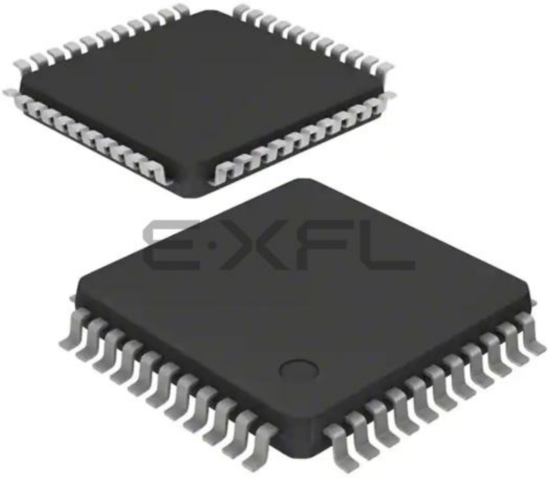
Zilog - Z84C0020FEG Datasheet



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	Z80
Number of Cores/Bus Width	1 Core, 8-Bit
Speed	20MHz
Co-Processors/DSP	-
RAM Controllers	-
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	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z84c0020feg

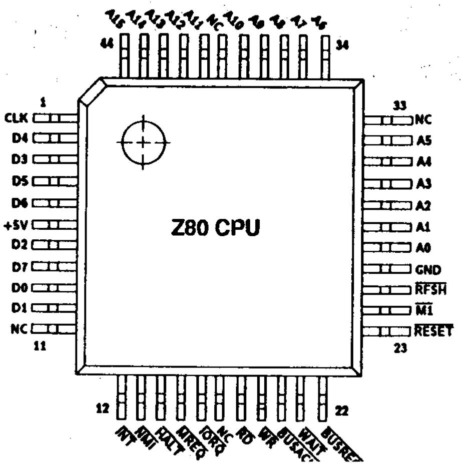


Figure 2a. 44-Pin LQFP, Pin Assignments (Only available for 84C00)

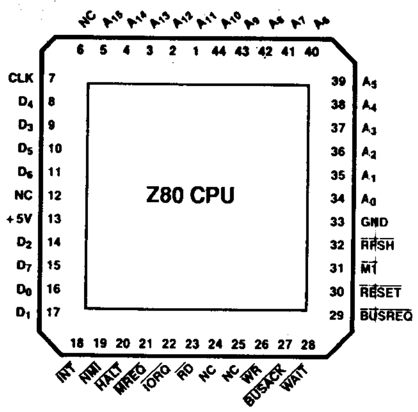


Figure 2b. 44-Pin Chip Carrier Pin Assignments

GENERAL DESCRIPTION

The CPUs are fourth-generation enhanced microprocessors with exceptional computational power. They offer higher system throughput and more efficient memory utilization than comparable second- and third-generation microprocessors. The internal registers contain 208 bits of read/write memory that are accessible to the programmer. These registers include two sets of six general-purpose registers which may be used individually as either 8-bit registers or as 16-bit register pairs. In addition, there are two sets of accumulator and flag registers. A group of "Exchange" instructions makes either set of main or alternate registers accessible to the programmer. The alternate set allows operation in foreground-background mode or it may be reserved for very fast interrupt response.

The CPU also contains a Stack Pointer, Program Counter, two index registers, a Refresh register (counter), and an Interrupt register. The CPU is easy to incorporate into a system since it requires only a single +5V power source. All output signals are fully decoded and timed to control standard memory or peripheral circuits; the CPU is supported by an extensive family of peripheral controllers. The internal block diagram (Figure 3) shows the primary functions of the processors. Subsequent text provides more detail on the I/O controller family, registers, instruction set, interrupts and daisy chaining, and CPU timing.

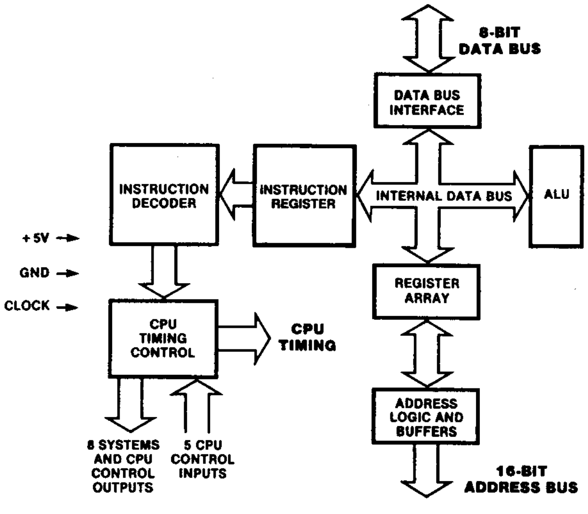


Figure 3. Z80C CPU Block Diagram

Table 1. Z80C CPU Registers

Register		Size (Bits)	Remarks
A, A′	Accumulator	8	Stores an operand or the results of an operation.
F, F′	Flags	8	See Instruction Set.
B, B′	General Purpose	8	Can be used separately or as a 16-bit register with C.
C, C′	General Purpose	8	Can be used separately or as a 16-bit register with C.
D, D′	General Purpose	8	Can be used separately or as a 16-bit register with E.
E, E′	General Purpose	8	Can be used separately or as a 16-bit register with E.
H, H′	General Purpose	8	Can be used separately or as a 16-bit register with L.
L, L′	General Purpose	8	Can be used separately or as a 16-bit register with L.
			Note: The (B,C), (D,E), and (H,L) sets are combined as follows: B — High byte C — Low byte D — High byte E — Low byte H — High byte L — Low byte
I	Interrupt Register	8	Stores upper eight bits of memory address for vectored interrupt processing.
R	Refresh Register	8	Provides user-transparent dynamic memory refresh. Automatically incremented and placed on the address bus during each instruction fetch cycle.
IX	Index Register	16	Used for indexed addressing.
IY	Index Register	16	Used for indexed addressing
SP	Stack Pointer	16	Holds address of the top of the stack. See Push or Pop in instruction set.
PC	Program Counter	16	Holds address of next instruction.
IFF_1-IFF_2	Interrupt Enable	Flip-Flops	Set or reset to indicate interrupt status (see Figure 4).
IMFa-IMFb	Interrupt Mode	Flip-Flops	Reflect Interrupt mode (see Figure 4).

failure has been detected. After recognition of the $\overline{\text{NMI}}$ signal (providing $\overline{\text{BUSREQ}}$ is not active), the CPU jumps to restart location 0066H. Normally, software starting at this address contains the interrupt service routine.

Maskable Interrupt ($\overline{\text{INT}}$). Regardless of the interrupt mode set by the user, the CPU response to a maskable interrupt input follows a common timing cycle. After the interrupt has been detected by the CPU (provided that interrupts are enabled and $\overline{\text{BUSREQ}}$ is not active) a special interrupt processing cycle begins. This is a special fetch ($\overline{\text{M1}}$) cycle in which $\overline{\text{IORQ}}$ becomes active rather than $\overline{\text{MREQ}}$, as in a normal $\overline{\text{M1}}$ cycle. In addition, this special $\overline{\text{M1}}$ cycle is automatically extended by two $\overline{\text{WAIT}}$ states, to allow for the time required to acknowledge the interrupt request.

Mode 0 Interrupt Operation. This mode is similar to the 8080 microprocessor interrupt service procedures. The interrupting device places an instruction on the data bus. This is normally a Restart instruction, which will initiate a call **to the selected one of eight restart locations in page zero of memory. Unlike the 8080, the Z80 CPU responds to the Call instruction with only one interrupt acknowledge cycle followed by two memory read cycles.**

Mode 1 Interrupt Operation. Mode 1 operation is very similar to that for the $\overline{\text{NMI}}$. The principal difference is that the Mode 1 interrupt has only one restart location, 0038H.

Mode 2 Interrupt Operation. This interrupt mode has been designed to most effectively utilize the capabilities of the Z80 microprocessor and its associated peripheral family.The interrupting peripheral device selects the starting address of the interrupt service routine. It does this by placing an 8-bit vector on the data bus during the interrupt acknowledge cycle. The CPU forms a pointer using this byte as the lower 8 bits and the contents of the I register as the upper 8 bits: This points to an entry in a table of addresses for interrupt service routines. The CPU then jumps to the routine at that

address. This flexibility in selecting the interrupt service routine address allows the peripheral device to use several different types of service routines. These routines may be located at any available location in memory. Since the interrupting device supplies the low-order byte of the 2-byte vector, bit 0 (A_0) must be a zero.

Interrupt Enable/Disable Operation. Two flip-flops, IFF_1 and IFF_2, referred to in the register description, are used to signal the CPU interrupt status. Operation of the two flip-flops is described in Table 2. For more details, refer to the *Z80 CPU Technical Manual* (03-0029-01) and *Z80 Assembly Language Programming Manual* (03-0002-01).

Table 2. State of Flip-Flops

Action	**IFF_1**	**IFF_2**	**Comments**
CPU Reset	0	0	Maskable interrupt $\overline{INT}$ disabled
DI instruction execution	0	0	Maskable interrupt $\overline{INT}$ disabled
EI instruction execution	1	1	Maskable interrupt $\overline{INT}$ enabled
LD A,I instruction execution	•	•	$IFF_2 \rightarrow$ Parity flag
LD A,R instruction execution	•	•	$IFF_2 \rightarrow$ Parity flag
Accept $\overline{NMI}$	0	•	Maskable interrupt $\overline{INT}$ disabled
RETN instruction execution	IFF_2	•	$IFF_2 \rightarrow IFF_1$ at completion of an $\overline{NMI}$ service routine.

INSTRUCTION SET

The microprocessor has one of the most powerful and versatile instruction sets available in any 8-bit microprocessor. It includes such unique operations as a block move for fast, efficient data transfers within memory, or between memory and I/O. It also allows operations on any bit in any location in memory.

The following is a summary of the instruction set which shows the assembly language mnemonic, the operation, the flag status, and gives comments on each instruction. For an explanation of flag notations and symbols for mnemonic tables, see the Symbolic Notations section which follows these tables. The *Z80 CPU Technical Manual* (03-0029-01), the *Programmer's Reference Guide* (03-0012-03), and *Assembly Language Programming Manual* (03-0002-01) contain significantly more details for programming use.

The instructions are divided into the following categories:

☐ 8-bit loads
☐ 16-bit loads
☐ Exchanges, block transfers, and searches
☐ 8-bit arithmetic and logic operations
☐ General-purpose arithmetic and CPU control
☐ 16-bit arithmetic operations
☐ Rotates and shifts
☐ Bit set, reset, and test operations
☐ Jumps
☐ Calls, returns, and restarts
☐ Input and output operations

A variety of addressing modes are implemented to permit efficient and fast data transfer between various registers, memory locations, and input/output devices. These addressing modes include:

☐ Immediate
☐ Immediate extended
☐ Modified page zero
☐ Relative
☐ Extended
☐ Indexed
☐ Register
☐ Register indirect
☐ Implied
☐ Bit

8-BIT LOAD GROUP

Mnemonic	Symbolic Operation	Flags								Opcode			Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments	
		S	Z		H		P/V	N	C	76	543	210						
LD r, r'	r ← r'	•	•	X	•	X	•	•	•	01	r	r'		1	1	4	r, r'	Reg.
LD r, n	r ← n	•	•	X	•	X	•	•	•	00	r	110		2	2	7	000	B
											← n →						001	C
LD r, (HL)	r ← (HL)	•	•	X	•	X	•	•	•	01	r	110		1	2	7	010	D
LD r, (IX+d)	r ← (IX+d)	•	•	X	•	X	•	•	•	11	011	101	DD	3	5	19	011	E
										01	r	110					100	H
											← d →						101	L
LD r, (IY+d)	r ← (IY+d)	•	•	X	•	X	•	•	•	11	111	101	FD	3	5	19	111	A
										01	r	110						
											← d →							
LD (HL), r	(HL) ← r	•	•	X	•	X	•	•	•	01	110	r		1	2	7		
LD (IX+d), r	(IX+d) ← r	•	•	X	•	X	•	•	•	11	011	101	DD	3	5	19		
										01	110	r						
											← d →							
LD (IY+d), r	(IY+d) ← r	•	•	X	•	X	•	•	•	11	111	101	FD	3	5	19		
										01	110	r						
											← d →							
LD (HL), n	(HL) ← n	•	•	X	•	X	•	•	•	00	110	110	36	2	3	10		
											← n →							
LD (IX+d), n	(IX+d) ← n	•	•	X	•	X	•	•	•	11	011	101	DD	4	5	19		
										00	110	110	36					
											← d →							
											← n →							

8-BIT LOAD GROUP (Continued)

Mnemonic	Symbolic Operation	Flags								Opcode			Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
		S	Z		H		P/V	N	C	76	543	210					
LD (IY+d), n	(IY+d) ← n	•	•	X	•	X	•	•	•	11	111	101	FD	4	5	19	
										00	110	110	36				
											← d →						
											← n →						
LD A, (BC)	A ← (BC)	•	•	X	•	X	•	•	•	00	001	010	0A	1	2	7	
LD A, (DE)	A ← (DE)	•	•	X	•	X	•	•	•	00	011	010	1A	1	2	7	
LD A, (nn)	A ← (nn)	•	•	X	•	X	•	•	•	00	111	010	3A	3	4	13	
											← n →						
											← n →						
LD (BC), A	(BC) ← A	•	•	X	•	X	•	•	•	00	000	010	02	1	2	7	
LD (DE), A	(DE) ← A	•	•	X	•	X	•	•	•	00	010	010	12	1	2	7	
LD (nn), A	(nn) ← A	•	•	X	•	X	•	•	•	00	110	010	32	3	4	13	
											← n →						
											← n →						
LD A, I	A ← I	↕	↕	X	0	X	IFF	0	•	11	101	101	ED	2	2	9	
										01	010	111	57				
LD A, R	A ← R	↕	↕	X	0	X	IFF	0	•	11	101	101	ED	2	2	9	
										01	011	111	5F				
LD I, A	I ← A	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	9	
										01	000	111	47				
LD R, A	R ← A	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	9	
										01	001	111	4F				

NOTE: IFF, the content of the interrupt enable flip-flop, (IFF$_2$), is copied into the P/V flag.

16-BIT LOAD GROUP

Mnemonic	Symbolic Operation	Flags								Opcode			Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
		S	Z		H		P/V	N	C	76	543	210					
LD dd, nn	dd ← nn	•	•	X	•	X	•	•	•	00	dd0	001		3	3	10	dd Pair
											← n →						00 BC
											← n →						01 DE
LD IX, nn	IX ← nn	•	•	X	•	X	•	•	•	11	011	101	DD	4	4	14	10 HL
										00	100	001	21				11 SP
											← n →						
											← n →						
LD IY, nn	IY ← nn	•	•	X	•	X	•	•	•	11	111	101	FD	4	4	14	
										00	100	001	21				
											← n →						
											← n →						
LD HL, (nn)	H ← (nn+1) L ← (nn)	•	•	X	•	X	•	•	•	00	101	010	2A	3	5	16	
											← n →						
											← n →						
LD dd, (nn)	dd_H ← (nn+1) dd_L ← (nn)	•	•	X	•	X	•	•	•	11	101	101	ED	4	6	20	
										01	dd1	011					
											← n →						
											← n →						

NOTE: (PAIR)$_H$, (PAIR)$_L$ refer to high order and low order eight bits of the register pair respectively. e.g., BC_L = C, AF_H = A.

16-BIT LOAD GROUP (Continued)

	Symbolic				Flags						Opcode			No. of	No. of M	No. of T	
Mnemonic	**Operation**	**S**	**Z**		**H**		**P/V**	**N**	**C**	**76**	**543**	**210**	**Hex**	**Bytes**	**Cycles**	**States**	**Comments**
LD IX, (nn)	$IX_H \leftarrow (nn+1)$	•	•	X	•	X	•	•	•	11	011	101	DD	4	6	20	
	$IX_L \leftarrow (nn)$									00	101	010	2A				
											← n →						
											← n →						
LD IY, (nn)	$IY_H \leftarrow (nn+1)$	•	•	X	•	X	•	•	•	11	111	101	FD	4	6	20	
	$IY_L \leftarrow (nn)$									00	101	010	2A				
											← n →						
											← n →						
LD (nn), HL	$(nn+1) \leftarrow H$	•	•	X	•	X	•	•	•	00	100	010	22	3	5	16	
	$(nn) \leftarrow L$										← n →						
											← n →						
LD (nn), dd	$(nn+1) \leftarrow dd_H$	•	•	X	•	X	•	•	•	11	101	101	ED	4	6	20	
	$(nn) \leftarrow dd_L$									01	dd0	011					
											← n →						
											← n →						
LD (nn), IX	$(nn+1) \leftarrow IX_H$	•	•	X	•	X	•	•	•	11	011	101	DD	4	6	20	
	$(nn) \leftarrow IX_L$									00	100	010	22				
											← n →						
											← n →						
LD (nn), IY	$(nn+1) \leftarrow IY_H$	•	•	X	•	X	•	•	•	11	111	101	FD	4	6	20	
	$(nn) \leftarrow IY_L$									00	100	010	22				
											← n →						
											← n →						
LD SP, HL	$SP \leftarrow HL$	•	•	X	•	X	•	•	•	11	111	001	F9	1	1	6	
LD SP, IX	$SP \leftarrow IX$	•	•	X	•	X	•	•	•	11	011	101	DD	2	2	10	
										11	111	001	F9				
LD SP, IY	$SP \leftarrow IY$	•	•	X	•	X	•	•	•	11	111	101	FD	2	2	10	
										11	111	001	F9				qq Pair
PUSH qq	$(SP-2) \leftarrow qq_L$	•	•	X	•	X	•	•	•	11	qq0	101		1	3	11	00 BC
	$(SP-1) \leftarrow qq_H$																01 DE
	$SP \rightarrow SP-2$																10 HL
PUSH IX	$(SP-2) \leftarrow IX_L$	•	•	X	•	X	•	•	•	11	011	101	DD	2	4	15	11 AF
	$(SP-1) \leftarrow IX_H$									11	100	101	E5				
	$SP \rightarrow SP-2$																
PUSH IY	$(SP-2) \leftarrow IY_L$	•	•	X	•	X	•	•	•	11	111	101	FD	2	4	15	
	$(SP-1) \leftarrow IY_H$									11	100	101	E5				
	$SP \rightarrow SP-2$																
POP qq	$qq_H \leftarrow (SP+1)$	•	•	X	•	X	•	•	•	11	qq0	001		1	3	10	
	qqL ← (SP)																
	$SP \rightarrow SP+2$																
POP IX	$IX_H \leftarrow (SP+1)$	•	•	X	•	X	•	•	•	11	011	101	DD	2	4	14	
	$IX_L \leftarrow (SP)$									11	100	001	E1				
	$SP \rightarrow SP+2$																
POP IY	$IY_H \leftarrow (SP+1)$	•	•	X	•	X	•	•	•	11	111	101	FD	2	4	14	
	$IY_L \leftarrow (SP)$									11	100	001	E1				
	$SP \rightarrow SP+2$																

NOTE: $(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively, e.g., $BC_L = C$, $AF_H = A$.

8-BIT ARITHMETIC AND LOGICAL GROUP (Continued)

	Symbolic				Flags						Opcode			No. of	No. of M	No. of T	
Mnemonic	**Operation**	**S**	**Z**		**H**		**P/V**	**N**	**C**	**76**	**543**	**210**	**Hex**	**Bytes**	**Cycles**	**States**	**Comments**
INC r	r←r+1	↕	↕	X	↕	X	V	0	•	00	r	$\boxed{100}$		1	1	4	
INC (HL)	(HL)←																
	(HL)+1	↕	↕	X	↕	X	V	0	•	00	110	$\boxed{100}$		1	3	11	
INC (IX+d)	(IX+d)←	↕	↕	X	↕	X	V	0	•	11	011	101	DD	3	6	23	
	(IX+d)+1									00	110	$\boxed{100}$					
											←d→						
INC (IY+d)	(IY+d)←	↕	↕	X	↕	X	V	0	•	11	111	101	FD	3	6	23	
	(IY+d)+1									00	110	$\boxed{100}$					
											←d→						
DEC m	m←m−1	↕	↕	X	↕	X	V	1	•			$\boxed{101}$					

NOTE: m is any of r, (HL), (IX+d), (IY+d) as shown for INC. DEC same format and states as INC. Replace $\boxed{100}$ with $\boxed{101}$ in opcode.

GENERAL-PURPOSE ARITHMETIC AND CPU CONTROL GROUPS

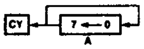
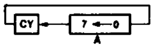
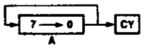
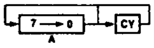
	Symbolic				Flags						Opcode			No. of	No. of M	No. of T	
Mnemonic	**Operation**	**S**	**Z**		**H**		**P/V**	**N**	**C**	**76**	**543**	**210**	**Hex**	**Bytes**	**Cycles**	**States**	**Comments**
DAA	@	↕	↕	X	↕	X	P	•	↕	00	100	111	27	1	1	4	Decimal adjust accumulator.
CPL	A←A	•	•	X	1	X	•	1	•	00	101	111	2F	1	1	4	Complement accumulator (one's complement).
NEG	A←0 − A	↕	↕	X	↕	X	V	1	↕	11	101	101	ED	2	2	8	Negate acc. (two's complement).
										01	000	100	44				
CCF	CY←CY	•	•	X	X	X	•	0	↕	00	111	111	3F	1	1	4	Complement carry flag.
SCF	CY←1	•	•	X	0	X	•	0	1	00	110	111	37	1	1	4	Set carry flag.
NOP	No operation	•	•	X	•	X	•	•	•	00	000	000	00	1	1	4	
HALT	CPU halted	•	•	X	•	X	•	•	•	01	110	110	76	1	1	4	
DI ★	IFF←0	•	•	X	•	X	•	•	•	11	110	011	F3	1	1	4	
EI ★	IFF←1	•	•	X	•	X	•	•	•	11	111	011	FB	1	1	4	
IM 0	Set interrupt mode 0	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	8	
										01	000	110	46				
IM 1	Set interrupt mode 1	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	8	
										01	010	110	56				
IM 2	Set interrupt mode 2	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	8	
										01	011	110	5E				

NOTES: @ converts accumulator content into packed BCD following add or subtract with packed BCD operands.
IFF indicates the interrupt enable flip-flop.
CY indicates the carry flip-flop.
★ indicates interrupts are not sampled at the end of EI or DI.

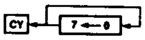
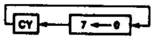
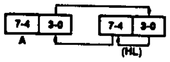
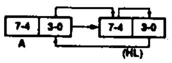
16-BIT ARITHMETIC GROUP

Mnemonic	Symbolic Operation	Flags S	Z		H		P/V	N	C	Opcode 76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
ADD HL, ss	HL←HL+ss	•	•	X	X	X	•	0	↕	00	ss1	001		1	3	11	ss Reg.
																	00 BC
ADC HL, ss	HL←																01 DE
	HL+ss+CY	↕	↕	X	X	X	V	0	↕	11	101	101	ED	2	4	15	10 HL
										01	ss1	010					11 SP
SBC HL, ss	HL←																
	HL−ss−CY	↕	↕	X	X	X	V	1	↕	11	101	101	ED	2	4	15	
										01	ss0	010					
ADD IX, pp	IX←IX+pp	•	•	X	X	X	•	0	↕	11	011	101	DD	2	4	15	pp Reg.
										01	pp1	001					00 BC
																	01 DE
																	10 IX
																	11 SP
ADD IY, rr	IY←IY+rr	•	•	X	X	X	•	0	↕	11	111	101	FD	2	4	15	rr Reg.
										00	rr1	001					00 BC
INC ss	ss←ss+1	•	•	X	•	X	•	•	•	00	ss0	011		1	1	6	01 DE
INC IX	IX←IX+1	•	•	X	•	X	•	•	•	11	011	101	DD	2	2	10	10 IY
										00	100	011	23				11 SP
INC IY	IY←IY+1	•	•	X	•	X	•	•	•	11	111	101	FD	2	2	10	
										00	100	011	23				
DEC ss	ss←ss−1	•	•	X	•	X	•	•	•	00	ss1	011		1	1	6	
DEC IX	IX←IX−1	•	•	X	•	X	•	•	•	11	011	101	DD	2	2	10	
										00	101	011	2B				
DEC IY	IY←IY−1	•	•	X	•	X	•	•	•	11	111	101	FD	2	2	10	
										00	101	011	2B				

ROTATE AND SHIFT GROUP

Mnemonic	Symbolic Operation	Flags S	Z		H		P/V	N	C	Opcode 76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
RLCA		•	•	X	0	X	•	0	↕	00	000	111	07	1	1	4	Rotate left circular accumulator.
RLA		•	•	X	0	X	•	0	↕	00	010	111	17	1	1	4	Rotate left accumulator.
RRCA		•	•	X	0	X	•	0	↕	00	001	111	0F	1	1	4	Rotate right circular accumulator.
RRA		•	•	X	0	X	•	0	↕	00	011	111	1F	1	1	4	Rotate right accumulator.

ROTATE AND SHIFT GROUP (Continued)

Mnemonic	Symbolic Operation	S	Z		Flags H		P/V	N	C	Opcode 76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
RLC r	 r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0 •	↕	11 00	001 $\boxed{000}$	011 r	CB	2	2	8	Rotate left circular register r. r Reg. 000 B 001 C 010 D 011 E 001 H 101 L 111 A
RLC (HL)		↕	↕	X	0	X	P	0	↕	11 00	001 000	011 110	CB	2	4	15	
RLC (IX+d)		↕	↕	X	0	X	P	0	↕	11 11 00	011 001 ←d→ $\boxed{000}$	101 011 110	DD CB	4	6	23	
RLC (IY+d)		↕	↕	X	0	X	P	0	↕	11 11 00	111 001 ←d→ $\boxed{000}$	101 011 110	FD CB	4	6	23	
RL m	 m = r,(HL,(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕		$\boxed{010}$						Instruction format and states are as shown for RLCs. To form new opcode replace $\boxed{000}$ or RLCs with shown code.
RRC m	 m = r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕		$\boxed{001}$						
RR m	 m = r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕		$\boxed{011}$						
SLA m	 m = r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕		$\boxed{100}$						
SRA m	 m = r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕		$\boxed{101}$						
SRL m	 m = r,(HL),(IX+d),(IY+d)	↕	↕	X	0	X	P	0	↕		$\boxed{111}$						
RLD		↕	↕	X	0	X	P	0	•	11 01	101 101	101 111	ED 6F	2	5	18	Rotate digit left and right between the accumulator and location (HL). The content of the upper half of the accumulator is unaffected.
RRD		↕	↕	X	0	X	P	0	•	11 01	101 100	101 111	ED 67	2	5	18	

JUMP GROUP

	Symbolic	Flags								Opcode				No. of	No. of M	No. of T	
Mnemonic	**Operation**	**S**	**Z**		**H**		**P/V**	**N**	**C**	**76**	**543**	**210**	**Hex**	**Bytes**	**Cycles**	**States**	**Comments**
JP nn	PC ← nn	•	•	X	•	X	•	•	•	11	000	011	C3	3	3	10	cc Condition
											← n →						000 NZ (non-zero)
											← n →						001 Z (zero)
JP cc, nn	If condition cc	•	•	X	•	X	•	•	•	11	cc	010		3	3	10	010 NC (non-carry)
	is true PC←nn,										← n →						011 C (carry)
	otherwise										← n →						100 PO (parity odd)
	continue																101 PE (parity even)
JR e	PC ← PC+e	•	•	X	•	X	•	•	•	00	011	000	18	2	3	12	110 P (sign positive)
											← e−2 →						111 M (sign negative)
JR C, e	If C=0,	•	•	X	•	X	•	•	•	00	111	000	38	2	2	7	If condition not met.
	continue										← e−2 →						
	If C=1,													2	3	12	If condition is met.
	PC ← PC+e																
JR NC, e	IF C=1,	•	•	X	•	X	•	•	•	00	110	000	30	2	2	7	If condition not met.
	continue										← e−2 →						
	If C=0,													2	3	12	If condition is met.
	PC ← PC+e																
JP Z, e	If Z=0	•	•	X	•	X	•	•	•	00	101	000	28	2	2	7	If condition not met.
	continue										← e−2 →						
	If Z=1,													2	3	12	If condition is met.
	PC ← PC+e																
JR NZ, e	If Z=1,	•	•	X	•	X	•	•	•	00	100	000	20	2	2	7	If condition not met.
	continue										← e−2 →						
	If Z=0,													2	3	12	If condition is met.
	PC ← PC+e																
JP (HL)	PC ← HL	•	•	X	•	X	•	•	•	11	101	001	E9	1	1	4	
JP (IX)	PC ← IX	•	•	X	•	X	•	•	•	11	011	101	DD	2	2	8	
										11	101	001	E9				
JP (IY)	PC ← IY	•	•	X	•	X	•	•	•	11	111	101	FD	2	2	8	
										11	101	001	E9				
DJNZ, e	B ← B−1	•	•	X	•	X	•	•	•	00	010	000	10	2	2	8	If B=0
	If B=0,										← e−2 →						
	continue																
	If B≠0,													2	3	13	If B≠0.
	PC ← PC+e																

NOTES: e represents the extension in the relative addressing mode.
e is a signal two's complement number in the range < −126, 129 >.
e−2 in the opcode provides an effective address of pc+e as PC is incremented by 2 prior to the addition of e.

PIN DESCRIPTIONS

A_0-A_{15}. *Address Bus* (output, active High, 3-state). A_0-A_{15} form a 16-bit address bus. The Address Bus provides the address for memory data bus exchanges (up to 64K bytes) and for I/O device exchanges.

$\overline{\mathrm{BUSACK}}$. *Bus Acknowledge* (output, active Low). Bus Acknowledge indicates to the requesting device that the CPU address bus, data bus, and control signals $\overline{\mathrm{MREQ}}$, $\overline{\mathrm{IORQ}}$, $\overline{\mathrm{RD}}$, and $\overline{\mathrm{WR}}$ have entered their high-impedance states. The external circuitry can now control these lines.

$\overline{\mathrm{BUSREQ}}$. *Bus Request* (input, active Low). Bus Request has a higher priority than $\overline{\mathrm{NMI}}$ and is always recognized at the end of the current machine cycle. $\overline{\mathrm{BUSREQ}}$ forces the CPU address bus, data bus, and control signals $\overline{\mathrm{MREQ}}$, $\overline{\mathrm{IORQ}}$, RD, and WR to go to a high-impedance state so that other devices can control these lines. $\overline{\mathrm{BUSREQ}}$ is normally wired-OR and requires an external pullup for these applications. Extended $\overline{\mathrm{BUSREQ}}$ periods due to extensive DMA operations can prevent the CPU from properly refreshing dynamic RAMs.

D_0-D_7. *Data Bus* (input/output, active High, 3-state). D_0-D_7 constitute an 8-bit bidirectional data bus, used for data exchanges with memory and I/O.

$\overline{\mathrm{HALT}}$. *Halt State* (output, active Low). $\overline{\mathrm{HALT}}$ indicates that the CPU has executed a Halt instruction and is awaiting either a nonmaskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOPs to maintain memory refresh.

$\overline{\mathrm{INT}}$. *Interrupt Request* (input, active Low). Interrupt Request is generated by I/O devices. The CPU honors a request at the end of the current instruction if the internal software-controlled interrupt enable flip-flop (IFF) is enabled. $\overline{\mathrm{INT}}$ is normally wired-OR and requires an external pullup for these applications.

$\overline{\mathrm{IORQ}}$. *Input/Output Request* (output, active Low, 3-state). $\overline{\mathrm{IORQ}}$ indicates that the lower half of the address bus holds a valid I/O address for an I/O read or write operation. $\overline{\mathrm{IORQ}}$ is also generated concurrently with $\overline{\mathrm{M1}}$ during an interrupt acknowledge cycle to indicate that an interrupt response vector can be placed on the data bus.

$\overline{\mathrm{M1}}$. *Machine Cycle One* (output, active Low). $\overline{\mathrm{M1}}$, together with $\overline{\mathrm{MREQ}}$, indicates that the current machine cycle is the opcode fetch cycle of an instruction execution. $\overline{\mathrm{M1}}$, together with $\overline{\mathrm{IORQ}}$, indicates an interrupt acknowledge cycle.

$\overline{\mathrm{MREQ}}$. *Memory Request* (output, active Low, 3-state). $\overline{\mathrm{MREQ}}$ indicates that the address bus holds a valid address for a memory read or memory write operation.

$\overline{\mathrm{NMI}}$. *Non-Maskable Interrupt* (input, negative edge-triggered). $\overline{\mathrm{NMI}}$ has a higher priority than $\overline{\mathrm{INT}}$. $\overline{\mathrm{NMI}}$ is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop, and automatically forces the CPU to restart at location 0066H.

$\overline{\mathrm{RD}}$. *Read* (output, active Low, 3-state). $\overline{\mathrm{RD}}$ indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.

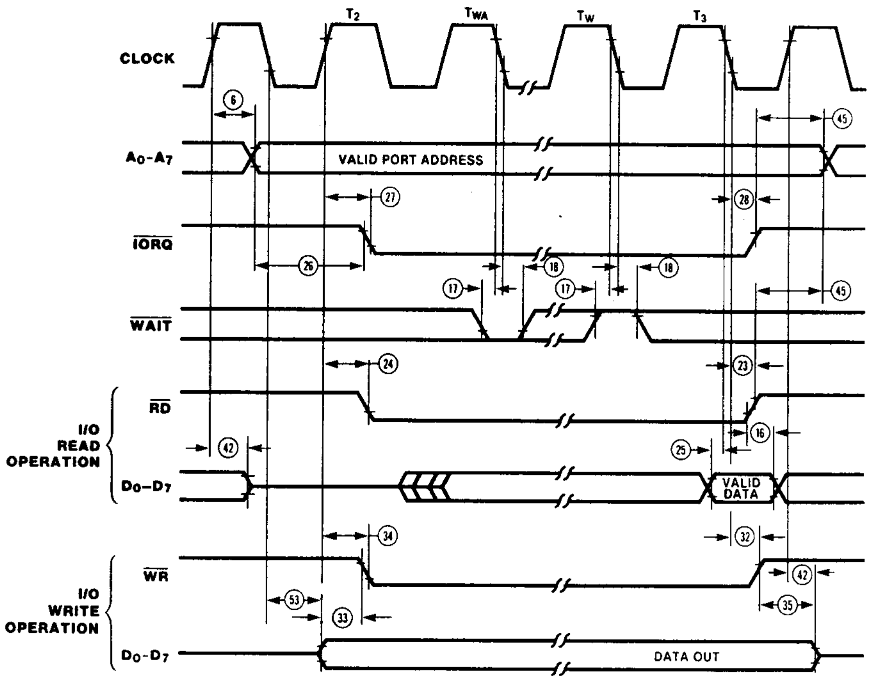
$\overline{\mathrm{RESET}}$. *Reset* (input, active Low). $\overline{\mathrm{RESET}}$ initializes the CPU as follows: it resets the interrupt enable flip-flop, clears the PC and Registers I and R, and sets the interrupt status to Mode 0. During reset time, the address and data bus go to a high-impedance state, and all control output signals go to the inactive state. Note that $\overline{\mathrm{RESET}}$ must be active for a minimum of three full clock cycles before the reset operation is complete.

$\overline{\mathrm{RFSH}}$. *Refresh* (output, active Low). $\overline{\mathrm{RFSH}}$, together with $\overline{\mathrm{MREQ}}$, indicates that the lower seven bits of the system's address bus can be used as a refresh address to the system's dynamic memories.

$\overline{\mathrm{WAIT}}$. *Wait* (input, active Low). $\overline{\mathrm{WAIT}}$ indicates to the CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter a Wait state as long as this signal is active. Extended $\overline{\mathrm{WAIT}}$ periods can prevent the CPU from properly refreshing dynamic memory.

$\overline{\mathrm{WR}}$. *Write* (output, active Low, 3-state). $\overline{\mathrm{WR}}$ indicates that the CPU data bus holds valid data to be stored at the addressed memory or I/O location.

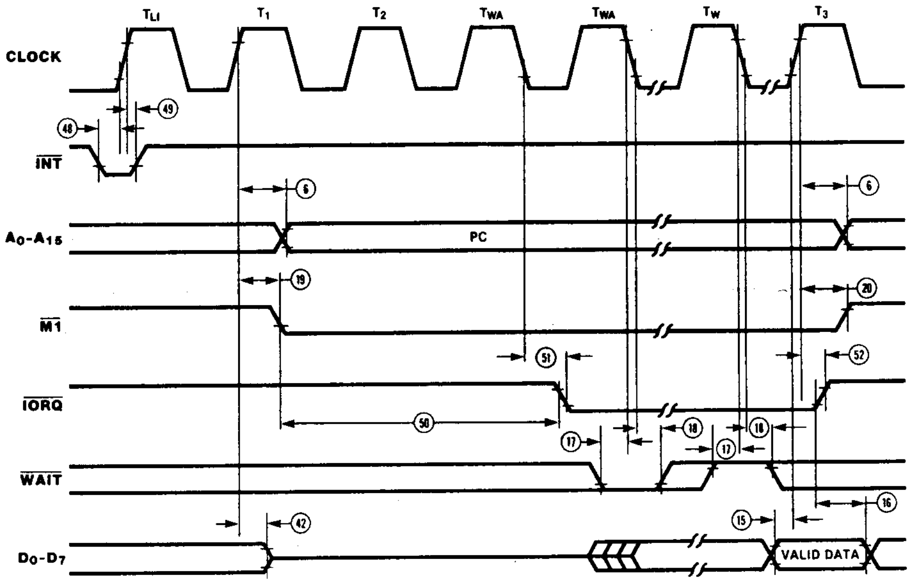
Input or Output Cycles. Figure 7 shows the timing for an I/O read or I/O write operation. During I/O operations, the CPU automatically inserts a single Wait state (T_{WA}). This extra Wait state allows sufficient time for an I/O port to decode the address from the port address lines.



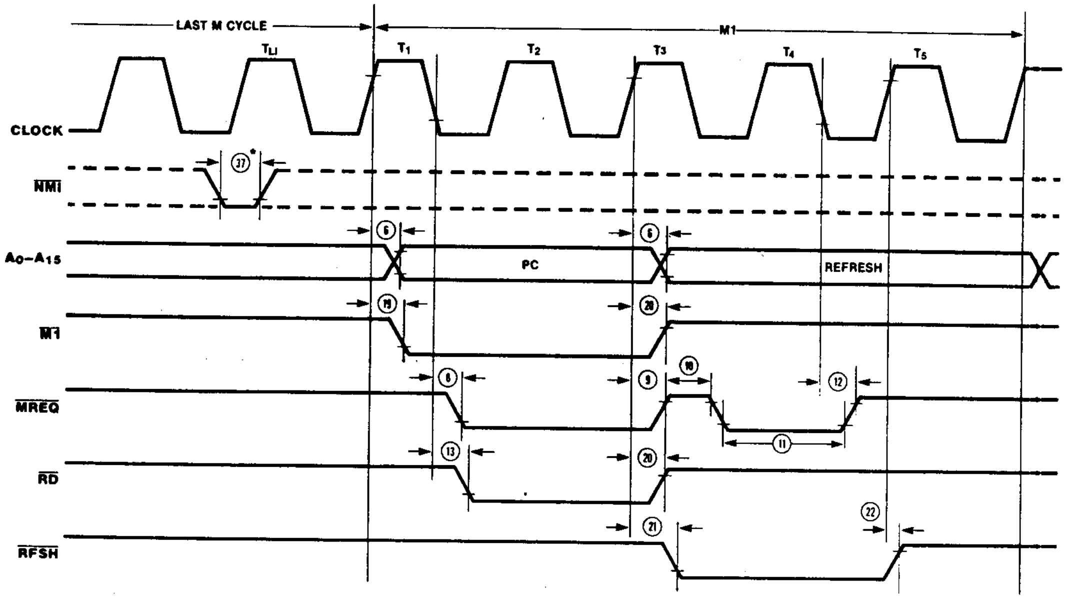
T_{WA} = One wait cycle automatically inserted by CPU.

Figure 7. Input or Output Cycles

Interrupt Request/Acknowledge Cycle. The CPU samples the interrupt signal with the rising edge of the last clock cycle at the end of any instruction (Figure 8). When an interrupt is accepted, a special $\overline{\text{M1}}$ cycle is generated. During this $\overline{\text{M1}}$ cycle, $\overline{\text{IORQ}}$ becomes active (instead of $\overline{\text{MREQ}}$) to indicate that the interrupting device can place an 8-bit vector on the data bus. The CPU automatically adds two Wait states to this cycle.



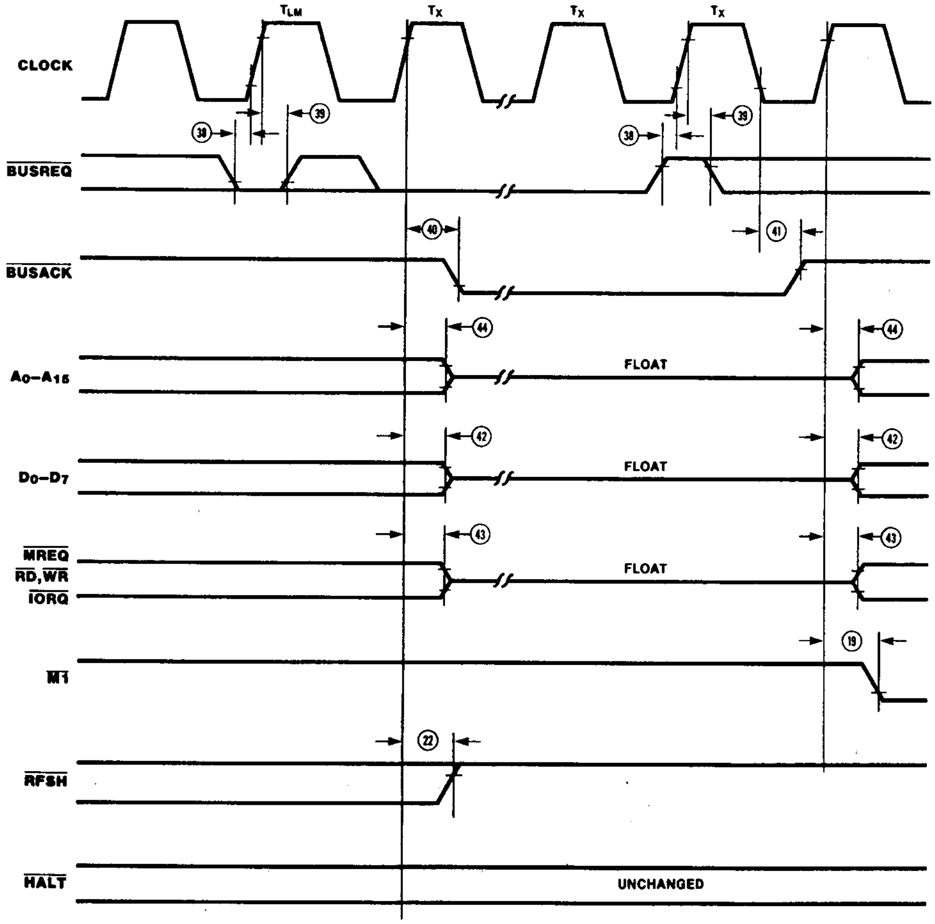
Non-Maskable Interrupt Request Cycle. $\overline{\text{NMI}}$ is sampled at the same time as the maskable interrupt input $\overline{\text{INT}}$ but has higher priority and cannot be disabled under software control. The subsequent timing is similar to that of a normal memory read operation except that data put on the bus by the memory is ignored. The CPU instead executes a restart (RST) operation and jumps to the $\overline{\text{NMI}}$ service routine located at address 0066H (Figure 9).



*Although $\overline{\text{NMI}}$ is an asynchronous input, to guarantee its being recognized on the following machine cycle, $\overline{\text{NMI}}$'s falling edge must occur no later than the rising edge of the clock cycle preceding the last state of any instruction cycle (T_{LI}).

Figure 9. Non-Maskable Interrupt Request Operation

Bus Request/Acknowledge Cycle. The CPU samples BUSREQ with the rising edge of the last clock period of any machine cycle (Figure 10). If BUSREQ is active, the CPU sets its address, data, and MREQ, IORQ, RD, and WR lines to a high-impedance state with the rising edge of the next clock pulse. At that time, any external device can take control of these lines, usually to transfer data between memory and I/O devices.



NOTES: 1) T_{LM} = Last state of any M cycle.
2) T_X = An arbitrary clock cycle used by requesting device.

Figure 10. BUS Request/Acknowledge Cycle

Power-Down Release Cycle. The system clock must be supplied to the CPU to release the power-down state. When the system clock is supplied to the CLK input, the CPU restarts operations from the point at which the power-down state was implemented.

The timing diagrams for the release from power-down mode **are shown in Figure 14.**

NOTES:

1) When the external oscillator has been stopped to enter the power-down state, some warm-up time may be required to obtain a stable clock for the release.
2) When the HALT instruction is executed to enter the power-down state, the CPU will also enter the Halt state. An interrupt signal (either $\overline{\text{NMI}}$ or $\overline{\text{INT}}$) or a $\overline{\text{RESET}}$ signal must be applied to the CPU after the system clock is supplied in order to release the power-down state.

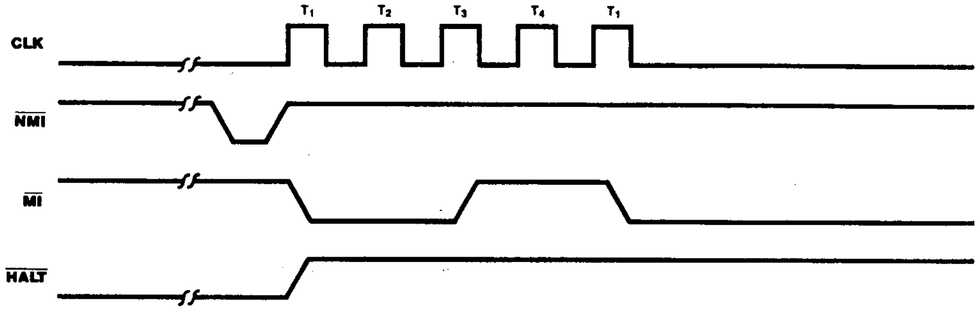


Figure 14a.

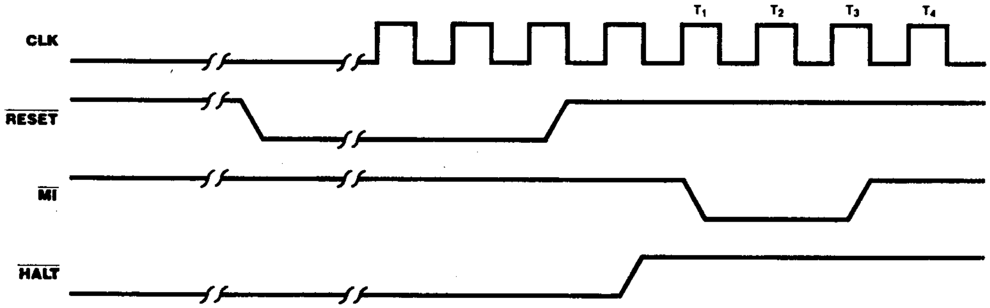


Figure 14b.

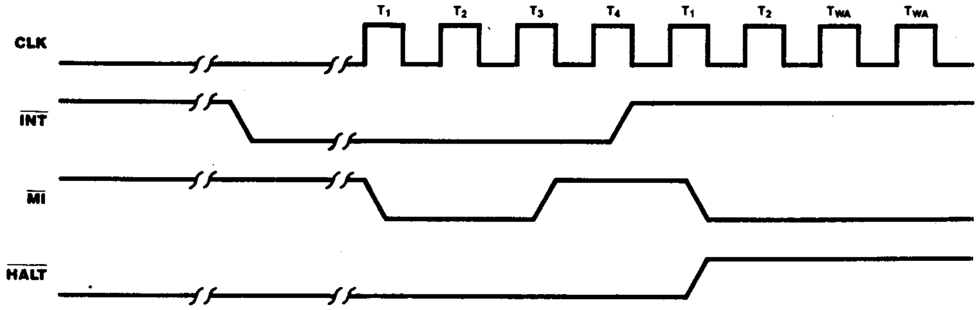


Figure 14c.

Figure 13. Power-Down Release

ABSOLUTE MAXIMUM RATINGS

Voltage on V_{CC} with respect to V_{SS} . . . . – 0.3V to + 7V
Voltages on all inputs with respect
to V_{SS} . . . . – 0.3V to V_{CC} + 0.3V
Operating Ambient
Temperature . . . . .See Ordering Information
Storage Temperature . . . . – 65 °C to + 150 °C

Stresses greater than those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

STANDARD TEST CONDITIONS

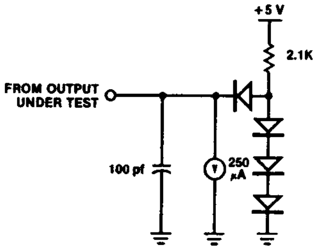
The DC Characteristics and capacitance sections below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND (0V). Positive current flows into the referenced pin.

Available operating temperature ranges are:

- **S = 0°C to +70°C**
  Voltage Supply Range:
  NMOS: +4.75V ≤ VCC ≤ +5.25V
  CMOS: +4.50V ≤ VCC ≤ +5.50V
- E= -40°C to 100°C, +4.50V ≤ VCC ≤ +5.50V

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 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.



DC CHARACTERISTICS (Z84C00/CMOS Z80 CPU)

Symbol	Parameter	Min	Max	Unit	Condition
V_{ILC}	Clock Input Low Voltage	-0.3	0.45	V	
V_{IHC}	Clock Input High Voltage	$V_{CC}-.6$	$V_{CC}+.3$	V	
V_{IL}	Input Low Voltage	-0.3	0.8	V	
V_{IH}	Input High Voltage	2.2	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	$I_{OL} = 2.0$ mA
V_{OH_1}	Output High Voltage	2.4		V	$I_{OH} = -1.6$ mA
V_{OH_2}	Output High Voltage	$V_{CC}-0.8$		V	$I_{OH} = -250\,\mu A$
I_{CC_1}	Power Supply Current **4 MHz**		**20**	**mA**	$V_{CC} = 5V$
	6 MHz		**30**	**mA**	$V_{IH} = V_{CC} - 0.2V$
	8 MHz		**40**	**mA**	$V_{IL} = 0.2V$
	10 MHz		**50**	**mA**	
	20 MHz		100	mA	$V_{cc} = 5V$
I_{CC_2}	Standby Supply Current		10	μA	$V_{CC} = 5V$ $\overline{CLK} = (0)$ $V_{IH} = V_{CC} - 0.2V$ $V_{IL} = 0.2V$
I_{LI}	Input Leakage Current	-10	10	μA	$V_{IN} = 0.4$ to V_{CC}
I_{LO}	3-State Output Leakage Current in Float	-10	10^2	μA	$V_{OUT} = 0.4$ to V_{CC}

1. Measurements made with outputs floating.
2. A_{15}-A_0, D_7-D_0, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$.
3. I_{CC_2} standby supply current is guaranteed only when the supplied clock is stopped at a low level during T_4 of the machine cycle immediately following the execution of a HALT instruction.

CAPACITANCE

Symbol	Parameter	Min	Max	Unit
C_{CLOCK}	Clock Capacitance		**10**	pf
C_{IN}	Input Capacitance		5	pf
C_{OUT}	Output Capacitance		15	pf

$T_A = 25°C$, f = 1 MHz.
Unmeasured pins returned to ground.

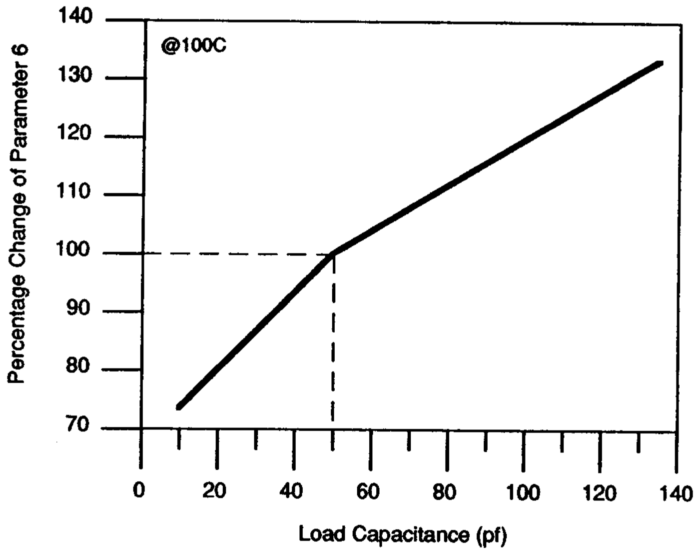


Figure 1. Address Delay Characteristics
(Parameter 6)

DC CHARACTERISTICS (Z8400/NMOS Z80 CPU)

All parameters are tested unless otherwise noted.

Symbol	Parameter	Min	Max	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	−0.3	0.45	V	
V_{IHC}	Clock Input High Voltage	$V_{CC}-.6$	$V_{CC}+.3$	V	
V_{IL}	Input Low Voltage	−0.3	0.8	V	
V_{IH}	Input High Voltage	2.0[1]	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	$I_{OL}=2.0$ mA
V_{OH}	Output High Voltage	2.4[1]		V	$I_{OH}=-250\ \mu A$
I_{CC}	Power Supply Current		200	mA	Note 3
I_{LI}	Input Leakage Current		10	μA	$V_{IN}=0$ to V_{CC}
I_{LO}	3-State Output Leakage Current in Float	−10	10[2]	μA	$V_{OUT}=0.4$ to V_{CC}

1. For military grade parts, refer to the Z80 Military Electrical Specification.
2. A_{15}-A_0, D_7-D_0, $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$.
3. Measurements made with outputs floating.

CAPACITANCE

Guaranteed by design and characterization.

Symbol	Parameter	Min	Max	Unit
C_{CLOCK}	Clock Capacitance		35	pf
C_{IN}	Input Capacitance		5	pf
C_{OUT}	Output Capacitance		15	pf

NOTES:
$T_A = 25°C$, f = 1 MHz.
Unmeasured pins returned to ground.

AC CHARACTERISTICS† (Z8400/NMOS Z80 CPU)

			Z0840004		Z0840006		Z0840008	
Number	**Symbol**	**Parameter**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**
1	TcC	Clock Cycle Time	250*		**162***		125*	
2	TwCh	Clock Pulse Width (High)	110	2000	65	2000	55	2000
3	TwCl	Clock Pulse Width (Low)	110	2000	65	2000	55	2000
4	TfC	Clock Fall Time		30		20		10
5	TrC	Clock Rise Time		30		20		10
6	TdCr(A)	Clock ↑ to Address Valid Delay		110		90		80
7	TdA(MREQf)	Address Valid to MREQ ↓ Delay	65*		35*		20*	
8	TdCf(MREQf)	Clock ↓ to MREQ ↓ Delay		85		70		60
9	TdCr(MREQr)	Clock ↑ to MREQ ↑ Delay		85		70		60
10	TwMREQh	MREQ Pulse Width (High)	110*††		65*††		45*††	
11	TwMREQl	MREQ Pulse Width (Low)	220*††		135*††		100*††	
12	TdCf(MREQr)	Clock ↓ to MREQ ↑ Delay		85		70		60
13	TdCf(RDf)	Clock ↓ to RD ↓ Delay		95		80		70
14	TdCr(RDr)	Clock ↑ to RD ↑ Delay		85		70		60
15	TsD(Cr)	Data Setup Time to Clock ↑	35		30		30	
16	ThD(RDr)	Data Hold Time to RD ↑		0		0		0
17	TsWAIT(Cf)	WAIT Setup Time to Clock ↓	70		60		50	
18	ThWAIT(Cf)	WAIT Hold Time after Clock ↓		0		0		0
19	TdCr(M1f)	Clock ↑ to M1 ↓ Delay		100		80		70
20	TdCr(M1r)	Clock ↑ to M1 ↑ Delay		100		80		70
21	TdCr(RFSHf)	Clock ↑ to RFSH ↓ Delay		130		110		95
22	TdCr(RFSHr)	Clock ↑ to RFSH ↑ Delay		120		100		85
23	TdCf(RDr)	Clock ↓ to RD ↑ Delay		85		70		60
24	TdCr(RDf)	Clock ↑ to RD ↓ Delay		85		70		60
25	TsD(Cf)	Data Setup to Clock ↓ during M$_2$, M$_3$, M$_4$, or M$_5$ Cycles	50		40		30	
26	TdA(IORQf)	Address Stable prior to IORQ ↓	180*		110*		75*	
27	TdCr(IORQf)	Clock ↑ to IORQ ↓ Delay		75		65		55
28	TdCf(IORQr)	Clock ↓ to IORQ ↑ Delay		85		70		60
29	TdD(WRf)	Data Stable prior to WR ↓	80*		25*		5*	
30	TdCf(WRf)	Clock ↓ to WR ↓ Delay		80		70		60
31	TwWR	WR Pulse Width	220*		135*		100*	
32	TdCf(WRr)	Clock ↓ to WR ↑ Delay		80		70		60
33	TdD(WRf)	Data Stable prior to WR ↓	−10*		−55*		55*	
34	TdCr(WRf)	Clock ↑ to WR ↓ Delay		65		60		55
35	TdWRr(D)	Data Stable from WR ↑	60*		30*		15*	
36	TdCf(HALT)	Clock ↓ to HALT ↑ or ↓		300		260		225
37	TwNMI	NMI Pulse Width	80		70		60*	
38	TsBUSREQ(Cr)	BUSREQ Setup Time to Clock ↑	50		50		40	

*For clock periods other than the minimums shown, calculate parameters using the table on the following page. Calculated values above assumed TrC = TfC = 20 ns.

†Units in nanoseconds (ns).

†† For loading ≥ 50 pf., Decrease width by 10 ns for each additional 50 pf.