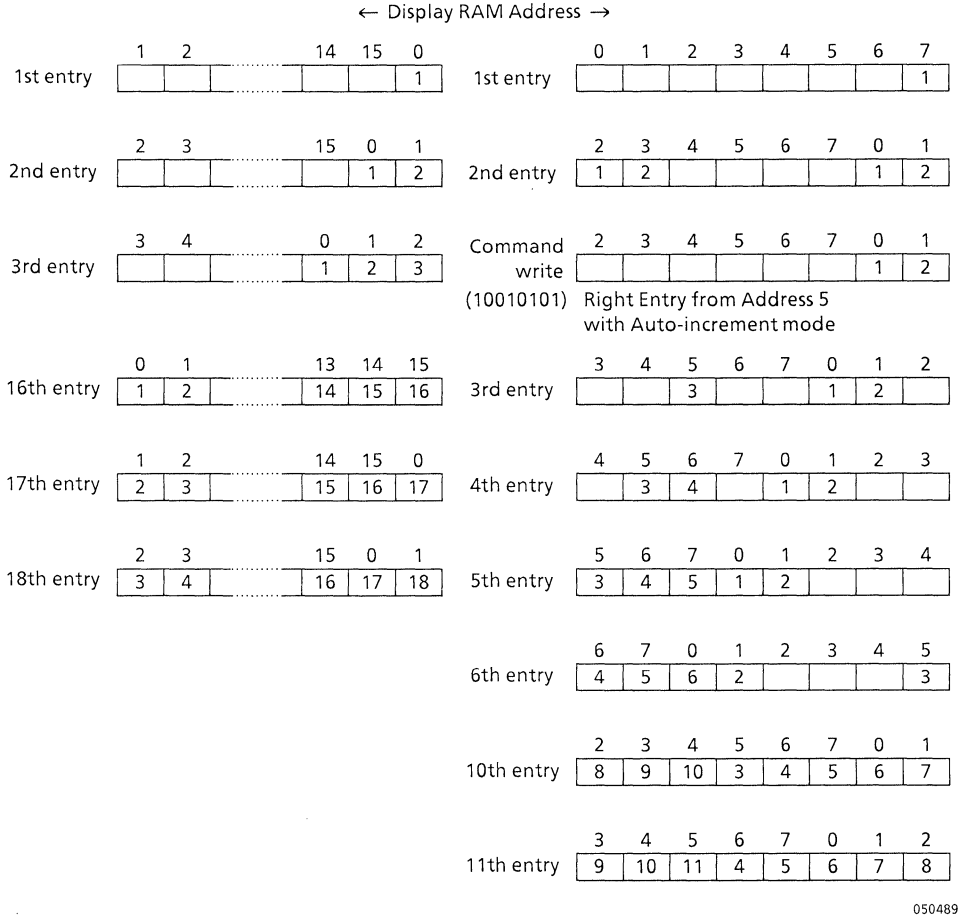
In Left Entry mode, address 0 of the Display RAM is the left-most side of the display and address 15 (address 7 in the case of 8-character display) is the right-most side. When characters are inputted onto the Display RAM with the auto-increment mode from address 0 of the display RAM, Characters are filled from the left-most position of the display. The 17th (or 9th) character is placed in the left-most position again. Address of the display RAM corresponds directly to each display position of the display, and so its position does not change every entry.



050489

10.2 RIGHT ENTRY

In Right Entry, the first entry if from the right-most position. Address of the Display RAM does not correspond to the display position.



050489

11. ELECTRIC CHARACTERISTICS

11.1 ABSOLUTE MAXIMUM RATINGS

SYMBOL	ITEM	RATING	UNIT
V_{CC}	V_{CC} Supply Voltage (with respect to GND (V_{SS}))	− 0.5 to + 7.0	V
V_{IN}	Input Voltage (with respect to GND (V_{SS}))	− 0.5 to + 7.0	V
V_{OUT}	Output Voltage (with respect to GND (V_{SS}))	− 0.5 to + 7.0	V
P_D	Power Dissipation	1	W
T_{sol}	Soldering Temperature (soldering time 10 sec)	260	°C
T_{stg}	Storage Temperature	− 65 to + 150	°C
T_{opr}	Operating Temperature	0 to 70	°C

050489

11.2 D.C. ELECTRICAL CHARACTERISTICS (Ta = 0 to 70°C, V_{CC} = 5V ± 10%, V_{SS} = 0V)

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{IL1}	Input Low Voltage (RL_0~RL_7)		− 0.5		1.4	V
V_{IL2}	Input Low Voltage (Others)		− 0.5		0.8	V
V_{IH1}	Input High Voltage (RL_0~RL_7)		2.2			V
V_{IH2}	Input High Voltage (Others)		2.0			V
V_{OL}	Output Low Voltage	I_{OL} = 2.2mA			0.45	V
V_{OH1}	Output High Voltage (IRQ)	I_{OH} = − 100μA	3.5			V
V_{OH2}	Output High Voltage (Others)	I_{OH} = − 400μA	2.4			V
I_{IL1}	Input Leak Current	$V_{IN} = V_{CC}$			+ 10	μA
	(SHIFT, CNTL, RL_0~RL_7)	V_{IN} = 0V			− 100	
I_{IL2}	Input Leak Current (Others)	$0V \leqq V_{IN} \leqq V_{CC}$			± 10	μA
I_{OFL}	Output Leak Current	$0.45V \leqq V_{OUT} \leqq V_{CC}$			± 10	μA
I_{CC}	Supply Current				120	mA

050489

11.3 INPUT CAPACITY

SYMBOL	PARAMETER	CONDITION	MIN.	TYP.	MAX.	UNIT
C_{IN}	Input Capacity	f_C = 1MHz Unmeasured Pins returned to V_{SS}.		5	10	pF
C_{OUT}	Output Capacity			10	20	pF

050489

11.4 A.C. ELECTRICAL CHARACTERISTICS (Ta = 0 to 70°C, V_{CC} = 5.0V ± 10%, V_{SS} = 0V)

SYMBOL	PARAMETER	TEST CONDITIONS	MIN.	TYP.	MAX.	UNIT
t_{AR}	Address Set up Time ($\overline{RD}\downarrow$)		0			ns
t_{RA}	Address Set up Time ($\overline{RD}\uparrow$)		0			ns
t_{RR}	$\overline{RD}$ Pulse Width		250			ns
t_{RD}	Valid Data ($\overline{RD}\uparrow$)	C_L = 150pF			150	ns
t_{AD}	Address to Valid Data	C_L = 150pF			250	ns
t_{DF}	Data Floating ($\overline{RD}\uparrow$)		10		100	ns
t_{RCY}	Read Cycle Time		1			μs
t_{AW}	Address Set up Time ($\overline{WR}\downarrow$)		0			ns
t_{WA}	Address Hold Time ($\overline{WR}\uparrow$)		0			ns
t_{WW}	$\overline{WR}$ Pulse Width		250			ns
t_{DW}	Data Set up Time ($\overline{WR}\uparrow$)		150			ns
t_{WD}	Data Hold Time ($\overline{WR}\uparrow$)		0			ns
t_{OW}	CLK Pulse Width		120			ns
t_{CY}	Clock period		320			ns

050489

12. TIMING DIAGRAM

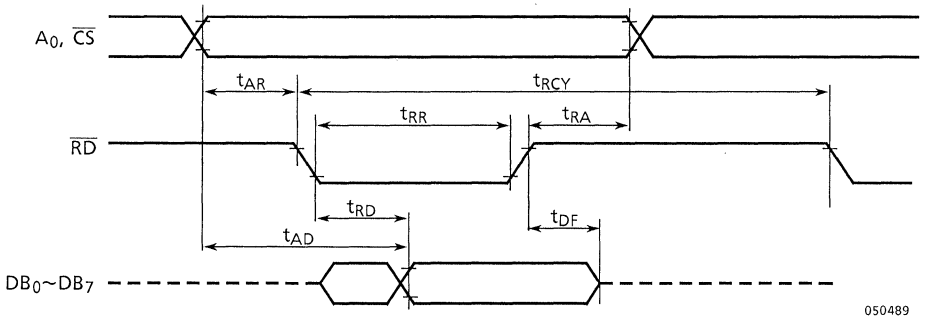


Figure 12.1 Read Timing

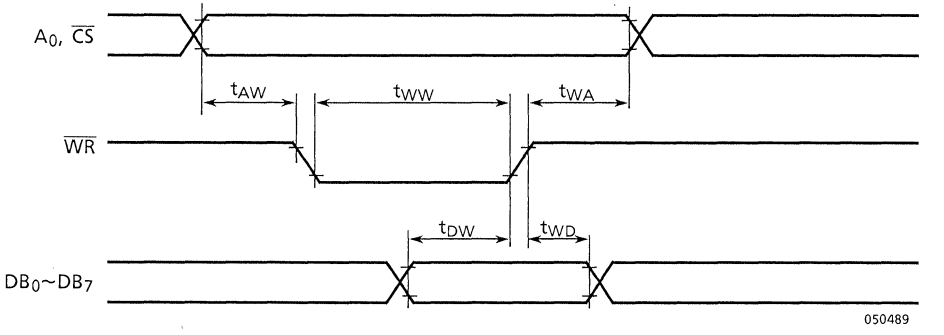


Figure 12.2 Write Timing

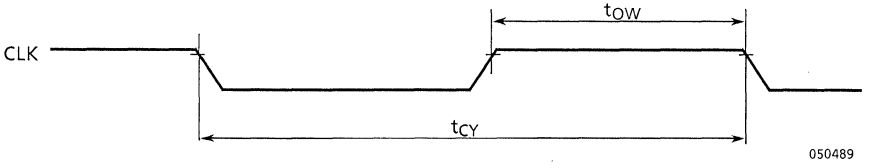


Figure 12.3 Clock Timing

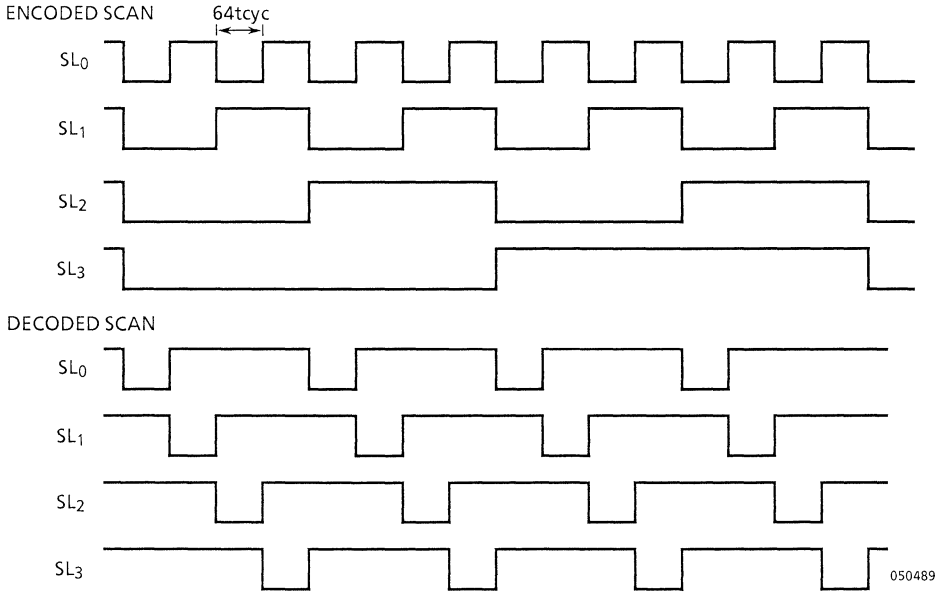
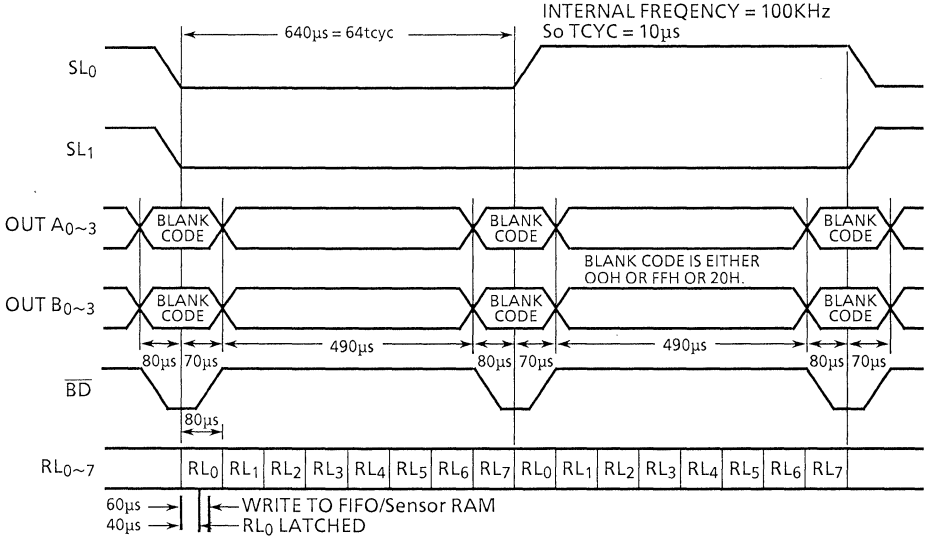


Figure 12.4 Scan Timing



Note : Shown is encoded scan, left entry SL2, SL3 are SL1 divided by 2 and 4. 050489

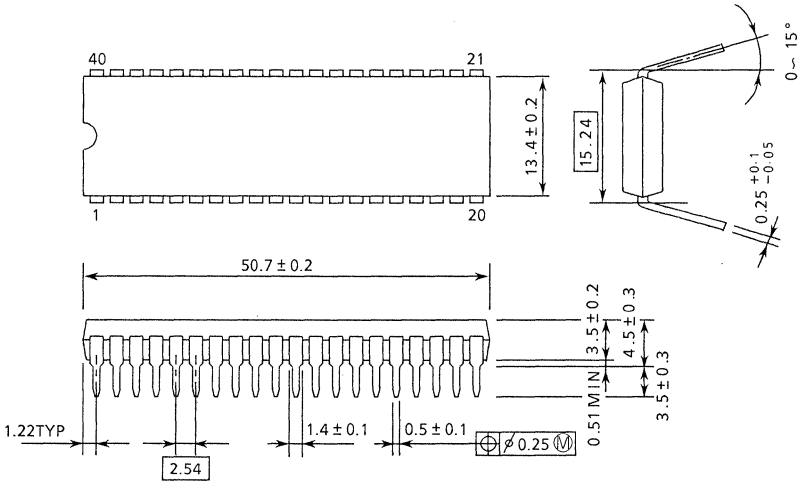
Figure 12.5 Example of Display

13. EXTERNAL DIMENSION

13.1 40 PIN DIP EXTERNAL DIMENSION

DIP40-P-600

Unit : mm



050489

Note: Each lead pitch is 2.54mm, and all the leads are located within ±0.25mm from their theoritical positions with respect to No.1 and No.40 leads.

14. EXAMPLE OF APPLICATION CIRCUIT

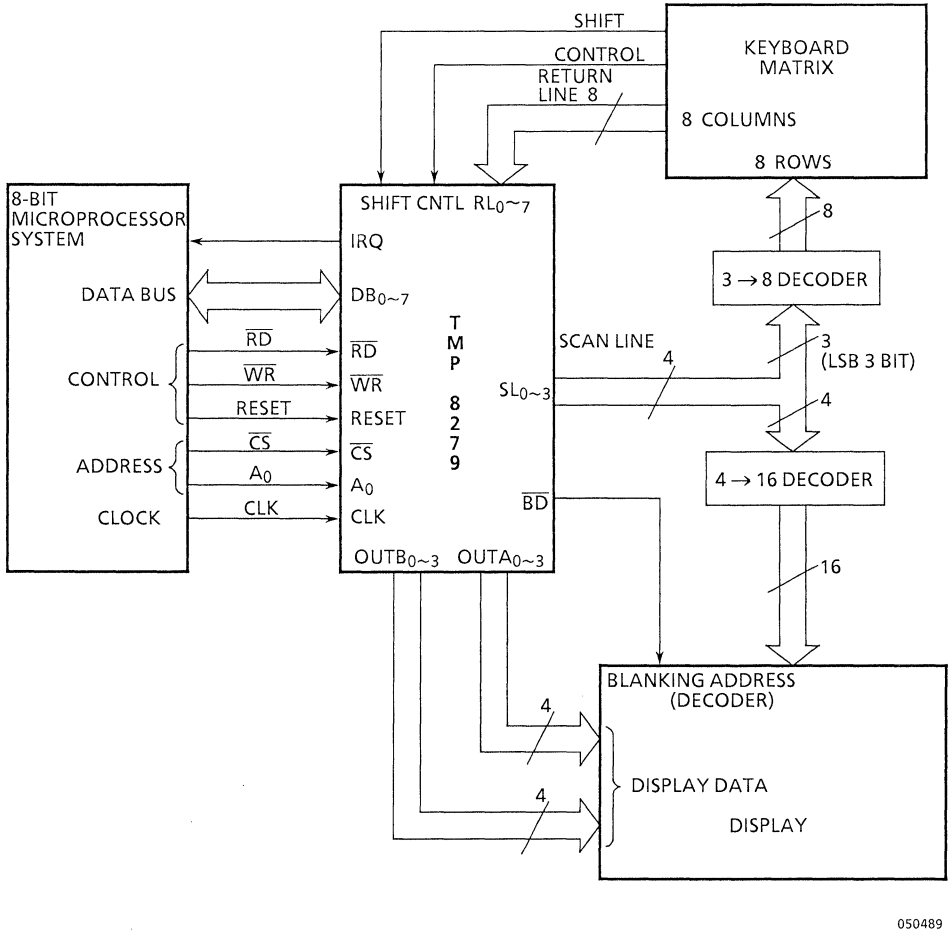


Figure 14.1 Example of Application Circuit

CMOS PROGRAMMABLE PERIPHERAL INTERFACE

TMP82C255AN-2 / TMP82C265AF-2
TMP82C255AN-10 / TMP82C265AF-10

1. GENERAL DESCRIPTION AND FEATURES

The TMP82C255A/TMP82C265A is a CMOS high speed programmable input/output interface with six 8-bit I/O ports. The function is almost equivalent to TMP82C55A×2. The TMP82C265A has the function that ports state are selectable immediately after reset by hardware either in input state or output state.
The TMP82C255A has not this function.

The TMP82C255A/TMP82C265A is fabricated using Toshiba's CMOS Silicon Gate Technology.

(1) 48 programmable I/O terminals (8 bit × 6 ports)

(2) The ports of TMP82C265A are capable to set to output after reset by hardware.

(3) High Speed Version (TRD = 100ns MAX : TMP82C255AN-10/TMP82C265AF-10)

(4) Low power consumption 3mA. Typ.
10μA. MAX. (@5V stand-by)

(5) Three operation modes (Mode 0, Mode 1, Mode 2)

(6) Bit set/reset capability (PC_{00} to PC_{07}, PC_{10} to PC_{17})

(7) All ports are capable of driving darlington transistors
−2.5mA. Typ. @VEXT = 1.5V, REXT = 1.1kΩ

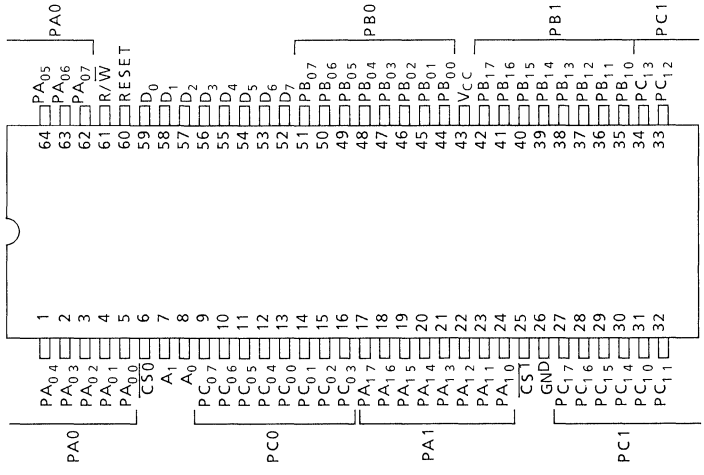
(8) Extended operating temperature : −40°C to +85°C

(9) Two packages

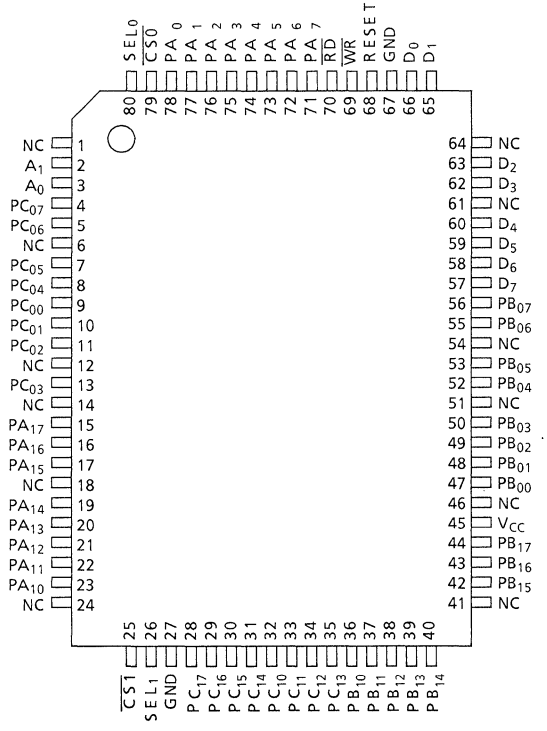
TMP82C255AN	64 PIN Shrink DIP
TMP82C265AF	80 PIN Mini Flat Package

2. PIN CONNECTIONS (TOP VIEW)

TMP82C255AN-2/TMP82C255AN-10



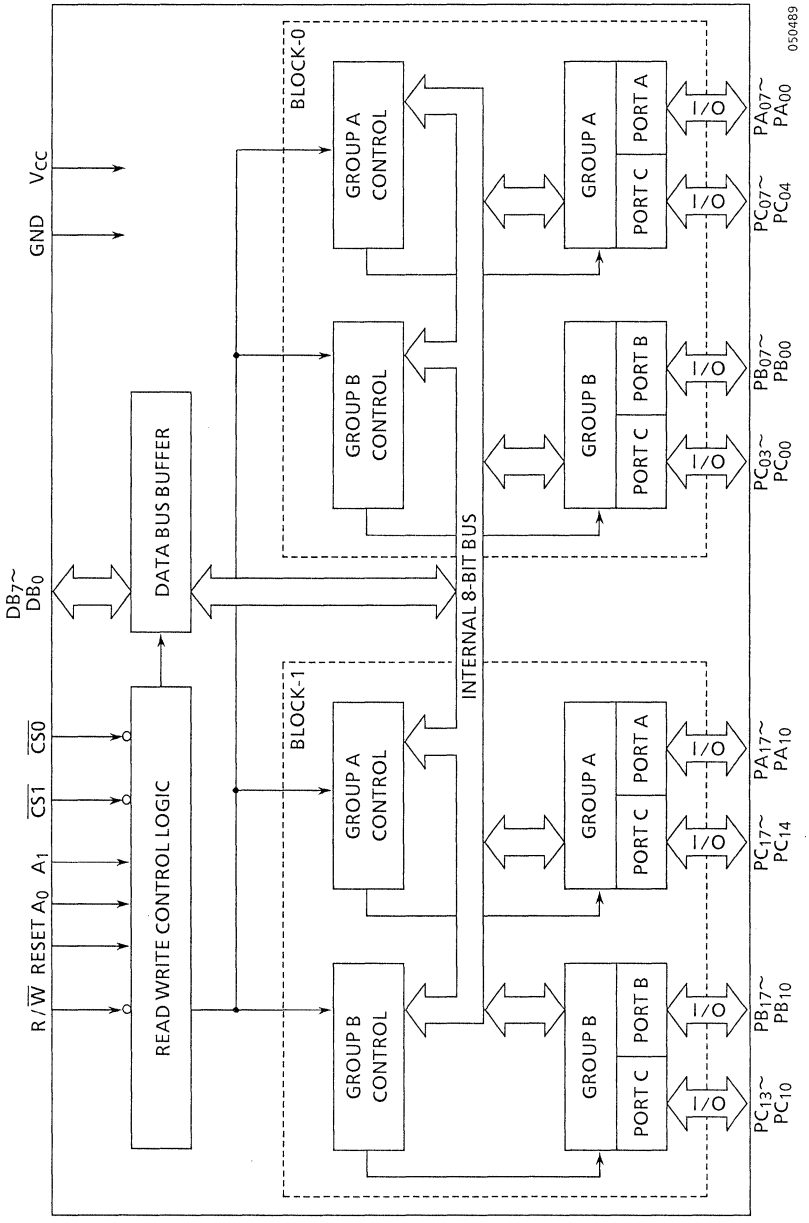
TMP82C265AF-2/TMP82C265AF-10



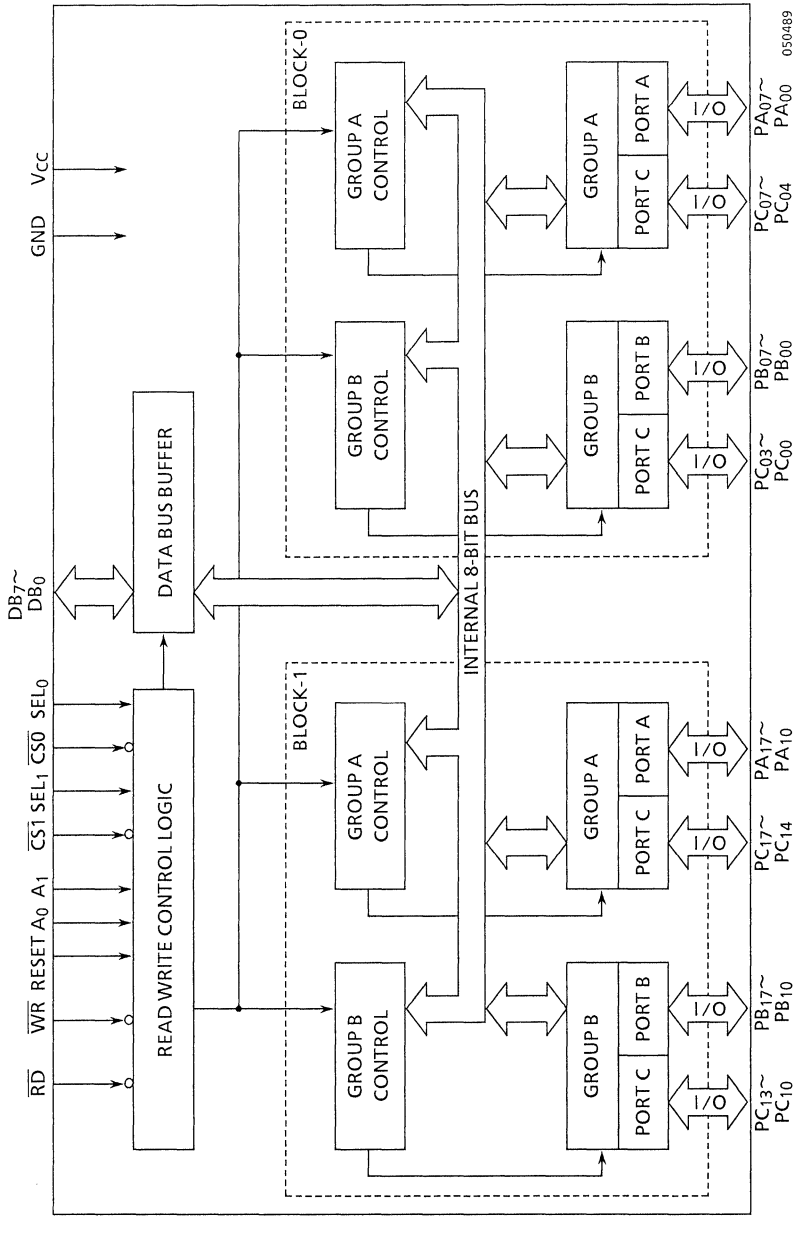
050489

3. BLOCK DIAGRAM

TMP82C255A



TMP82C265A


050489
GND
V_{CC}
BLOCK-0
BLOCK-1
GROUP A CONTROL
GROUP B CONTROL
GROUP A
GROUP B
PORT A
PORT B
PORT C
I/O
DATA BUS BUFFER
INTERNAL 8-BIT BUS
READ WRITE CONTROL LOGIC
DB_7~ DB_0
$\overline{RD}$ $\overline{WR}$ RESET A_0 A_1 $\overline{CS1}$ SEL_1 $\overline{CS0}$ SEL_0
PA_{07}~ PA_{00}
PC_{07}~ PC_{04}
PB_{07}~ PB_{00}
PC_{03}~ PC_{00}
PA_{17}~ PA_{10}
PC_{17}~ PC_{14}
PB_{17}~ PB_{10}
PC_{13}~ PC_{10}

4. PIN NAMES AND PIN FUNCTIONS

Pin Name	Number of Pin	Input/Output 3-state	Function
D_7~D_0	8	I/O 3-state	3-state bidirectional 8-bit data bus. Used for data transfer with MPU. Also, used for transfer of control words to this device and status information from this device.
PA_{07}~PA_{00}	8	I/O 3-state	There are two 3-state 8-bit I/O Port named PA0 and PA1. Operation mode and input/output configuration are defined by software. Port A contains the output latch buffer and input latch.
PA_{17}~PA_{10}	8		
PB_{07}~PB_{00}	8	I/O 3-state	There are two 3-state 8-bit I/O Port named PB0 and PB1. Operation mode and input/output configuration are defined by software. Port A contains the output latch buffer and input latch.
PB_{17}~PB_{10}	8		
PC_{07}~PC_{00}	8	I/O 3-state	There are two 3-state 8-bit I/O Port named PC0 and PC1. Operation mode and input/output configuration are defined by software. Port C can be divided into two 4-bit ports by the mode control and also, used as the control signal for Port A and Port B. In this case, each 3 bits of PC_{00} to PC_{02} and PC_{10} to PC_{12} are used for Port B control and each 5 bits of PC_{03} to PC_{07} and PC_{13} to PC_{17} for Port A control.
PC_{17}~PC_{10}	8		
$\overline{CS0}$	1	Input	Chip select input. When the terminal $\overline{CS0}$ is at "L" level, data transfer between BLOCK-0 and MPU is possible. And when the terminal $\overline{CS1}$ is at "L" level, data transfer between BLOCK-1 and MPU is possible. At "H" level, the data bus is placed in the high impedance state and control from the processor is ignored.
$\overline{CS1}$	1		
A_0, A_1	2	Input	Used for selecting Port A, B, C and the control registers. Normally, this terminal is connected to low order 2 bits of the address bus.
$\overline{RD}$	1	Input	Read signal. Only the TMP82C265A has this terminal. When this terminal is at "L" level, data or status information in this device is transferred to MPU.
$\overline{WR}$	1	Input	Write signal. Only the TMP82C265A has this terminal. When this terminal is at "L" level, data or control word is written into this device from MPU.
R/$\overline{W}$	1	Input	READ/WRITE signal. Only TMP82C255A has this terminal. When the terminal $\overline{CS0}$ or $\overline{CS1}$ is at "L" level, if this terminal is at "H" level, data or status information in this device is transferred to MPU. If this terminal is at "L" level, data or control word is written into this device from MPU.
SEL_0	1	Input	Only the TMP82C265A has these terminals. When SEL_0 or SEL_1 is a "H" level, the ports are set to input state after reset, when it is a "L" level, the ports are set to output state after reset. If the terminals is at "L" level, the function is only capable to set to output state in mode 0. BLOCK-0 ports of block diagram is controled by SEL_0. BLOCK-1 ports of block diagram is controled by SEL_1.
SEL_1	1		
RESET	1	Input	When this terminal is at "H" level, all internal registeres including the control register are cleared, and the function turn out input state in mode 0 for the TMP82C55A and TMP82C265A with SEL_0 or SEL_1 "H" level. THe function turn out output state in mode 0 for the TMP82C265A with SEL_0 or SEL_1 "L" level. Each BLOCK of TMP82C265A is capable to set to different mode by SEL_0 and SEL_1.

050489

Pin Name	Number of Pin	Input/Output 3-state	Function
V_{CC}	1	Power supply	5V
V_{SS}	1	Power supply	GND

050489

5. DESCRIPTION OF BASIC OPERATION

5.1 TMP82C255A FUNCTIONS AND FEATURES

The TMP82C255A is a programmable peripheral interface consisting of Block-0 and Block-1 each of which has 3 sets, total 6 sets of 8-bit ports (PA0, PB0, PC0, PA1, PB1 and PC1) each of which has 2 built-in control registers.

Block-0 and Block-1 have the independent chip select input terminals $\overline{CS0}$ and $\overline{CS1}$, respectivery. Data buses D_7-D_0, address input terminals A_1 and A_0, RESET input terminals, and R/$\overline{W}$ terminal are commonly used by Block-0 and Block-1.

Read and Write functions are controlled by one R/$\overline{W}$ terminal and therefore, $\overline{RD}$ and $\overline{WR}$ signals and address must be synthesized and connected to the R/$\overline{W}$, $\overline{CS0}$ and $\overline{CS1}$ terminals.

Total 24 ports input/output terminals each of Block-0 and Block-1 are divided into 12-bit group A and Group B. Group A consist of upper 4 bits of Ports A and Port C, while Group B consists of lower 4 bits of Port B and Port C. Each group is programmable independently by control word from MPU. Three operation modes are available; Mode 0, Mode 1 and Mode 2. In mode 0, it is programmable to use 28-bit input/output ports as the input or output port independently. In Mode 1, the input/output ports are divided into group A and Group B. In each group, 8 bits are used for the input or output port and the remaining 4 bits are used as the control signal. Mode 2 is applicable only to Group A and 12 bits are used for 8-bit two-way bus and 5-bit control signal. Further, when port C is used as the output port, any bit of port C can be set/reset. There are two control registers; one is used for mode setting and the other is used for bit seting/resetting. The control register cannot be read. Further, when the reset input becomes "H" level, the input/output internal registers are all reset and all the input/output terminals are placed in Mode 0 and input Mode (high impedance state).

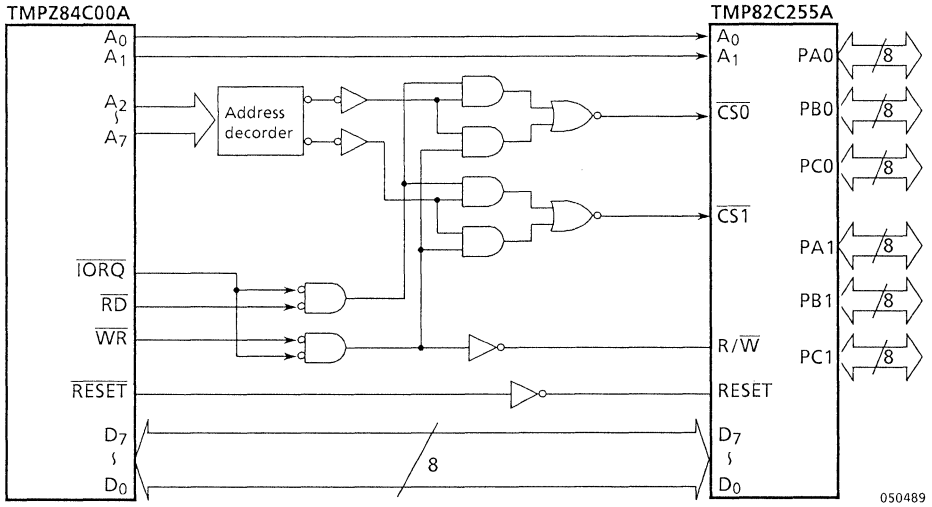


Figure 5.1 Connecting example of TMPZ84C00A and TMP82C255A

In case of the TMP82C255, the read operation is controlled by the terminals $\overline{CS0}$ or $\overline{CS1}$ when the R/$\overline{W}$ terminals is at "H" level and therefore, there is no AC electrical characteristics for rise and fall of the $\overline{RD}$ terminal but anf fall of the $\overline{CS0}$ or $\overline{CS1}$ terminal are controlled.

The write opertion is controlled by the terminal $\overline{CS0}$ or $\overline{CS1}$ when the R/$\overline{W}$ is at "L" level. Further, the maximum control of the setup time TWC of the $\overline{CS0}$ or $\overline{CS1}$ for fall of R/$\overline{W}$ terminal and the hold time T_{WC} of the $\overline{CS0}$ or $\overline{CS1}$ for the rise of R/$\overline{W}$ terminal is added. This is because the $\overline{CS0}$ or $\overline{CS1}$ terminal controls the read operation when the R/$\overline{W}$ is at "H" level.

5.2 TMP82C255A BASIC OPERATION

Basic operation of the TMP82C255A for BLOCK-0. PC0 are as shown in Table 5.1.

Table 5.1 Basic Operation for Block-0 of the TMP82C255A

$\overline{CS0}$	RESET	A_1	A_0	R/$\overline{W}$	Function
×	1	×	×	×	Set to all ports (six ports) input state in mode 0. Ports terminals are in High Impedance state.
0	0	0	0	1	Data Bus ← PA0
0	0	0	1	1	Data Bus ← PB0
0	0	1	0	1	Data Bus ← PC0
0	0	0	0	0	PA_0 ← Data Bus
0	0	0	1	0	PB_0 ← Data Bus
0	0	1	0	0	PC_0 ← Data Bus
0	0	1	1	0	Control register for BLOCK-0 ← Data Bus
0	0	1	1	1	Inhibition of combination
1	0	×	×	×	Non active

050489

Basic Operation for BLOCK-1 of the TMP82C255A is equal to the one shown in Table 5.1. BLOCK-1 is selected when $\overline{CS1} = 0$.

5.3 TMP82C265A FUNCTION AND FEATURES

The TMP82C265A can be set to general mode when the terminal SEL_0 or SEL_1 is fixed at "H" level and to the output mode when it is fixed at "L" level.

In the general mode, the TMP82C265A has the functions similar to those of the TMP82C55A but for the $\overline{WR}$ and $\overline{RD}$ functions, the independent $\overline{RD}$ and $\overline{WR}$ terminals are provided as in the former TMP82C255A unlike R/$\overline{W}$ 1 terminal in the TMP82C255A. Therefore, the TMP82C265A operates when connected in the same connecting method as before.

In the output mode, all ports of the TMP82C265A are set in "L" output state in mode 0 after reset. Further, even when the write operation is made in any mode, the ports are set in output state in mode 0, and "L" level signals are output from the ports.

5.4 TMP82C265A BASIC OPERATION

Basic operation of the TMP82C265A in the general mode (the terminal SEL_0-SEL_1 is "L" level) is equal to that of the TMP82C255A.

Basic operation for BLOCK-0 of the TMP82C265A in the output only mode (SEL_0: "L" level) is as shown in Table 5.2.

Table 5.2 Output Mode Basic Operation for Block-0 of the TMP82C265A

SEL_0	$\overline{CS0}$	RESET	D_7	A_1	A_0	$\overline{RD}$	$\overline{WR}$	Function
0	×	1	×	×	×	×	×	Set to PA0, PB0 and PC0 ports "L" level output state in mode 0
0	0	0	1	1	1	1	0	PA0, PB0 and PC0 ports "L" level output state in mode 0 set command
×	0	0	0	1	1	1	0	Bit set/reset control
△	0	0	×	0	0	1	0	PA0 ← Data Bus
△	0	0	×	0	1	1	0	PB0 ← Data Bus
△	0	0	×	1	0	1	0	PC0 ← Data Bus
△	0	0	×	0	0	0	1	Data Bus ← PA0 (Read the terminal state)
△	0	0	×	0	1	0	1	Data Bus ← PB0 (Read internal output latch)
△	0	0	×	1	0	0	1	Data Bus ← PC0 (Read internal output latch)

050489

△ : The output only mode is maintained when SEL_0 is either in "L" or "H" level if set to the output state.
However, if the RESET terminal is set to "H" level or mode is written when SEL_0 is at "H" level, the output mode is changed to the general mode.

Basic operation of the TMP82C265A for Block-1 is equal to that shown in Table 5.2 but the control is made by the terminal SEL_1 and $\overline{CS1}$.

6. DETAILED OPERATIONAL DESCRIPTION

The operation of the TMP82C255A and TMP82C265A in the general made is described in detail. Further, the operation is described for Block-0. Block-1 has the same functions as those of Block-0 and therefore, it is omitted.

The operation of the TMP82C265A in the output mode is considered to be a special case of the description for the general mode provided below, in which the TMP82C265A has been set in the output state in mode 0 from immediately after reset irrespective of command input and therefore, the description is omitted here.

6.1 MODE SELECTION

There are three basic modes of operation that can be selected by control words.

Mode 0 - Basic I/O (Group A, Group B)

Mode 1 - Strobe input/Strobe output (Group A, Group B)

Mode 2 - Two-way bus(Port A only)

Operation modes for Group A and Group B can be independently defined by the control word from the MPU. If D_7 is set to "1" in writing a control word into the PPI, on operation mode is selected, while of D_7 = "0" , the set/reset function for Port C is selected.

6.1.1 Control Word to Define Operation Mode

Figure 6.1 shows the control words to define operation mode of the TMP82C255A/TMP82C265A.

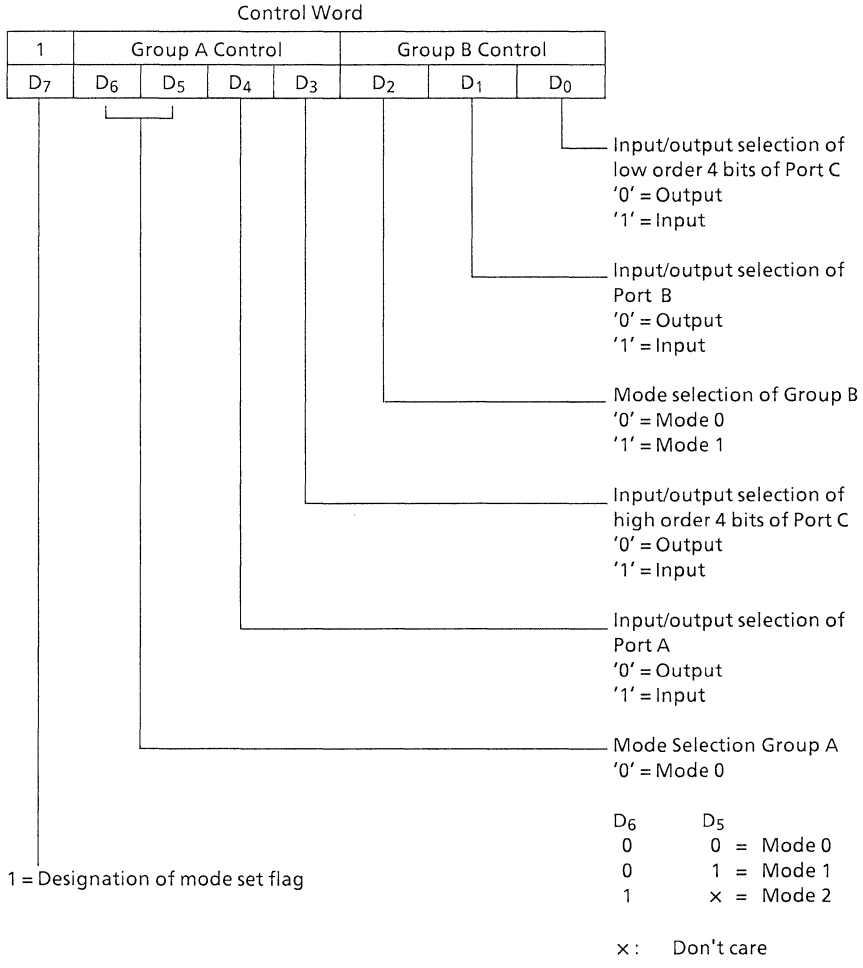


Figure 6.1 Control Word for Mode Selection

6.1.2 Port C Bit Set/Reset Control Word

Any bit of 8 bits of Port C can be set/reset by Port C bit set/reset control word. Figure 6.2 shows the Port C bit set/reset control word.

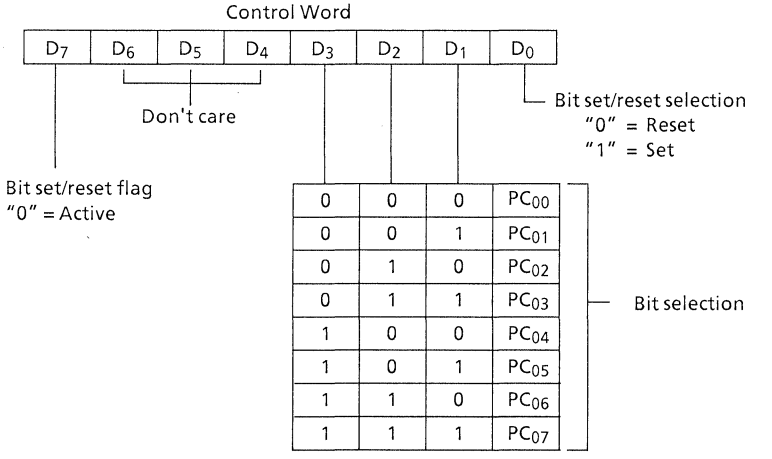


Figure 6.2 Control Word for Bit Set/Reset

6.2 OPERATION MODES

6.2.1 Mode 0 (Basic I/O)

This functional configuration is used for simple input or output operations. No 'handshaking' is required and data is simply written to or read from a specified part. Output data to the ports from MPU are latched out but input data from the ports are not latched.

In Mode 0, 24 I/O terminals are divided into four groups of Port A (8 bits) , Port B (8 bits) , high order 4 bits of Port C and low order 4 bits of Port C. Each port can be programmed to be input or output. The configuration of each port are determined according to the contents of Bit 4 (D_4) , 3 (D_3) , 1 (D_1) and 0 (D_0) of the control word for mode selection.

The I/O configuration of each port in Mode 0 are shown in Table 6.2.

Table 6.2 Port definition in Mode 0

Mode Setting Control Word				Port A	Port C (PC_{07}~PC_{04})	Port B	Port C (PC_{03}~PC_{00})
D_4	D_3	D_1	D_0				
0	0	0	0	Out	Out	Out	Out
0	0	0	1	Out	Out	Out	In
0	0	1	0	Out	Out	In	Out
0	0	1	1	Out	Out	In	In
0	1	0	0	Out	In	Out	Out
0	1	0	1	Out	In	Out	In
0	1	1	0	Out	In	In	Out
0	1	1	1	Out	In	In	In
1	0	0	0	In	Out	Out	Out
1	0	0	1	In	Out	Out	In
1	0	1	0	In	Out	In	Out
1	0	1	1	In	Out	In	In
1	1	0	0	In	In	Out	Out
1	1	0	1	In	In	Out	In
1	1	1	0	In	In	In	Out
1	1	1	1	In	In	In	In

050489

6.2.2 Mode 1 (Strobe I/O)

In Mode 1, input/output of port data is performed in conjunction with the srtobe signals or 'handshaking' signals. Port C is used to control Port A or Port B.

The basic operatings in Mode 1 are as follows:

- Mode 1 can be set for two groups of Group A and Group B.
- Each group consist of 8-bit data port and 4-bit control/data port.
- The 8-bit data port can be set as input or output port.
- The control/data port is used as control or status of the 8-bit data port.

(1) When used as the input port in Mode 1 :

- $\overline{STB}$ (Strobe Input)

  At "0", input data is loaded in the internal input latch in the port.
  In this case, a control signal from MPU is not concerned and data is input from the port any time. This data is not read out on the data bus unless MPU executes an input instruction.

- IBF (Input Buffer Full F/F Output)

  When data is loaded in the internal input latch from the port, this output is set to "1". IBF is set ("1") by $\overline{STB}$ input being reset and is reset ("0") by the rising edge of $\overline{RD}$ input.

- INTR (Interrupt Request Output)

  Used for the interrupt process of data loaded in the internal input latch. When $\overline{STB}$ input is at "0" if INTE (INTE flag) in the PPI is in the enabled state ("1"), IBF is set to "1". INTR is set to "1" immediately after the rising edge of this $\overline{STB}$ input and reset to "0" by the falling edge of $\overline{RD}$ input.

  The INTE flags of Group A and Group B are controlled as follows:

  INTEA – Control by bit set/reset of PC_{04}

  INTEB – Control by bit set/reset of PC_{02}

(2) When used as the output port in Mode 1:

- $\overline{OBF}$ (Output Buffer Full F/F Output)

  This is a flag which shows that MPU has written data into a specified port. $\overline{OBF}$ is set to becomes "0" at the rising edge of $\overline{WR}$ signal and is set to "1" at the falling edge of $\overline{ACK}$ (Acknowledge input) signal.

- $\overline{ACK}$ (Acknowledge Input)

  $\overline{ACK}$ signal is sent to the PP1 as a response from a peripheral device that received data from the port.

- INTR (Interrupt Request Output)

  When a peripheral device received data from MPU, INTR is set to "1" and the interrupt is requested to MPU. If $\overline{ACK}$ signal is received when INTE flag is in the enable state, $\overline{OBF}$ is set to "1" and INTR signal becomes "1" immediately after the rising edge of $\overline{ACK}$ signal. Further, INTR is reset at the falling edge of $\overline{WR}$ signal when data is written into the PPI by MPU.

  The INTE flags of Group A and Group B are controlled as follows:

  INTEA – Control by bit set/reset of PC_{06}

  INTEB – Control by bit set/reset of PC_{02}

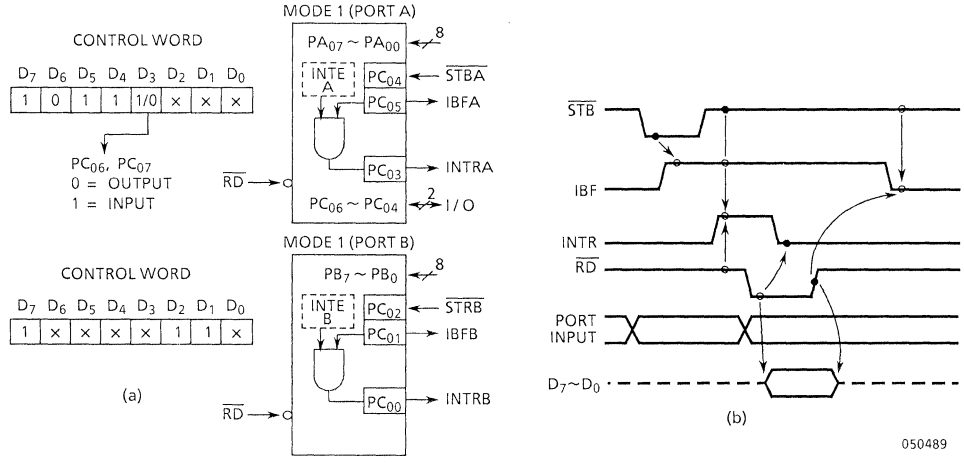


Figure 6.3 Example of Strobe Input in Mode 1

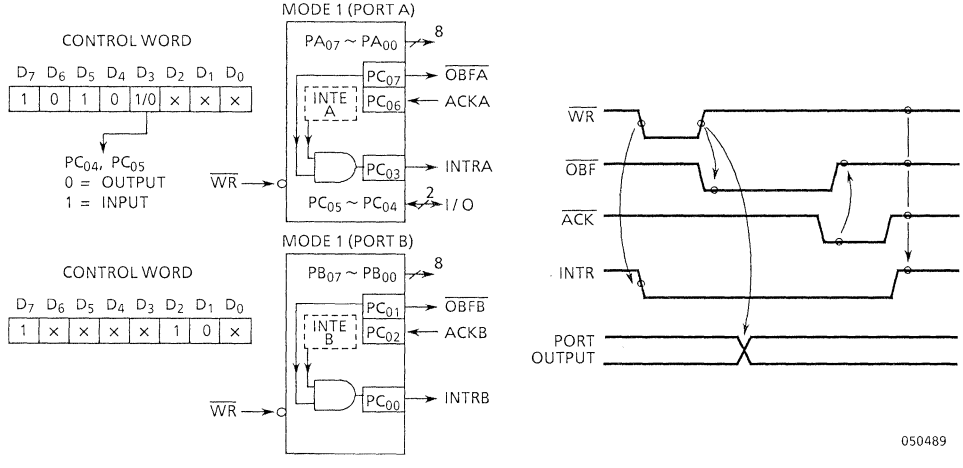


Figure 6.4 Example of Strobe Output in Mode 1

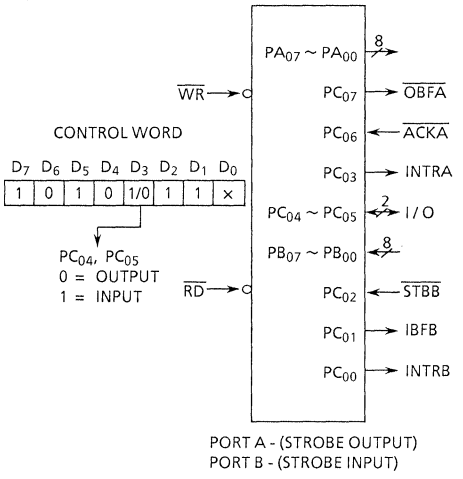
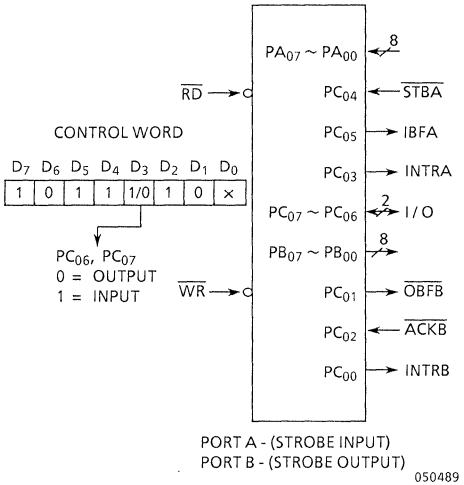


Figure 6.5 Example of Port A Output, port B Input in Mode 1



050489

Figure 6.6 Example of Port A Input, Port B Output in Mode 1

6.2.3 Mode 2 (Strobed Bidirectional Bus I/O)

In this mode, Port A is used as 8 bits bidirectional bus for data transfer with a peripheral device. This mode is applicable only to Group A, which consists of an 8-bit bidirectional bus (Port A 8-bit) and 5-bit control signals (high order 5 bits of Port C). The bidirectional bus (Port A) has both the internal input and output registers. When group A is set in Mode 2, Group B can be set independently. There are 5 control signals as follows when Group A is used in Mode 2.

- OBF (Output buffer Full F/F Output)

  When MPU writes data into of Port A, $\overline{OBF}$ is set to "0" to inform a peripheral device that the PPI is ready to output data. However, Port A is kept in the floating (high impedance) state until $\overline{ACK}$ input signal is received.

- $\overline{ACK}$ (Acknowledge Input)

  When $\overline{ACK}$ signal is set to "0", the data of the 3-state output buffer of Port A is send out. If $\overline{ACK}$ signal is at "1", Port A is in the high impedance state.

- $\overline{STB}$ (Strobe Input)

  When $\overline{STB}$ input is set to "0", the data from peripheral devices are held in the input latch. When the active $\overline{RD}$ signal is input into the PPI, the latched input data are output on the system data bus (D_7-D_0).

- IBF (Input Buffer Full F/F Output)

  When data from peripheral devices are held in the input latch, IBF is set to "1".

- INTR (Interrupt Request Output)

  INTR is the output to request the interrupt to MPU and its function is the same as that in Mode 1. There are two interrupt enable flip-flop (INTE), INTE1 corresponds to INTEA in Mode 1 output and INTE2 to INTEA in Mode 1 input.

  INTE 1 – Used to generate INTR signal in conjunction with $\overline{OBF}$ and $\overline{ACK}$ signals, and is controlled by PC_{06} bit set/reset.

  INTE2 – Used to generate INTR signal in conjunction with IBF and $\overline{STB}$ singals, and is coutrolled by PC_{04} bit set/reset.

Figure 6.7 shows the operating example and the timing diagram in Mode 2.

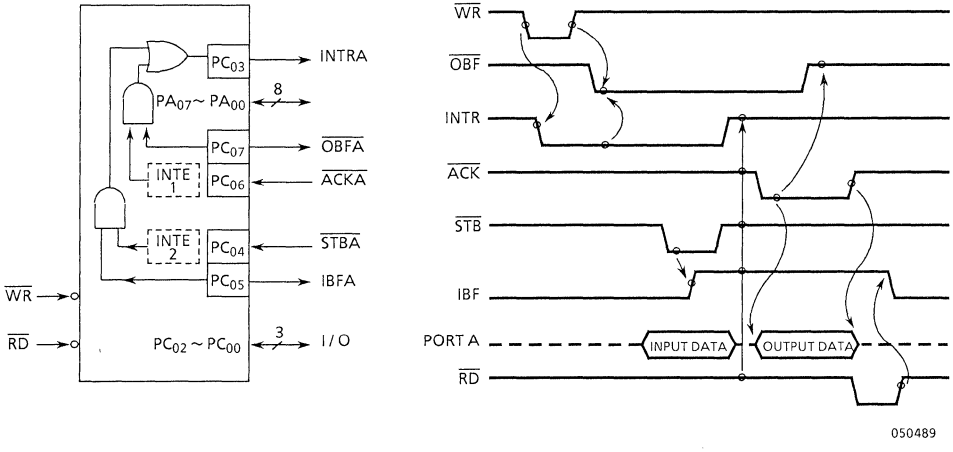


Figure 6.7 Oerating example in Mode 2

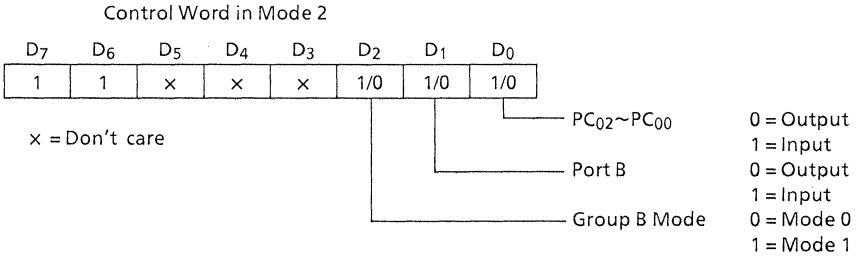


Figure 6.8 Control Word and Configuration in Mode 2

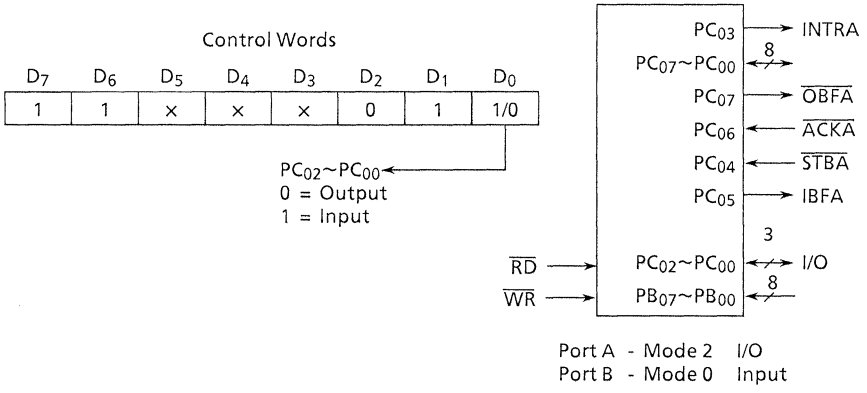


Figure 6.9 Example in Combination with Mode 2 and Other Mode

6.2.4 Precautions for use in Mode 1 and 2

When used in Mode 1 and 2, bits which are not used as control or status in Port C can be used as follow.

If porgrammed as the input, they are accessed by normal Port C read.

If porgrammed as the output, high order bits of Port C (PC_{07}-PC_{04}) are accessed using the bit set/reset function. As to low order bits of Port C (PC_{03}-PC_{00}), in additions to access by the bit set/reset function, 3 bits only can be accessed by normal writing.

6.3 READING PORT C STATUS

When Port C is used as the control port, that is, when Port C is used in Mode 1 or Mode 2, the status information of the control word can be read out by a normal read operation of Port C.

Table 6.3 Status Word Format of Port C

Data Mode	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Mode 1 Input	I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB
Mode 1 Output	$\overline{OBFA}$	INTEA	I/O	I/O	INTRA	INTEB	$\overline{OBFB}$	INTRB
Mode 2	$\overline{OBFA}$	INTE1	IBFA	INTE2	INTRA	By Group B Mode		

050489

7. ELECTRICAL CHARACTERISTICS

7.1 ABSOLUTE MAXIMUM RATINGS

Symbol	Item	Rating	Unit
V_{CC}	Supply Voltage	− 0.5~7.0	V
V_{IN}	Input Voltage	− 0.5~V_{CC} + 0.5	V
P_D	Power Dissipation	250	mW
T_{SOLDER}	Soldring Temperature (10sec)	260	°C
T_{STG}	Storage Temperature	− 65~ + 150	°C
T_{OPR}	Operating Temperature	− 40~ + 85	°C

050489

7.2 DC ELECTRICAL CHARACTERISTICS

TA = − 40°C to + 85°C, V_{CC} = 5V ± 10%, V_{SS} = 0V

Symbol	Item	Test Condition	Min.	Typ.	Max.	Unit
V_{IL}	Input Low Voltage		− 0.5	−	0.8	V
V_{IH}	Input High Voltage		2.2	−	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2.5mA	−	−	0.45	V
V_{OH1}	Output High Voltage	I_{OH} = − 400μA	2.4	−	−	V
V_{OH2}	Output High Voltage	I_{OH} = − 100μA	V_{CC} − 0.8	−	−	V
I_{IL}	Input Leak Current	$0 \leqq V_{IN} \leqq V_{CC}$	−	−	± 10	μA
I_{OL}	Output LeaK Current (High Impedance State)	$0 \leqq V_{OUT} \leqq V_{CC}$	−	−	± 10	μA
I_{DAR}	Darlington Drive Current	V_{EXT} = 1.5V R_{EXT} = 1.1kΩ	− 1.0	−	-5.0	mA
I_{CC1}	Operating Supply Current	I / O cycle Time 1μsec	−	3.0	5.0	mA
I_{CC2}	Stand-by Supply Current	$V_{IH} > V_{CC}$-0.2V $V_{IL} < 0.2V$ $\overline{CS} > V_{CC}$ − 0.2V	−	−	10	μA

050489

* Total current of all darlington drive ports must not exceed 60mA.

7.3 CAPACITANCE

$T_A = 25°C$, $V_{CC} = V_{SS} = 0V$

Symbol	Item	Test Condition	Min.	Typ.	Max.	Unit
C_{IN}	Input Capacitance	f = 1MHz (*)	—	—	10	pF
$C_{I/O}$	I/O Capacitance		—	—	20	pF

050489

(*): All terminals except that to be measured should be earthed.

7.4 AC ELECTRICAL CHARACTERISTICS (1/2)

$T_A = -40°C$ to $+85°C$, $V_{CC} = 5V \pm 10\%$, $V_{SS} = 0V$

Symbol	Parameter	A - 2		A - 10		Unit
		Min.	Max.	Min.	Max.	
t_{AR}	Address set-up time for $\overline{RD}$ fall	0	—	0	—	ns
t_{RA}	Address hold time for $\overline{RD}$ rise	0	—	0	—	ns
t_{RR}	$\overline{RD}$ pulse width	160	—	150	—	ns
t_{RD}	Delay from $\overline{RD}$ fall to decided data output	—	140	—	100	ns
t_{DF}	Time from $\overline{RD}$ rise to data bus floating	0	40	0	40	ns
t_{RV}	Time from $\overline{RD}$ or $\overline{WR}$ rise to next $\overline{RD}$ or $\overline{WR}$ fall	200	—	180	—	ns
t_{AW}	Address set-up time for $\overline{WR}$ fall	20	—	20	—	ns
t_{WA}	Address holding time for $\overline{WR}$ rise	0	—	0	—	ns
t_{WW}	$\overline{WR}$ pulse width	120	—	120	—	ns
t_{DW}	Bus data set-up time for $\overline{WR}$ rise	100	—	100	—	ns
t_{WD}	Bus data holding time for $\overline{WR}$ rise	20	—	20	—	ns
t_{WB}	Delay from $\overline{WR}$ rise to decided data output	—	350	—	350	ns
t_{IR}	Port data set-up time for $\overline{RD}$ fall	0	—	0	—	ns
t_{HR}	Port data holding time for $\overline{RD}$ rise	0	—	0	—	ns
t_{AK}	$\overline{ACK}$ pulse width	300	—	300	—	ns
t_{ST}	$\overline{STB}$ pulse width	350	—	350	—	ns
t_{PS}	Port data set-up time for $\overline{STB}$ rise	20	—	20	—	ns
t_{PH}	Port data holding time for $\overline{STB}$ rise	150	—	150	—	ns
t_{AD}	Delat from $\overline{ACK}$ fall to decided data output Delay	—	300	—	300	ns
t_{KD}	Time from $\overline{ACK}$ rise up to port (Port in Mode 2) floating	25	250	20	250	ns
t_{WOB}	Delay from $\overline{WR}$ rise to $\overline{OBF}$ fall	—	300	—	300	ns
t_{AOB}	Delay from $\overline{ACK}$ fall to $\overline{OBF}$ rise	—	350	—	350	ns
t_{SIB}	Delay from $\overline{STB}$ fall to $\overline{IBF}$ rise	—	300	—	300	ns

050489

AC ELECTRICAL CHARACTERISTICS (2/2)

Symbol	Parameter	A - 2		A - 10		Unit
		Min.	Max.	Min.	Max.	
t_{RIB}	Delay from $\overline{RD}$ fall to $\overline{IBF}$ rise	—	300	—	300	ns
t_{RIT}	Delay from $\overline{RD}$ fall to INTR fall	—	400	—	400	ns
t_{SIT}	Delay from $\overline{ACK}$ rise to INTR rise	—	300	—	300	ns
t_{AIT}	Delay from $\overline{ACK}$ rise to INTR rise	—	350	—	350	ns
t_{WIT}	Delay from $\overline{WR}$ rise to INTR fall	—	450	—	450	ns
t_{AC}	Address set-up time for $\overline{CSO}$, $\overline{CSI}$ fall	0	—	0	—	ns
t_{CA}	Address hold time for $\overline{CSO}$, $\overline{CSI}$ rise	0	—	0	—	ns
t_{CC}	$\overline{CSO}$, $\overline{CSI}$ pulse width	160	—	150	—	ns
t_{CD}	Delay from $\overline{CSO}$, $\overline{CSI}$ fall to decided data output	—	140	—	100	ns
t_{DFC}	Time from $\overline{CSO}$, $\overline{CSI}$ rise to data bus floating	0	40	0	40	ns
T_{CW}	$\overline{CSO}$, $\overline{CSI}$ set-up time for R/$\overline{W}$ fall	—	0	—	0	ns
T_{WC}	$\overline{CSO}$, $\overline{CSI}$ holding time for R/$\overline{W}$ rise	—	50	—	50	ns
t_{IC}	Port data set-up time for $\overline{CSO}$, $\overline{CSI}$ fall	0	—	0	—	ns
t_{HC}	Port data holding time for $\overline{CSO}$, $\overline{CSI}$ rise	0	—	0	—	ns
t_{CIB}	Delay from $\overline{CSO}$, $\overline{CSI}$ fall to $\overline{IBF}$ rise	—	300	—	300	ns
t_{CIT}	Dekay from $\overline{CSO}$, $\overline{CSI}$ fall to $\overline{INTR}$ fall Delay	—	400	—	400	ns
t_{RES}	RESET pulse width	500	—	500	—	ns

050489

Note 1 :

A-2 : TMP82C255AN-2, TMP82C265AF-2

A-10 : TMP82C255AN-10, TMP82C265AF-10

Note 2 :

Following AC specifications of TMP82C255A are applied to the logical AND timing between R/$\overline{W}$ terminal and $\overline{CS0}$ or $\overline{CS1}$ terminal.

Symbol			
T_{AW}	T_{WA}	T_{DW}	T_{WD}
T_{WB}	T_{CW}	T_{WC}	

050489

Note 3 :

AC Measuring Point Input Voltage $V_{IH}=2.4V$, $V_{IL}=0.45V$

Output Voltage $V_{OH}=2.2V$, $V_{OL}=0.8V$

$CL=150pF$.

8.1 TMP82C255A TIMING DIAGRAM (1)

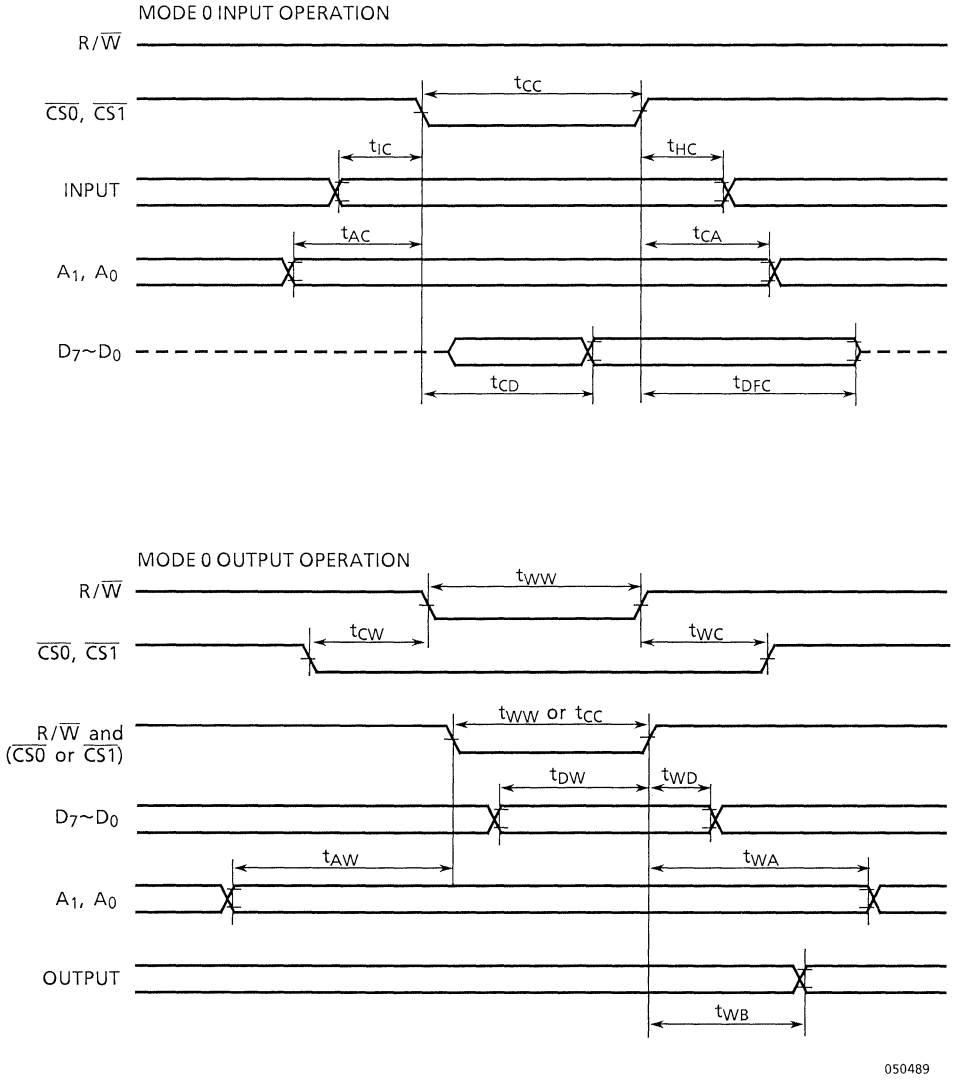


Figure 8.1 TMP82C255A Timing diagram (1)

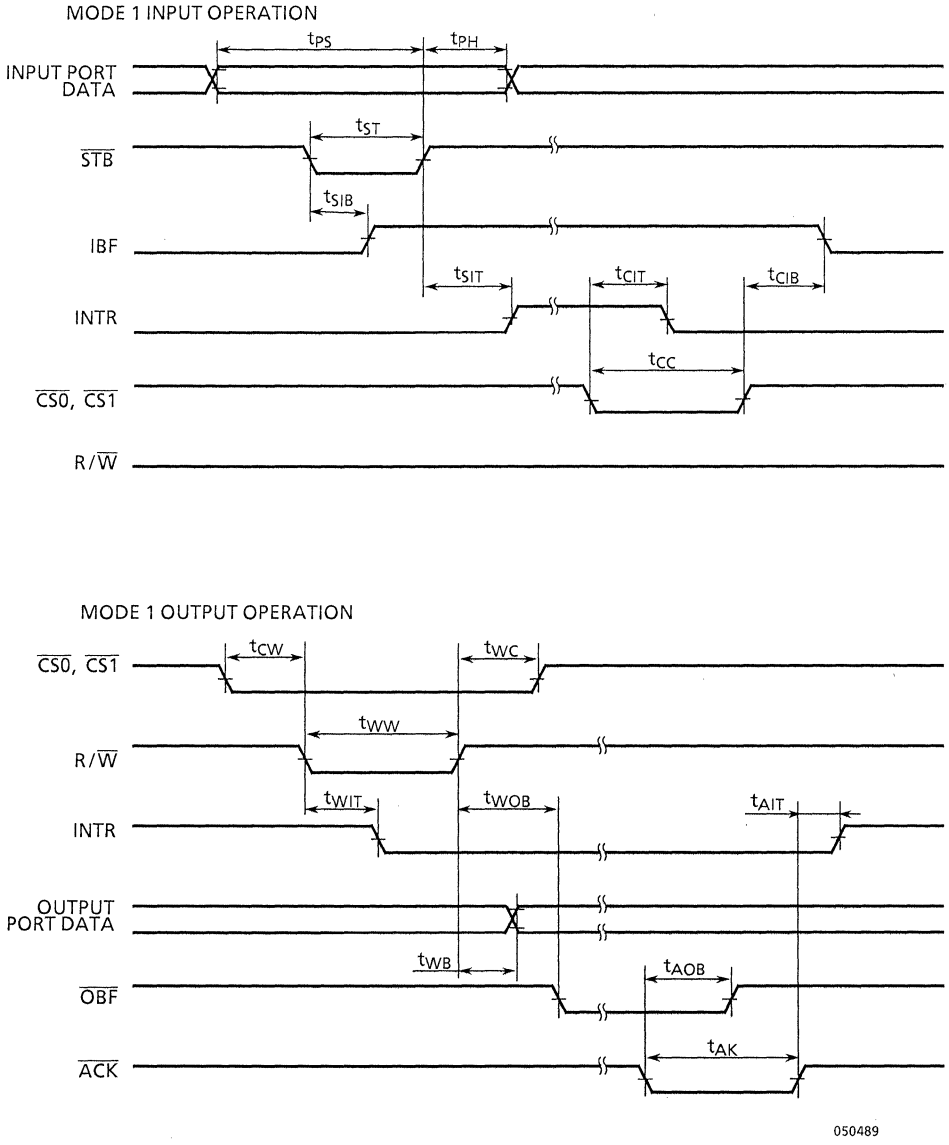


Figure 8.1 TMP82C255A Timing diagram (2)

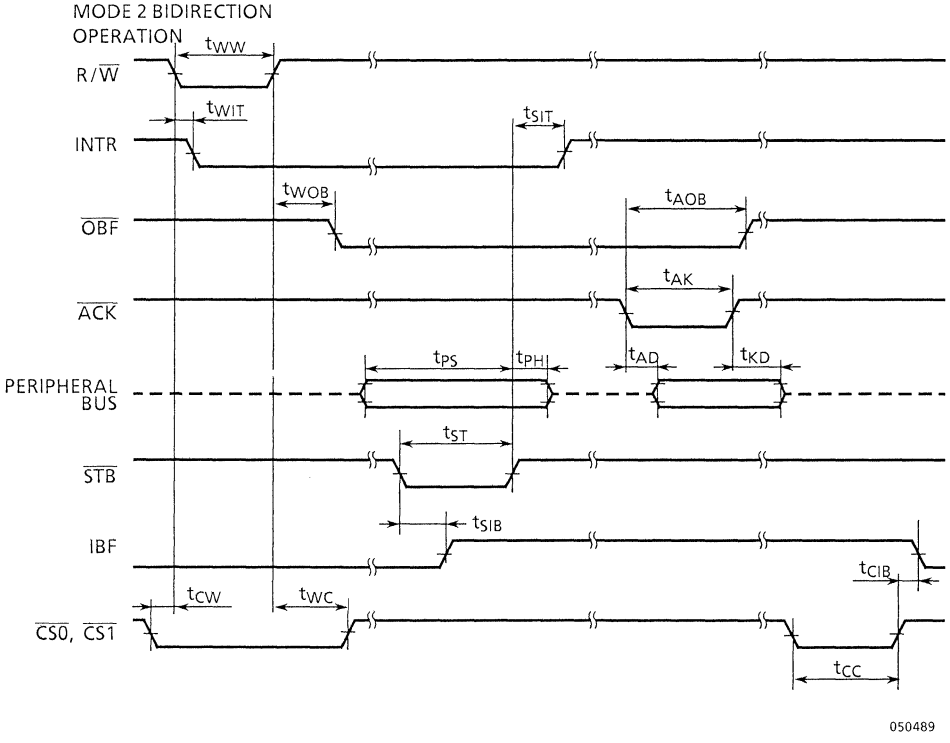


Figure 8.1 TMP82C255A Timing diagram (3)

8.2 TMP82C265A TIMING DIAGRAM (1)

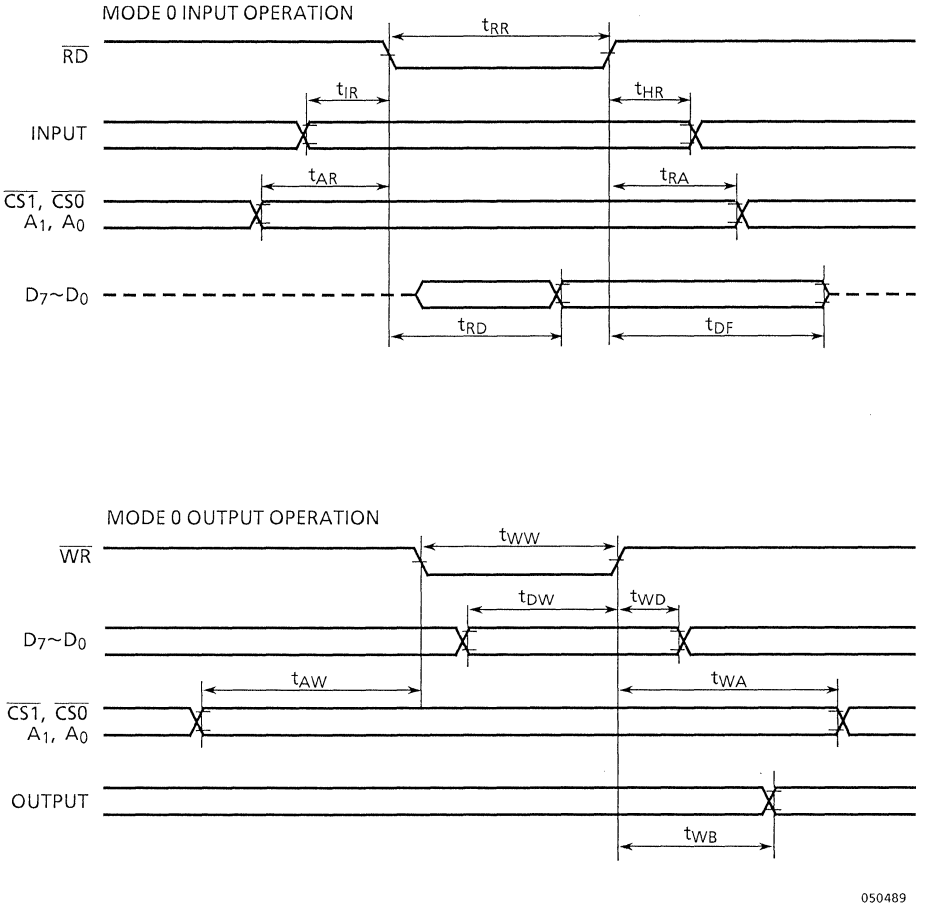


Figure 8.2 TMP82C265A Timing diagram (1)

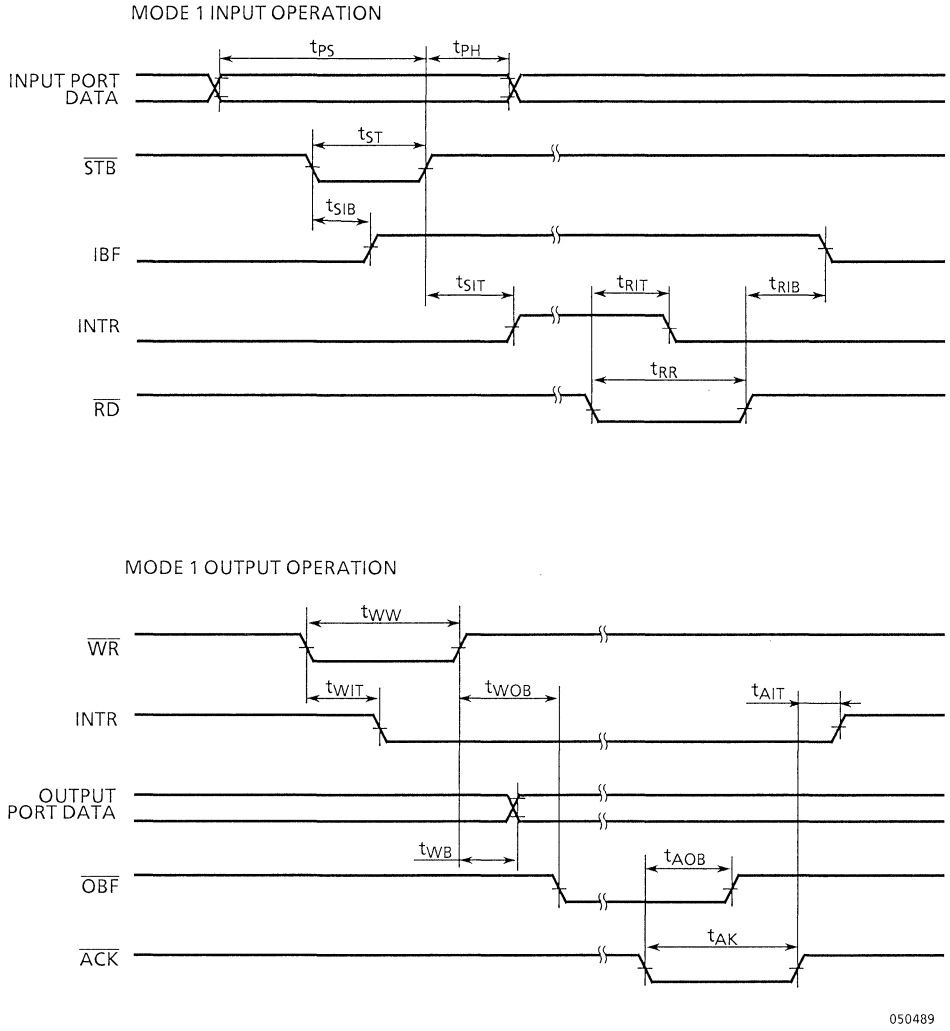


Figure 8.2 TMP82C265A Timing diagram (2)

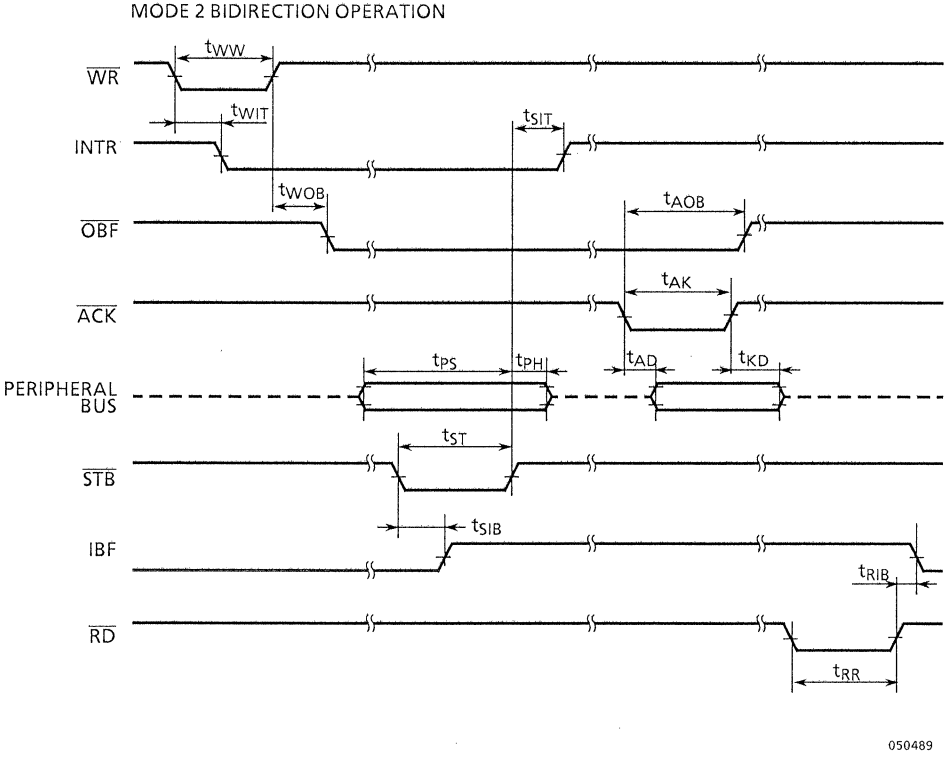
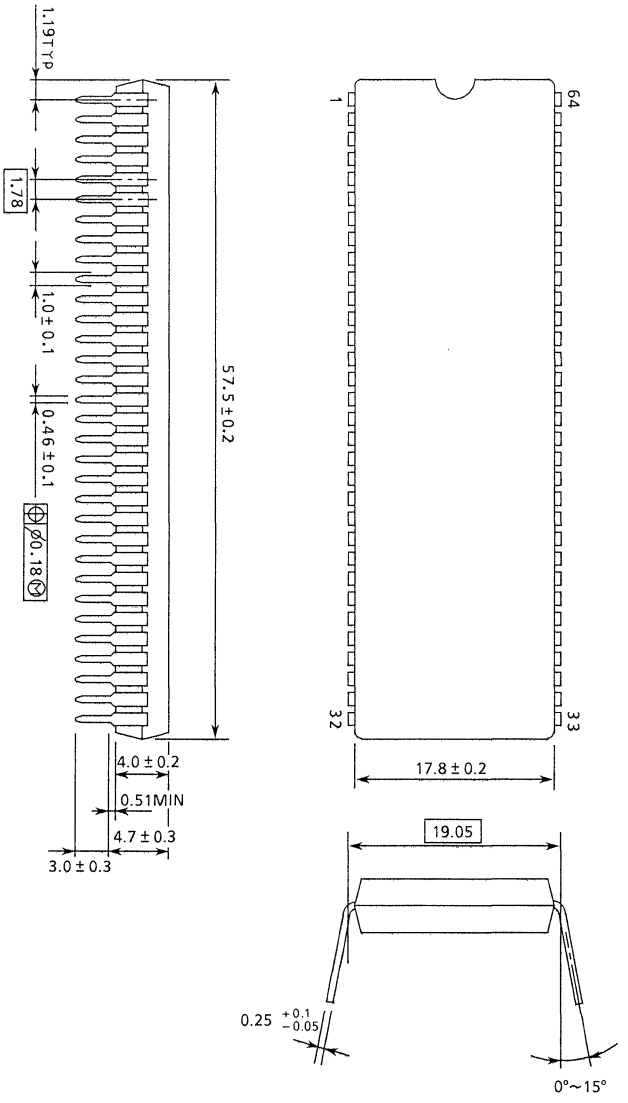


Figure 8.2 TMP82C265A Timing diagram (3)

9. OUTLINE DRAWINGS

SDIP64-P-750

Unit : mm

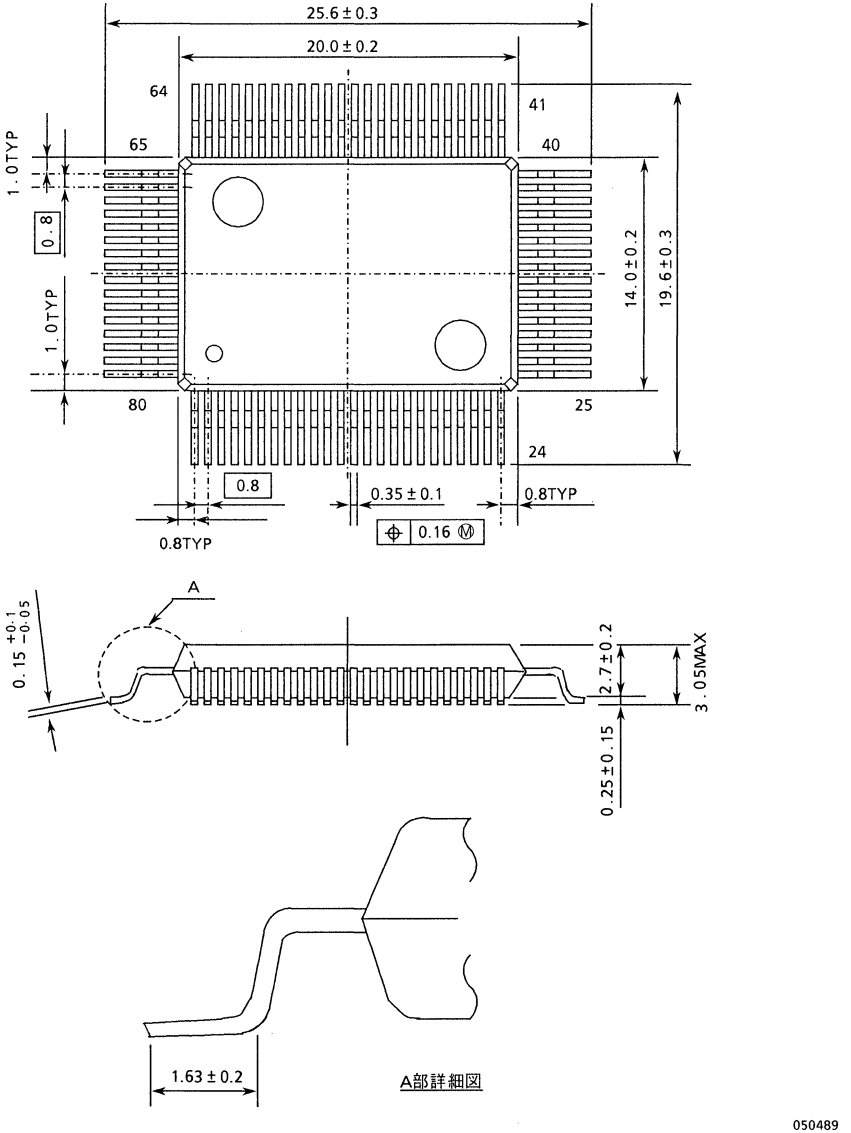


160289

Note : Each lead pitch is 1.78mm. All leads are located within 0.25mm of their trve longitudical position with respect to No.1 and No.64 leads.

QFP80-P-1420

Unit : mm



050489

SALES SUBSIDIARIES

Toshiba America Electronic Components, Inc.
Irvine Head Office
9775 Toledo Way, Irvine, CA 92718, U.S.A.
Tel.: (714) 455-2000 Fax: (714) 859-3963
Eastern Area Office
25 Mall Road, 5th Floor,
Burlington, MA 01803, U.S.A.
Tel.: (617) 272-4352, 5548 Fax: (617) 272-3089
South Eastern Regional Office
Waterford Center 5555, Triangle Parkway
Ste 300 Norcross, GA 30092, U.S.A.
Tel.: (404) 368-0203 Fax: (404) 368-0075
North Western Office
1220 Midas Way, Sunnyvale, CA 94086, U.S.A.
Tel.: (408) 737-9844 Fax: (408) 737-9905
ASIC Marketing Office
1220 Midas Way, Sunnyvale, CA 94086, U.S.A.
Tel.: (408) 733-3223 Fax: (408) 733-4539
South Western Regional Office
15621 Redhill Ave., Suite 205
Tustin, CA 92680, U.S.A.
Tel.: (714) 259-0368 Fax: (714) 259-9439
Central Area Office
One Parkway North, Suite 500, Deerfield,
IL 60015-2547, U.S.A.
Tel.: (708) 945-1500 Fax: (708) 945-1044
South Central Regional Office
777 East Campbell Rd., Suite 650,
Richardson, TX 75081, U.S.A.
Tel.: (214) 480-0470 Fax: (214) 235-4114
Fishkill Sales Office
RR 1 Box 6E, Windsor Park,
Fishkill, NY. 12524, U.S.A.
Tel.: (914) 896-6500 Fax: (914) 297-6851
Boca Raton Sales Office
1200 N. Federal Highway Suite 407,
Boca Raton, FL 33432, U.S.A.
Tel.: (305) 394-3004 Fax: (305) 394-3006
Detroit Office
1000 Prudential Town Center, Suite 2160,
Southfield, MI 48075, U.S.A.
Tel.: (313) 827-7700 Fax: (313) 827-4444
Portland Field Office
1700 NW 167 Place, Suite 240,
Beavertown, OR 97006, U.S.A.
Tel.: (503) 629-0818 Fax: (503) 629-0827
Marlton Field Office
One Greentree Center, Suite 201,
Marlton, NJ 08053, U.S.A.
Tel.: (609) 985-3737 Fax: (609) 596-8359

Toshiba Electronics Europe GmbH
Düsseldorf Head Office
Hansaallee 181, 4000 Düsseldorf 11,
F.R. Germany
Tel.: (0211) 5296-0 Fax: (0211) 5296-400
Telex: 8582685
Liaison Offices
Stuttgart: Vertriebsburo Baden-Wurttemberg
Ettinger Str. 61
D-7250 Leonberg, F.R. Germany
Tel.: (07152) 21061-66 Fax: (07152) 27658
Telex: 7245706
München: Buro München Arabellastr. 33/v.
8000 München 81, F.R. Germany
Tel.: (089) 928091-0 Fax: 089-9280942
Telex: 5-212363

Toshiba Electronics (UK) Limited
Riverside Way, Camberley,
Surrey GU 15 3YA, U.K.
Tel.: 0276-694600 Fax: 0276-691583

Toshiba Electronics Scandinavia AB
Gustavslundsvagen 141, 4th Floor
S-161 15 Bromma, Sweden
Tel.: 46-8-704 0900 Fax: 46-8-80 8459
Telex: 14169 TSBSTK S
(Mailing Address: P.O. Box 15031
S-161 15 Bromma Sweden)

Toshiba Electronics France SARL
Immeuble Robert Schumann, 3 Rue de Rome,
93561 Rosny-Sous-Bois Cedex
Tel.: (1) 48 94 20 20 Fax: (1) 48 94 51 15
Teles: 232 030

Toshiba Electronics Italiana S.R.L.
Centro Direzionale Colleoni Palazzo Orione-
Ingresso 3 (3° Piano) 20041 Agrate
Brianza (Milano), Italy
Tel.: 039-6057234 Fax: 039-6057252
Telex: 326423 SIAVBC

Toshiba Electronics España, S.A.
Torres Heron Plaza Colón No. 2. Torre II,
Planta 6-Pte-2,
28046 Madrid, Spain
Tel.: (1) 53-25-846 Fax: (1) 41-91-266
Telex: 44672 TOSHE (E)

Toshiba Electronics Taiwan Corp.
Taipei Head Office
8F, Min Sheng Chan-kwo Building, 348-350 Min Sheng
East Road Taipei, Taiwan
Tel.: 02-502-9641 Fax: 02-503-7964
Telex: 26874 TETTP
Kaohsiung Office
16F-A Chung-Cheng Building
No. 2 Chung-Cheng 3rd. 80027, Kaohsiung
Tel.: (07) 241-0826 Fax: (07) 282-7446

Toshiba Electronics Asia, Ltd.
Hong Kong Head Office
10th Floor, Sun Plaza, Canton Road, Tsimshatsui,
Kowloon, Hong Kong
Tel.: 3-721-6111 Fax: 3-739-8969
Telex: 38501 TSBEH HX
Seoul Branch Office
Room 1061, Chamber Building, 45, 4-KA
Namdaemun-Ro, Chung-Ku, Seoul, Korea
Tel.: (2) 757-2472 – 4 Fax: (2) 757-2475
Singapore Branch Office
460 Alexandra Road #21-00, PSA Building,
Singapore 0511
Tel.: 278 5252 Fax: 271 5155
Shanghai Branch Office
Room 2612 26F, Rui Jin Building,
205 Mao Ming South Road, Shanghai, China
Tel.: 86-21-4334077 86-21-4334131
Fax: 86-21-4333928
Bangkok Office
8/F, Thai Wah Tower, 21/20 South Sathorn Road,
Tungmahamek, Yannawa, Bangkok 10120, Thailand
Tel.: (2) 240-9447 Fax: (2) 240-9446

OVERSEAS OFFICES

São Paulo:
Toshiba Brasileira Representacoes Ltda.
Av., Paulista, 807, 21 Andar Cjto 2106,
Cerqueira Cesar,
Cep. 01311-São Paulo-S.P.-Brasil
Tel.: 283-4511, 4714, 4964 Fax: (11) 251-4104

Beijing:
Toshiba Corporation Beijing Office
Room 1622/1624 Beijing Hotel, Dong Chang An Jie,
Beijing, The People's Republic of China
Tel.: (01)55-4179, 4768 Fax: (01)513-7019
Telex: 22807 TOSPK CN Cable: TOSHIBA PEKING

Guangzhou:
Toshiba Corporation Guangzhou Office
Room 1161-1162 China Hotel, Office Tower, Liu Hua, Lu,
Guangzhou, The People's Republic of China
Tel.: (020)677420, 677800
Fax: (020)67-7427 Telex: 44585 TSBGZ CN
Cable: TOSHIBA GUANGZHOU

MANUFACTURING SUBSIDIARIES AND JOINT VENTURES

Toshiba America Electronic Components, Inc.
Microelectronics Center
1220, Midas Way, Sunnyvale, CA 94086, U.S.A.
Tel.: (408) 739-0560 Fax: (408) 746-0577
Telex: 346378

Toshiba Display Devices Inc.
Westinghouse Circle, Horseheads, New York
14845, U.S.A.
Tel.: (607) 796-3500 Fax: (607) 796-3564

Industria Mexicana Toshiba, S.A.
Calzada de Guadalupe, No. 303, Cuatitlan,
Edo de Mexico, Mexico
Tel.: 5-65-00-88 Telex: 017-72-560
Cable: Toshiba Mexico

Toshiba Semiconductor G.m.b.H.
Grotrian-Steinweg Str. 10, 3300
Braunschweig, F.R. Germany
Tel.: (0531) 31-0060 Fax: (0531) 31006139
Telex: 952368 TSCD

Toshiba Electronics Malaysia Sdn. Bhd.
42057 Telok Panglima, Grang, 15KM Klang-Banting
Road, Kuala Langat Selangor, Malaysia
Tel.: 03-352-6001-7 Fax: 03-352-6139
Telex: TOEL MA 39506
Penang Branch Office
Lot 2.08 2nd Floor, Wisma Chocolate Products,
41 Adoo Sittle Lane Panany 10400, Malaysia
Tel.: 04-368523, 04-368529 Fax: 04-368515
Kuala Lumpur Branch Office
6th Floor, Main Tower Block, Wisma Mcis, Jalan Barat
46200 Petaling Jaya, Malaysia
Tel.: 03-7565561 Fax: 03-7565409

Toshiba Display Devices (Thailand) Co., Ltd.
201 Vibhavadee Rangist Road, Ladyao Sub-district,
Bangkhen District, Bangkok
Tel.: (2)513-4260 Fax: (2)513-4260

9003

TOSHIBA

TOSHIBA CORPORATION
INTERNATIONAL OPERATIONS—ELECTRONIC COMPONENTS
1-1, SHIBAURA 1-CHOME, MINATO-KU, TOKYO, 105-01, JAPAN
Tel.: (03)457-3495 Fax: (03)451-0576 Telex: J22587

TOSHIBA AMERICA ELECTRONIC COMPONENTS, INC.

REGIONAL SALES OFFICES

NORTHWEST REGION
1220 Midas Way
Sunnyvale, CA 94086
TEL: (408) 737-9844
FAX: (408) 737-9905

SOUTHWEST REGION
15621 Redhill Ave.
Suite 205
Tustin, CA 92680
TEL: (714) 259-0368
FAX: (714) 259-9439

CENTRAL REGION
One Parkway North
Suite 500
Deerfield, IL 60015-2547
TEL: (708) 945-1500
FAX: (708) 945-1044
TWX: 29-7131

SOUTHCENTRAL REGION
777 E. Campbell Road
Suite 650
Richardson, TX 75081
TEL: (214) 480-0470
FAX: (214) 235-4114

EASTERN REGION
25 Mall Road
5th Floor
Burlington, MA 01803
TEL: (617) 272-4352
FAX: (617) 272-3089
TWX: 710-321-6730

SOUTHEAST REGION
Waterford Centre
5555 Triangle Parkway
Suite 300
Norcross, GA 30092
TEL: (404) 368-0203
FAX: (404) 368-0075

TOSHIBA

CSM-4402/6-90