Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Obsolete
Core Processor	Z80180
Number of Cores/Bus Width	1 Core, 8-Bit
Speed	6MHz
Co-Processors/DSP	-
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	-40°C ~ 100°C (TA)
Security Features	-
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8018006veg

Sections

Z8018X MPU Operation

Presents features, a general description, pins descriptions, block diagrams, registers, and details of operating modes for the Z8018x MPUs.

Software Architecture

Provides instruction sets and CPU registers for the Z8018x MPUs.

DC Characteristics

Presents the DC parameters and absolute maximum ratings for the Z8X180 MPUs.

AC Characteristics

Presents the AC parameters for the Z8018x MPUs.

Timing Diagrams

Contains timing diagrams and standard test conditions for the Z8018x MPUs.

Appendices

The appendixes in this manual provide additional information applicable to the Z8018x family of ZiLOG MPUs:

- Instruction set
- Instruction summary table
- Op Code map
- Bus Control signal conditions in each machine cycle and interrupt conditions
- Operating mode summary
- Status signals
- I/O registers and ordering information

address to 0. These instructions are IN0, OUT0, OTIM, OTIMR, OTDM, OTDMR and TSTIO (see Instruction Set).

When writing to an internal I/O register, the same I/O write occurs on the external bus. However, the duplicate external I/O write cycle exhibits internal I/O write cycle timing. For example, the $\overline{\text{WAIT}}$ input and programmable Wait State generator are ignored. Similarly, internal I/O read cycles also cause a duplicate external I/O read cycle. However, the external read data is ignored by the Z8X180.

Normally, external I/O addresses should be chosen to avoid overlap with internal I/O addresses and duplicate I/O accesses.

Table 6. I/O Address Map for Z80180-Class Processors Only

	Register	Mnemonic	Address		
			Binary	Hex	Page
ASCI	ASCI Control Register A Ch 0	CNTLA0	XX000000	00H	125
	ASCI Control Register A Ch 1	CNTLA1	XX000001	01H	128
	ASCI Control Register B Ch 0	CNTLB0	XX000010	02H	132
	ASCI Control Register B Ch 1	CNTLB1	XX000011	03H	132
	ASCI Status Register Ch 0	STAT0	XX000100	04H	120
	ASCI Status Register Ch 1	STAT1	XX000101	05H	123
	ASCI Transmit Data Register Ch 0	TDR0	XX000110	06H	118
	ASCI Transmit Data Register Ch 1	TDR1	XX000111	07H	118
	ASCI Receive Data Register Ch 0	RDR0	XX001000	08H	119
	ASCI Receive Data Register Ch 1	RDR1	XX001001	09H	119
CSI/O	CSI/O Control Register	CNTR	XX001010	0AH	147
	CSI/O Transmit/Receive Data Register	TRD	XX1011	0BH	149

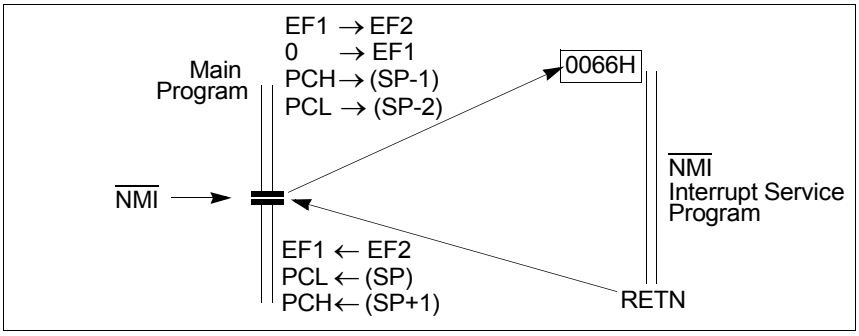
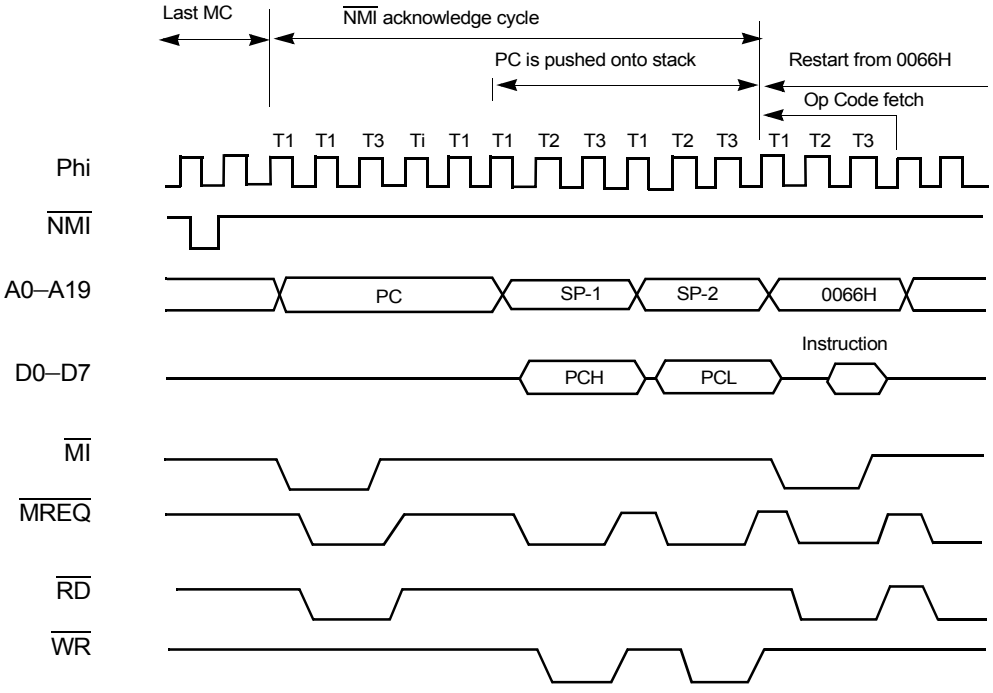


Figure 34. NMI Use



also the interrupt response sequence used for all internal interrupts (except TRAP).

As depicted in Figure 41, the low-order byte of the vector table address has the most significant three bits of the software programmable IL register while the least significant five bits are a unique fixed value for each interrupt ($\overline{INT1}$, $\overline{INT2}$ and internal) source:

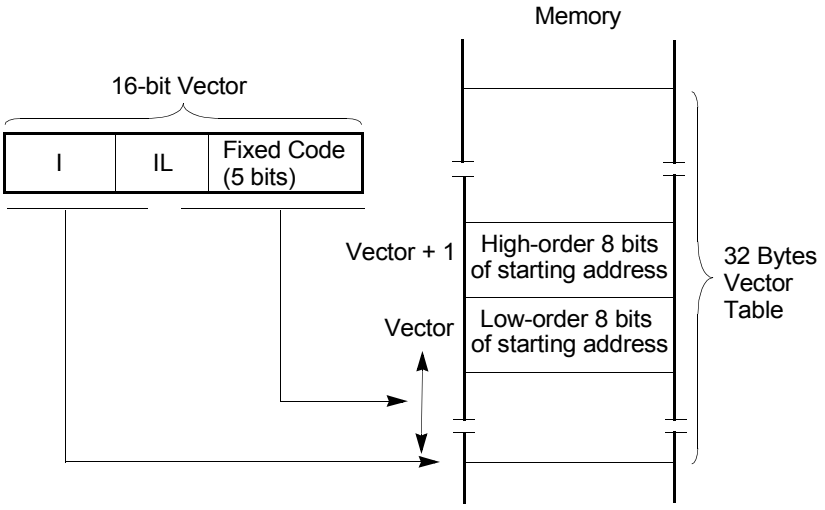


Figure 41. $\overline{INT1}$, $\overline{INT2}$ Vector Acquisition

$\overline{INT1}$ and $\overline{INT2}$ are globally masked by IEF1 is `0`. Each is also individually maskable by respectively clearing the ITE1 and ITE2 (bits 1,2) of the INT/TRAP control register to `0`.

During RESET, IEF1, ITE1 and ITE2 bits are reset to `0`.

Internal Interrupts

Internal interrupts (except TRAP) use the same vectored response mode as $\overline{INT1}$ and $\overline{INT2}$. Internal interrupts are globally masked by IEF1 is `0`. Individual internal interrupts are enabled/disabled by programming each

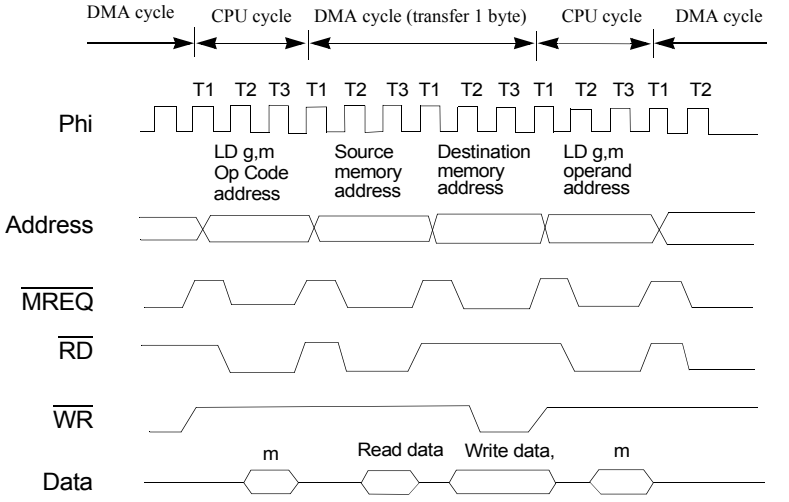


Figure 46. DMA Timing Diagram-CYCLE STEAL Mode

To initiate memory to/from memory DMA transfer for channel 0, perform the following operations.

1. Load the memory source and destination address into SAR0 and DAR0
2. Specify memory to/from memory mode and address increment/ decrement in the SM0 SM1, DM0 and DM1 bits of DMODE.
3. Load the number of bytes to transfer in BCR0.
4. Specify burst or cycle steal mode in the MMOD bit of DCNTL.
5. Program DE0 = `1` (with $\overline{\text{DWE0}}$ = `0` in the same access) in DSTAT and the DMA operation starts one machine cycle later. If interrupt occurs at the same time, the DIE0 bit must be set to `1`.

2. Specify memory ↔ I/O transfer mode and address increment/decrement in the SM0, SM1, DM0 and DM1 bits of DMODE.
3. Load the number of bytes to transfer in BCR0
4. The DMA request sense mode (DMS0 bit in DCNTL) must be specified as *edge sense*.
5. Enable or disable DMA termination interrupt with the DIE0 bit in DSTAT.
6. Program DE0 =1 (with $\overline{\text{DWE0}}$ = 0 in the same access) in DSTAT and the DMA operation with the ASCI begins under control of the ASCI generated internal DMA request.

The ASCI receiver or transmitter using DMA is initialized to allow the first DMA transfer to begin.

The ASCI receiver must be *empty* as shown by RDRF = 0.

The ASCI transmitter must be *full* as shown by TDRE = 0. Thus, the first byte is written to the ASCI Transmit Data Register under program control. The remaining bytes are transferred using DMA.

Channel 1 DMA

DMAC Channel 1 performs memory to/from I/O transfers. Except for different registers and status/control bits, operation is exactly the same as described for channel 0 memory to/from I/O DMA.

To initiate a DMA channel 1 memory to/from I/O transfer, perform the following operations:

1. Load the memory address (20 bits) into MAR1.
2. Load the I/O address (16 bits) into IAR1.
3. Program the source/destination and address increment/decrement mode using the DIM1 and DIM0 bits in DCNTL.

4. Specify whether $\overline{\text{DREQ1}}$ is level- or edge- sense in the DMS1 bit in DCNTL.
5. Enable or disable DMA termination interrupt with the DIE1 bit in DSTAT.
6. Program DE1 = 1 (with $\overline{\text{DWE1}}$ = 0 in the same access) in DSTAT and the DMA operation with the external I/O device begins using the external $\overline{\text{DREQ1}}$ input and $\overline{\text{TEND1}}$ output.

DMA Bus Timing

When memory (and memory mapped I/O) is specified as a source or destination, $\overline{\text{MREQ}}$ goes Low during the memory access. When I/O is specified as a source or destination, $\overline{\text{IORQ}}$ goes Low during the I/O access.

When I/O (and memory mapped I/O) is specified as a source or destination, the DMA timing is controlled by the external $\overline{\text{DREQ}}$ input and the $\overline{\text{TEND}}$ output indicates DMA termination

Note: External I/O devices may not overlap addresses with internal I/O and control registers, even using DMA.

For I/O accesses, one Wait State is automatically inserted. Additional Wait States can be inserted by programming the on-chip wait state generator or using the external $\overline{\text{WAIT}}$ input.

Note: For memory mapped I/O accesses, this automatic I/O Wait State is not inserted.

For memory to memory transfers (channel 0 only), the external $\overline{\text{DREQ0}}$ input is ignored. Automatic DMA timing is programmed as either BURST or CYCLE STEAL.

When a DMA memory address carry/borrow between bits A15 and A16 of the address bus occurs (crossing 64KB boundaries), the minimum bus

Bit Position	Bit/Field	R/W	Value	Description
0	TIE	R/W		**Transmit Interrupt Enable** — TIE must be set to 1 to enable ASCI transmit interrupt requests. If TIE is 1, an interrupt is requested when TDRE is 1. TIE is cleared to 0 during RESET.

Bit Position	Bit/Field	R/W	Value	Description
4	TE	R/W		**Transmit Enable** — A CSI/O transmit operation is started by setting TE to 1. When TE is set to 1, the data clock is enabled. When in internal clock mode, the data clock is output from the CKS pin. In external clock mode, the clock is input on the CKS pin. In either case, data is shifted out on the TXS pin synchronous with the (internal or external) data clock. After transmitting 8 bits of data, the CSI/O automatically clears TE to 0, EF is set to 1, and an interrupt (if enabled by EIE = 1) is generated. TE and RE are never both set to 1 at the same time. TE is cleared to 0 during RESET and IOSTOP mode.
2–0	SS2–0	R/W		**Speed Select** — Selects the CSI/O transmit/receive clock source and speed. SS2, SS I and SS0 are all set to 1 during RESET. Table 22 shows CSI/O Baud Rate Selection.

CSI/O Transmit/Receive Data Register (TRDR: I/O Address = 0BH).

TRDR is used for both CSI/O transmission and reception. Thus, the system design must insure that the constraints of half-duplex operation are met (Transmit and receive operation cannot occur simultaneously). For example, if a CSI/O transmission is attempted while the CSI/O is receiving data, the CSI/O does not work.

TRDR is not buffered. Attempting to perform a CSI/O transmit while the previous transmit data is still being shifted out causes the shift data to be immediately updated, thereby corrupting the transmit operation in progress. Similarly, reading TRDR during a transmit or receive must be avoided.

Flag Registers (F, F')

The flag registers store status bits (described in the next section) resulting from executed instructions.

General Purpose Registers (BC, BC', DE, DE', HL, HL')

The General Purpose Registers are used for both address and data operation. Depending on the instruction, each half (8 bits) of these registers (B, C, D, E, H, and I) may also be used.

Interrupt Vector Register (I)

For interrupts that require a vector table address to be calculated ($\overline{\text{INT0}}$ Mode 2, $\overline{\text{INT1}}$, $\overline{\text{INT2}}$, and internal interrupts), the Interrupt Vector Register (I) provides the most significant byte of the vector table address. I is cleared to 00H during reset.

R Counter (R)

The least significant seven bits of the R counter (R) count the number of instructions executed by the Z80180. R increments for each CPU Op Code fetch cycle (each $\overline{\text{M1}}$ cycle). R is cleared to 00H during reset.

Index Registers (IX, and IY)

The Index Registers are used for both address and data operations. For addressing, the contents of a displacement specified in the instruction are added to or subtracted from the Index Register to determine an effective operand address.

STANDARD TEST CONDITIONS

The previous DC Characteristics and Capacitance sections apply to the following standard test conditions, unless otherwise noted. All voltages are referenced to GND (0V). Positive current flows in to the referenced pin.

All AC parameters assume a load capacitance of 100 pF. Add 10 ns delay for each 50 pF increase in load up to a maximum of 200 pF for the data bus and 100 pF for the address and control lines. AC timing measurements are referenced to 1.5 volts (except for CLOCK, which is referenced to the 10% and 90% points).

The Ordering Information section lists temperature ranges and product numbers. Package drawings are in the Package Information section. Refer to the Literature List for additional documentation.

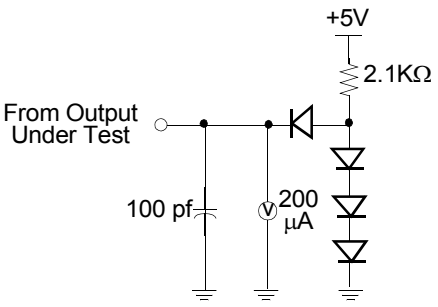


Figure 93. Test Setup

Table 39. Rotate and Shift Instructions (Continued)

Operation Name	Mnemonics	Op Code	Addressing							Bytes	States	Operation	Flags					
			Immed	Ext	Ind	Reg	Regi	Imp	Rel				7	6	4	2	1	0
													S	Z	H	P/V	N	C
	SRL (HL)	11 001 011				S/D				2	3		↑	↑	R	P	R	↑
		00 111 110																
	SRL (IX + d)	11 011 101			S/D					4	19		↑	↑	R	P	R	↑
		11 001 011																
		<d>																
		00 111 110																
	SRL (IY + d)	11 111 101			S/D					4	19		↑	↑	R	P	R	↑
		11 001 011																
		<d>																
		00 111 110																
Bit Set	SET b,g	11 001 011				S/D				2	7	1→b•gr	•	•	•	•	•	•
		11 b g																
	SET b,(HL)	11 001 011					S/D			2	13	$1 \rightarrow b \bullet (HL)_M$	•	•	•	•	•	•
		11 b 110																
	SET b,(IX + d)	11 011 101			S/D					4	19	$1 \rightarrow b \bullet (IX + d)_M$	•	•	•	•	•	•
		11 001 011																
		<d>																
		11 b 110																
	SET b,(IY + d)	11 111 101			S/D					4	19	$I \rightarrow b \bullet (IY + d)_M$	•	•	•	•	•	•
		11 001 011																
		<d>																
		11 b 110																
Bit Reset	RES b,g	11 001 011				S/D				2	7	0 →b•gr	•	•	•	•	•	•
		10 b g																
	RES b,(HL)	11 001 011					S/D			2	13	$0 \rightarrow 6 * b \bullet (HL)_M$	•	•	•	•	•	•
		10 b 110																
	RES b,(IX + d)	11 011 101			S/D					4	19	$0 \rightarrow \bullet b \bullet (IX + d)_M$	•	•	•	•	•	•
		11 001 011																
		<d>																
		10 b 110																

Table 42. 16-Bit Load (Continued)

Operation Name	Mnemonics	Op Code	Addressing							Bytes	States	Operation	Flags					
			Immed	Ext	Ind	Reg	Regl	Imp	Rel				7	6	4	2	1	0
													S	Z	H	P/V	N	C
Load 16-bit Data	LD (mn),HL	00 100 010 <n> <m>		D				S		3	16	Hr→(mn + 1)$_M$ Lr→(mn)$_M$	•	•	•	•	•	•
	LD (mn),IX	11 011 101 00 100 010 <n> <m>		D				S		4	19	IXHr-(mn + 1)$_M$ IXLr→(mn)$_M$	•	•	•	•	•	•
	LD (mn),IY	11 111 101 00 100 010 <n> <m>		D				S		4	19	IYHr→(mn + 1)$_M$ IYLr→(mn)$_M$	•	•	•	•	•	•

Table 43. Block Transfer

Operation Name	Mnemonics	Op Code	Addressing							Bytes	States	Operation	Flags					
			Immed	Ext	Ind	Reg	Regl	Imp	Rel				7	6	4	2	1	0
													S	Z	H	P/V	N	C
Block Transfer Search Data	CPD	11 101 101 10 101 001					S	S		2	12	Ar = (HL)$_M$ BC$_R$-1→BC$_R$ HL$_R$-1→HL$_R$	↑	(3) ↑	↑	(2) ↑ (3)	S	•
	CPDR	11 101 101 10 111 001					S	S		2	14 12	BC$_R$ ≠ 0 Ar ≠ (HL)$_M$ BC$_R$ = 0 or Ar = (HL)$_M$ Q Ar-(HL)$_R$ BC$_R$-1-BC$_R$ HL$_R$-1→HL$_R$ Repeat Q until Ar = (HL)$_M$ or BC$_R$ = 0	↑	↑ (3)	↑	(2) ↑ (2)	S	•

MNEMONICS	Bytes	Machine Cycles	States
	3	6	16 (If condition is true)
CALL mn	3	6	16
CCF	1	1	3
CPD	2	6	12
CPDR	2	8	14 (If $BC_R \neq 0$ and $Ar \neq (HL)_M$
	2	6	12 (If $BC_R = 0$ or $Ar = (HL)_M$
CP (HL)	1	2	6
CPI	2	6	12
CPIR	2	8	14 (If $BC_R \neq 0$ and $Ar \neq (HL)_M$
	2	6	12 (If BCR = 0 or $Ar = (HL)_M$
CP (IX+d)	3	6	14
CP (IY+d)	3	6	14
CPL	1	1	3
CP m	2	2	6
CP g	1	2	4
DAA	1	2	4
DEC (HL)	1	4	10
DEC IX	2	3	7
DEC IY	2	3	7
DEC (IX+d)	3	8	18
DEC (IY+d)	3	8	18
DEC g	1	2	4
DEC ww	1	2	4
DI	1	1	3
DJNZ j	2	5	9 (if Br ≠ 0)

MNEMONICS	Bytes	Machine Cycles	States
RRC (IY+d)	4	7	19
RRC g	2	3	7
RRD	2	8	16
RR (HL)	2	5	13
RR (IX+d)	4	7	19
RR (IY+d)	4	7	19
RR g	2	3	7
RST v	1	5	11
SBC A,(HL)	1	2	6
SBC A, (IX+d)	3	6	14
SBC A,(IY+d)	3	6	14
SBC A,m	2	2	6
SBC A,g	1	2	4
SBC HL,ww	2	6	10
SCF	1	1	3
SET b,(HL)	2	5	13
SET b,(IX+d)	4	7	19
SET b,(IY+d)	4	7	19
SET b,g	2	3	7
SLA (HL)	2	5	13
SLA (IX+d)	4	7	19
SLA (IY+d)	4	7	19
SLA g	2	3	7
SLP**	2	2	8
SRA (HL)	2	5	13
SRA (IX+d)	4	7	19
SRA (IY+d)	4	7	19
SRA g	2	3	7
SRL (HL)	2	5	13
SRL (IX+d)	4	7	19

Table 49. 2nd Op Code Map Instruction Format: CB XX

				b (L0 = 0~7)													
				0	2	4	6	0	2	4	6	0	2	4	6		
		HI		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
		LO		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
g (HI = ALL)	B	0000	0	RLC g	RL g	SLA g		BIT b,g				RES b,g				SET b,g				0
	C	0001	1																	1
	D	0010	2																	2
	E	0011	3																	3
	H	0100	4																	4
	L	0101	5																	5
	(HL)	0110	6	NOTE 1)	NOTE 1)	NOTE 1)		NOTE1)				NOTE1)				NOTE1)				6
	A	0111	7																	7
	B	1000	8	RRC g	RR g	SRA g	SRL g	BIT b,g				RES b,g				SET b,g				8
	C	1001	9																	9
	D	1010	A																	A
	E	1011	B																	B
	H	1100	C																	C
	L	1101	D																	D
	(HL)	1110	E	NOTE 1)	NOTE 1)	NOTE 1)	NOTE 1)	NOTE1)				NOTE1)				NOTE 1)				E
	A	1111	F																	F
				0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
								1	3	5	7	1	3	5	7	1	3	5	7	
				b (LO = 8 ~ F)																

Table 51. Bus and Control Signal Condition in Each Machine Cycle (Continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{MREQ}$	$\overline{IORQ}$	$\overline{M1}$	$\overline{HALT}$	ST
IN g,(C)	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2nd Op Code	0	1	0	1	0	1	1
	MC3	T1T2T3	BC	DATA	0	1	1	0	1	1	1
INO g,(m)**	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2nd Op Code	0	1	0	1	0	1	1
	MC3	T1T2T3	1st operand Address	m	0	1	0	1	1	1	1
	MC4	T1T2T3	m to A0~A7 00H to A8~A15	DATA	0	1	1	0	1	1	1
INI IND	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2nd Op Code	0	1	0	1	0	1	1
	MC3	T1T2T3	BC	DATA	0	1	1	0	1	1	1
	MC4	T1T2T3	HL	DATA	1	0	0	1	1	1	1
INIR INDR (If Br≠0)	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2nd Op Code	0	1	0	1	0	1	1
	MC3	T1T2T3	BC	DATA	0	1	1	0	1	1	1
	MC4	T1T2T3	HL	DATA	1	0	0	1	1	1	1
	MC5~MC6	TiTi	*	Z	1	1	1	1	1	1	1

Table 51. Bus and Control Signal Condition in Each Machine Cycle (Continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{MREQ}$	$\overline{IORQ}$	$\overline{M1}$	$\overline{HALT}$	ST
OTIM** OTDM**	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2nd Op Code	0	1	0	1	0	1	1
	MC3	Ti	*	Z	1	1	1	1	1	1	1
	MC4	T1T2T3	HL	DATA	0	1	0	1	1	1	1
	MC5	T1T2T3	C to A0~A7 00H to A8~A15	DATA	1	0	1	0	1	1	1
	MC6	Ti	*	Z	1	1	1	1	1	1	1
OTIMR** OTDMR** (If Br≠0)	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2nd Op Code	0	1	0	1	0	1	1
	MC3	Ti	*	Z	1	1	1	1	1	1	1
	MC4	T1T2T3	HL	DATA	0	1	0	1	1	1	1
	MC5	T1T2T3	C to A0~A7 00H to A8~A15	DATA	1	0	1	0	1	1	1
	MC6~MC8	TiTiTi	*	Z	1	1	1	1	1	1	1

Operating Modes Summary

REQUEST ACCEPTANCES IN EACH OPERATING MODE

Table 53. Request Acceptances in Each Operating Mode

Current Status / Request		Normal Operation (CPU mode and IOSTOP Mode)	WAIT State	Refresh Cycle	Interrupt Acknowledge Cycle	DMA Cycle	BUS RELEASE Mode	SLEEP Mode	SYSTEM STOP Mode
$\overline{WAIT}$		Acceptable	Acceptable	Not acceptable	Acceptable	Acceptable	Not acceptable	Not acceptable	Not acceptable
Refresh Request Request of Refresh by the on-chip Refresh Controller		Refresh cycle begins at the end of Machine Cycle (MC)	Not acceptable	Not acceptable	Refresh cycle begins at the end MC	Refresh cycle begins at the end of MC	Not acceptable	Not acceptable	Not acceptable
$\overline{DREQ0}$ $\overline{DREQ1}$		DMA cycle begins at the end of MC	DMA cycle begins at the end of MC	Acceptable Refresh cycle precedes. DMA cycle begins at the end of one MC	Acceptable DMA cycle begins at the end of MC.	Acceptable Refer to "DMA Controller" for details.	Acceptable *After BUS RELEASE cycle, DMA cycle begins at the end of one MC	Not acceptable	Not acceptable
$\overline{BUSREQ}$		Bus is released at the end of MC	Not acceptable	Not acceptable	Bus is released at the end of MC	Bus is released at the end of MC	Continue BUS RELEASE mode	Acceptable	Acceptable
Interrupt	$\overline{INT0}$, $\overline{INT1}$, $\overline{INT2}$	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not acceptable	Not acceptable	Not acceptable	Not acceptable	Acceptable Return from SLEEP mode to normal operation.	Acceptable Return from SYSTEM STOP mode to normal operation

Status Signals

PIN OUTPUTS IN EACH OPERATING MODE

Table 55 describes pin outputs in each operating mode.

Table 55. Pin Outputs in Each Operating Mode

Mode		$\overline{M1}$	$\overline{MREQ}$	$\overline{IORQ}$	$\overline{RD}$	$\overline{WR}$	$\overline{RFSH}$	$\overline{HALT}$	$\overline{BUSACK}$	ST	Address BUS	Data BUS
CPU Operation	Op Code Fetch (1st Op Code)	0	0	1	0	1	1	1	1	0	A	IN
	Op Code Fetch (except 1 st Op Code)	0	0	1	0	1	1	1	1	1	A	IN
	MemRead	1	0	1	0	1	1	1	1	1	A	IN
	Memory Write	1	0	1	1	0	1	1	1	1	A	OUT
	I/O Read	1	1	0	0	1	1	1	1	1	A	IN
	I/O Write	1	1	0	1	0	1	1	1	1	A	OUT
	Internal Operation	1	1	1	1	1	1	1	1	1	A	IN
Refresh		1	0	1	1	1	0	1	1	*	A	IN
Interrupt Acknowledge Cycle (1st Machine Cycle)	$\overline{NMI}$	0	0	1	0	1	1	1	1	0	A	IN
	$\overline{INT0}$	0	1	0	1	1	1	1	1	0	A	IN
	$\overline{INT1}$, $\overline{INT2}$ & Internal Interrupts	1	1	1	1	1	1	1	1	0	A	IN
BUS RELEASE		1	Z	Z	Z	Z	1	1	0	*	Z	IN
HALT		0	0	1	0	1	1	0	1	0	A	IN
SLEEP		1	1	1	1	1	1	0	1	1	1	IN