Zilog - Z8018010FSC00TR Datasheet

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	10MHz
Co-Processors/DSP	-
RAM Controllers	DRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	5.0V
Operating Temperature	0°C ~ 70°C (TA)
Security Features	-
Package / Case	80-BQFP
Supplier Device Package	80-QFP
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8018010fsc00tr

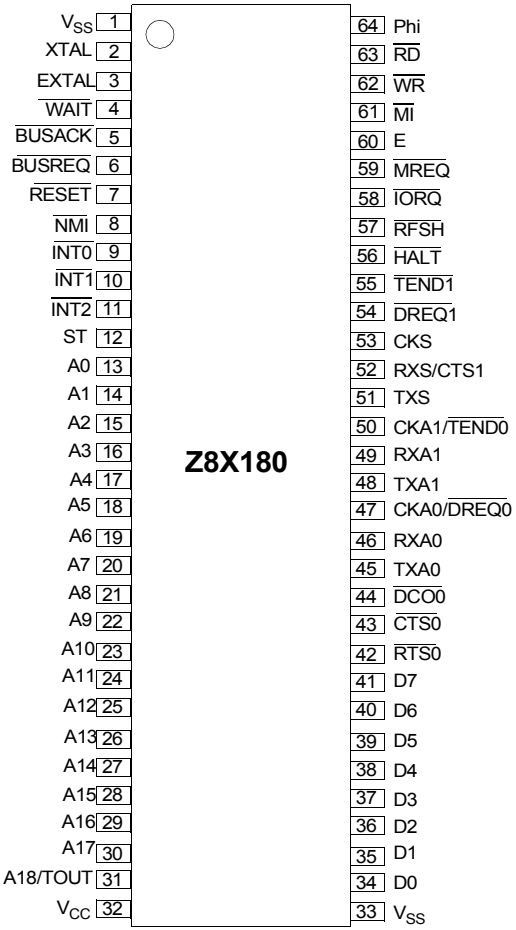

Figure 1. 64-Pin DIP

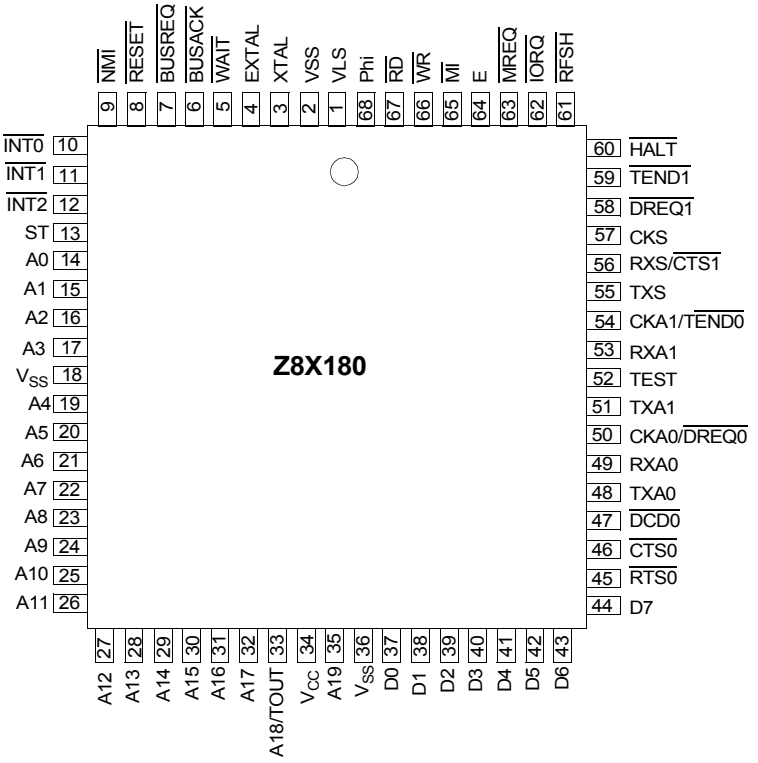

Figure 2. 68-Pin PLCC

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

MMU Register Description

MMU Common/Bank Area Register (CBAR)

CBAR specifies boundaries within the Z8X180 64KB logical address space for up to three areas; Common Area 0, Bank Area and Common Area 1.

MMU Common/Bank Area Register (CBAR: 3AH)

Bit	7	6	5	4	3	2	1	0
Bit/Field	CA3	CA2	CA1	CA0	BA3	BA2	BA1	BA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	0	0	0	0

Note: R = Read W = Write X = Indeterminate ? = Not Applicable

Bit Position	Bit/Field	R/W	Value	Description
7–4	CA7–4	R/W		CA specifies the start (low) address (on 4KB boundaries) for the Common Area 1. This also determines the last address of the Bank Area.
3–0	BA3–0	R/W		BA specifies the start (low) address (on 4KB boundaries) for the Bank Area. This also determines the last address of the Common Area 0.

disabling all maskable interrupts. The interrupt service routine normally terminates with the EI (Enable Interrupts) instruction followed by the RETI (Return from Interrupt) instruction, to reenable the interrupts. Figure 37 depicts the use of $\overline{\text{INT0}}$ (Mode 1) and RETI for the Mode 1 interrupt sequence.

Figure 37. INT0 Mode 1 Interrupt Sequence

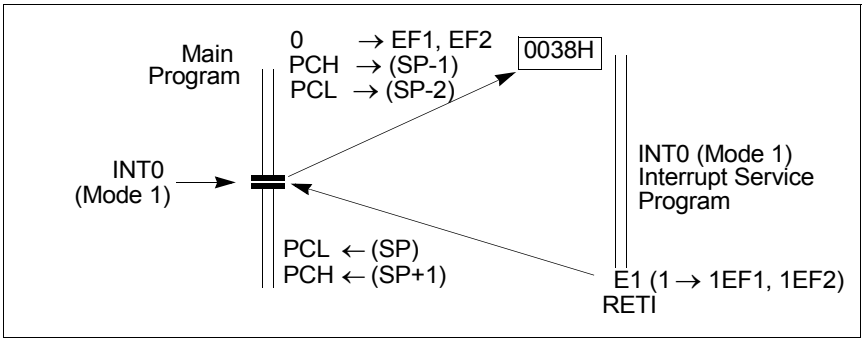

Figure 38 illustrates $\overline{\text{INT0}}$ Mode 1 Timing.

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:

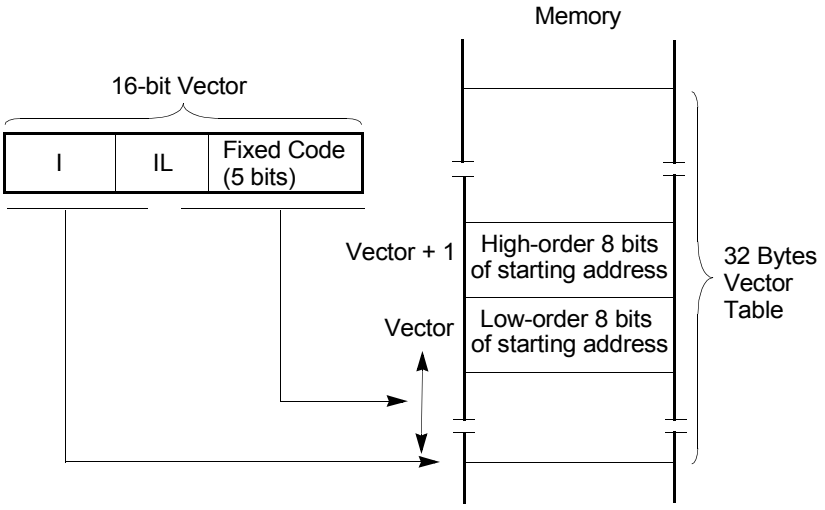

Figure 41. $\overline{INT1}$, $\overline{INT2}$ Vector Acquisition

$\overline{INT1}$ 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

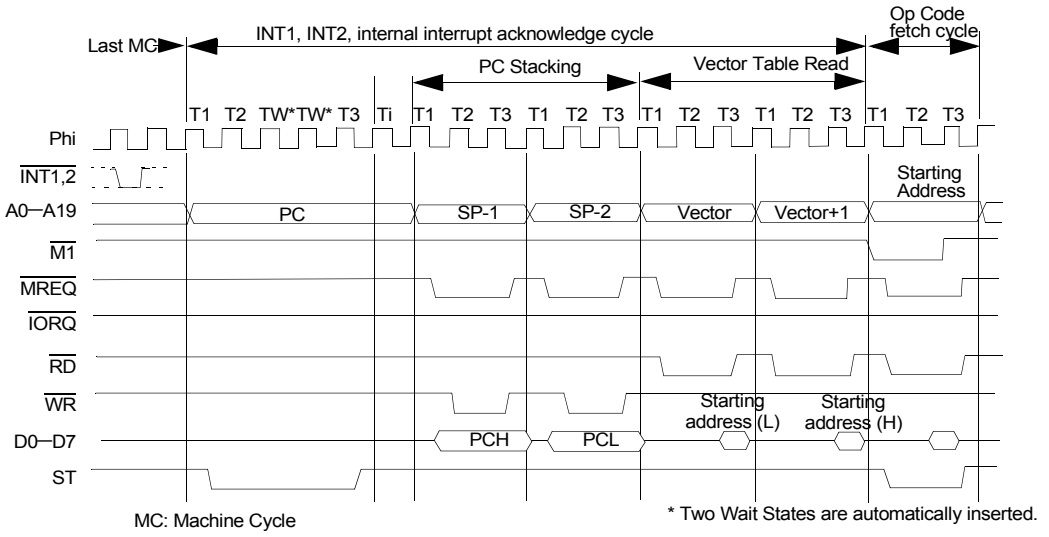

Figure 43. INT1, INT2 and Internal Interrupts Timing Diagram

Dynamic RAM Refresh Control

The Z8X180 incorporates a dynamic RAM refresh control circuit including 8-bit refresh address generation and programmable refresh timing. This circuit generates asynchronous refresh cycles inserted at the programmable interval independent of CPU program execution. For systems which do not use dynamic RAM, the refresh function can be disabled.

When the internal refresh controller determines that a refresh cycle should occur, the current instruction is interrupted at the first breakpoint between machine cycles. The refresh cycle is inserted by placing the refresh address on A0–A7 and the $\overline{\text{RFSH}}$ output is driven Low.

ASCI Receive Shift Register 0,1(RSR0, 1)

This register receives data shifted in on the RXA pin. When full, data is automatically transferred to the ASCI Receive Data Register (RDR) if it is empty. If RSR is not empty when the next incoming data byte is shifted in, an overrun error occurs. The RSR is not program-accessible.

ASCI Receive Data Register 0,1 (RDR0, 1: I/O Address = 08H, 09H)

When a complete incoming data byte is assembled in RSR, it is automatically transferred to the RDR if RDR is empty. The next incoming data byte can be shifted into RSR while RDR contains the previous received data byte. Thus, the ASCI receiver on Z80180 is double-buffered.

ASCI Receive Data Register Ch. 0 (RDR0: 08H)

Bit	7	6	5	4	3	2	1	0
Bit/Field	ASCI Receive Channel 0							
R/W	R/W							
Reset	0							
Note: R = Read W = Write X = Indeterminate ? = Not Applicable								

ASCI Receive Data Register Ch. 1 (RDR1: 09H)

Bit	7	6	5	4	3	2	1	0
Bit/Field	ASCI Receive Channel 1							
R/W	R/W							
Reset	0							
Note: R = Read W = Write X = Indeterminate ? = Not Applicable								

\

On the Z8S180 and Z8L180-class processors are quadruple buffered. The ASCI Receive Data Register is a read-only register. However, if RDRF =

ASCI to/from DMAC Operation

Operation of the ASCI with the on-chip DMAC channel 0 requires that the DMAC be correctly configured to use the ASCI flags as DMA request signals.

ASCI and RESET

During RESET, the ASCI status and control registers are initialized as defined in the individual register descriptions.

Receive and Transmit operations are stopped during RESET. However, the contents of the transmit and receive data registers (TDR and RDR) are not changed by RESET.

ASCI Clock

When in external clock input mode, the external clock is directly input to the sampling rate (÷16/÷64) as depicted in Figure 56.

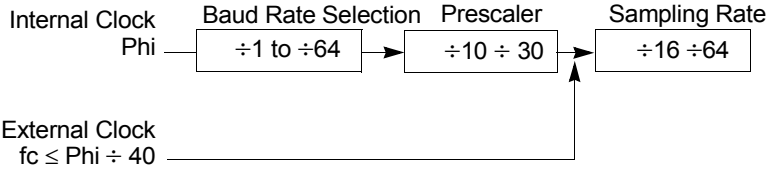

Figure 56. ASCI Clock

CSI/O and RESET

During RESET each bit in the CNTR is initialized as defined in the CNTR register description. CSI/O transmit and receive operations in progress are aborted during RESET. However, the contents of TRDR are not changed.

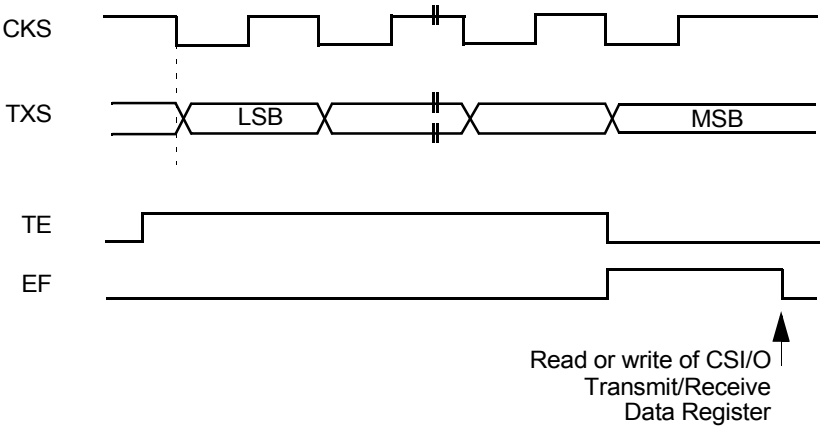

Figure 59. Transmit Timing Diagram–Internal Clock

return accurate data without requiring the timer to be stopped. The write procedure requires the PRT to be stopped.

For reading (without stopping the timer), TMDR is read in the order of lower byte - higher byte (TMDRnL, TMDRnH). The lower byte read (TMDRnL) stores the higher byte value in an internal register. The following higher byte read (TMDRnH) accesses this internal register. This procedure insures timer data validity by eliminating the problem of potential 16-bit timer updating between each 8-bit read. Specifically, reading TMDR in higher byte–lower byte order may result in invalid data. Note the implications of TMDR higher byte internal storage for applications which may read only the lower and/or higher bytes. In normal operation all TMDR read routines must access both the lower and higher bytes, in that order. For writing, the TMDR down counting must be inhibited using the TDE (Timer Down Count Enable) bits in the TCR (Timer Control Register). Then, any or both higher and lower bytes of TMDR can be freely written (and read) in any order.

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.

Timer Reload Register Channel 1L (RLDR1L: 16H)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Timer Reload Data							
R/W	R/W							
Reset	0							

Note: R = Read W = Write X = Indeterminate ? = Not Applicable

Timer Reload Register Channel 1H (RLDR1H: 17H)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Timer Reload Data							
R/W	R/W							
Reset	0							

Note: R = Read W = Write X = Indeterminate ? = Not Applicable

Timer Control Register (TCR)

TCR monitors both channels (PRT0, PRT1) TMDR status. It also controls enabling and disabling of down counting and interrupts along with controlling output pin A18/TOUT for PRT1.

Timer Control Register (TCR: 10H)

Bit	7	6	5	4	3	2	1	0
Bit/Field	TIF1	TIF0	TIE1	TIE0	TOC1	TOC0	TDE1	TDE0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Note: R = Read W = Write X = Indeterminate ? = Not Applicable

Addressing Modes

The Z80180 instruction set includes eight addressing modes.

- Implied Register
- Register Direct
- Register Indirect
- Indexed
- Extended
- Immediate
- Relative
- IO

Implied Register (IMP)

Certain Op Codes automatically imply register usage, such as the arithmetic operations that inherently reference the Accumulator, Index Registers, Stack Pointer, and General Purpose Registers.

Register Direct (REG)

Many Op Codes contain bit fields specifying registers used for operation. The exact bit field definitions vary depending on instruction depicted in Figure 75.

Z80180 DC CHARACTERISTICS

V_{CC} = 5V ± 10%, V_{SS} = OV, Ta = 0° to +70°C, unless otherwise noted.)

Table 28. Z80180 DC Characteristics

Symbol	Item	Condition	Minimum	Typical	Maximum	Unit
VIH1	Input High Voltage $\overline{RESET}$, EXTAL $\overline{NMI}$		V_{CC} –0.6	–	V_{CC} +0.3	V
VIH2	Input High Voltage except $\overline{RESET}$, EXTAL $\overline{NMI}$		2.0		V_{CC} +0.3	V
VIL1	Input Low Voltage $\overline{RESET}$, EXTAL $\overline{NMI}$		–0.3		0.6	V
VIL2	Input Low Voltage except $\overline{RESET}$, EXTAL $\overline{NMI}$		–0.3		0.8 Standard 7 TL_{VIL}	V
VOH	Output High Voltage all outputs	IOH = –200 µA IOH = –20 µA	2.4 V_{CC} –1.2	– –	– –	V V
VOL	Output Low Voltage all outputs	IOL = 2.2 mA	–	–	0.45	V
I_{IL}	Input Leakage Current all inputs except XTAL, EXTAL	V_{IN} = 0.5~ V_{CC} –0.5	–	–	1.0	µA
ITL	Three-State Leakage Current		–	–	1.0	µA
ICC	Power Dissipation* (Normal Operation)	f = 6 MHz f = 8 MHz f = 33 MHz	– – –	15 20 25	40 50 60	mA mA mA

Table 38. Arithmetic and Logical Instructions (8-bit) (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
	XOR (IY + d)	11 111 101			S			D		3	14	$Ar \oplus + (IY + d))_M \rightarrow Ar$	↑	↑	R	P	R	R
		10 101 110 <d>																

Table 39. Rotate and Shift Instructions

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
Rotate and Shift Data	RL A	00 010 1111						S/D		1	3		•	•	R	•	R	↑
	RL g	11 001 011 00 010 g				S/D				2	7		↑	↑	R	P	R	↑
	RL (HL)	11 001 011 00 010 110					S/D			2	13		↑	↑	R	P	R	↑
	RL (IX + d)	11 011 101 11 001 011 <d> 00 010 110			S/D					4	19		↑	↑	R	P	R	↑
	RL (IY + d)	11 111 101 11 001 011 <d> 00 010 110			S/D					4	19		↑	↑	R	P	R	↑
	RLC A	00 000 111						S/D		1	3		•	•	R	•	R	↑
	RLC g	11 001 011 00 000 g				S/D				2	7		↑	↑	R	P	R	↑
	RLC (HL)	11 001 011 00 000 110					S/D			2	13		↑	↑	R	P	R	↑

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
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	Ti	*	Z	1	1	1	1	1	1	1
OUTI OUTD	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	HL	DATA	0	1	0	1	1	1	1
	MC4	T1T2T3	BC	DATA	1	0	1	0	1	1	1
OTIR OTDR (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	HL	DATA	0	1	0	1	1	1	1
	MC4	T1T2T3	BC	DATA	1	0	1	0	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
RLC (HL) RL (HL) RRC (HL) RR (HL) SLA (HL) SRA (HL) SRL (HL)	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	HL	DATA	0	1	0	1	1	1	1
	MC4	Ti	*	Z	1	1	1	1	1	1	1
	MC5	T1T2T3	HL	DATA	1	0	0	1	1	1	1
RLC (IX + d) RLC (IY + d) RL (IX + d) RL (IY + d) RRC (IX + d) RRC (IY + d) RR (IX + d) RR (IY + d) SLA (IX + d) SLA (IY + d) SRA (IX + d) SRA (IY + d) SRL (IX + d) SRL (IY + d)	MC1	T1T2T3	1st Op Code Address	1st Op Code	0	1	0	1	0	1	0
	MC2	T1T2T3	2nd Op Code Address	2ndOp Code	0	1	0	1	0	1	1
	MC3	T1T2T3	1st operand Address	d	0	1	0	1	1	1	1
	MC4	T1T2T3	3rd Op Code Address	3rd Op Code	0	1	0	1	0	1	1
	MC5	T1T2T3	IX+d IY+d	DATA	0	1	0	1	1	1	1
	MC6	Ti	*	Z	1	1	1	1	1	1	1
	MC7	T1T2T3	IX+d IY+d	DATA	1	0	0	1	1	1	1
RLD RRD	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	HL	DATA	0	1	0	1	1	1	1
	MC4~MC7	TiTiTiTi *		Z	1	1	1	1	1	1	1
	MC8	T1T2T3	HL	DATA	1	0	0	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