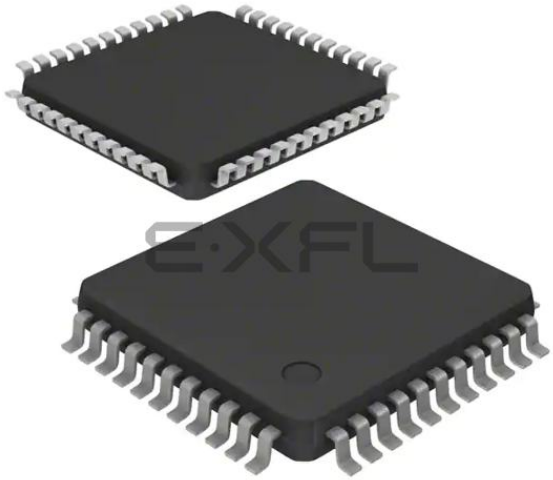


Welcome to E-XFL.COM

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	Z80
Number of Cores/Bus Width	1 Core, 8-Bit
Speed	10MHz
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/z84c0010fec

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

Maskable Interrupt ($\overline{INT}$). 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 $\overline{BUSREQ}$ is not active) a special interrupt processing cycle begins. This is a special fetch ($\overline{M1}$) cycle in which $\overline{IORQ}$ becomes active rather than $\overline{MREQ}$, as in a normal $\overline{M1}$ cycle. In addition, this special $\overline{M1}$ cycle is automatically extended by two $\overline{WAIT}$ 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 $\overline{NMI}$. 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

8-BIT LOAD GROUP (Continued)

Mnemonic	Symbolic Operation	S	Z	H	P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
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	X	IFF	0	•	11	101	101	ED	2	2	9	
									01	010	111	57				
LD A, R	A ← R	‡	‡	X	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	S	Z	H	P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments	
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)	•	•	X	X	•	•	•	00	101	010	2A	3	5	16	
	L ← (nn)															
									← n →							
									← n →							
LD dd, (nn)	dd_H ← (nn+1)	•	•	X	X	•	•	•	11	101	101	ED	4	6	20	
	dd_L ← (nn)								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)

Mnemonic	Symbolic Operation	S	Z	H	P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T 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				
PUSH qq	$(SP-2) \leftarrow qq_L$	•	•	X	X	•	•	11	qq0	101		1	3	11	
	$(SP-1) \leftarrow qq_H$														
	$SP \rightarrow SP-2$														
PUSH IX	$(SP-2) \leftarrow IX_L$	•	•	X	X	•	•	11	011	101	DD	2	4	15	
	$(SP-1) \leftarrow IX_H$							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	
	$qqL \leftarrow (SP)$														
	$SP \rightarrow SP+2$														
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				
	$SP \rightarrow SP+2$														

qq	Pair
00	BC
01	DE
10	HL
11	AF

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)

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

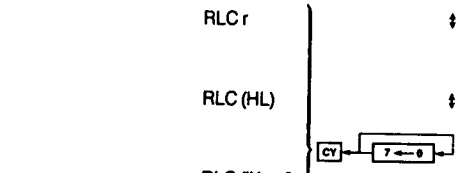
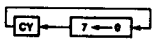
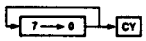
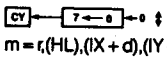
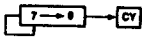
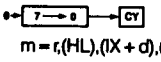
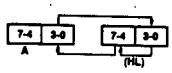
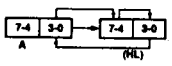
NOTE: m is any of r, (HL), (IX + d), (IY + d) as shown for INC. DEC same format and states as INC. Replace 100 with 101 in opcode.

GENERAL-PURPOSE ARITHMETIC AND CPU CONTROL GROUPS

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
DAA	@	‡	‡	X	‡	X	P	•	‡	00	100	111	27	1	1	4	Decimal adjust accumulator.
CPL	A ← Ā	•	•	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 ← C̄Y	•	•	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	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	8	
	mode 0									01	000	110	46				
IM 1	Set interrupt	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	8	
	mode 1									01	010	110	56				
IM 2	Set interrupt	•	•	X	•	X	•	•	•	11	101	101	ED	2	2	8	
	mode 2									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.

ROTATE AND SHIFT GROUP (Continued)

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

BIT SET, RESET AND TEST GROUP

Mnemonic	Symbolic Operation	S	Z	H	P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
BIT b, r	$Z \leftarrow r_b$	X	‡	X	X	0	•	11	001	011	CB	2	2	8	r Reg.
								01	b	r					000 B
BIT b, (HL)	$Z \leftarrow (HL)_b$	X	‡	X	X	0	•	11	001	011	CB	2	3	12	001 C
								01	b	110					010 D
BIT b,(IX+d)	$Z \leftarrow (IX+d)_b$	X	‡	X	X	0	•	11	011	101	DD	4	5	20	011 E
								11	001	011	CB				100 H
									←d→						101 L
								01	b	110					111 A
BIT b, (IY+d)	$Z \leftarrow (IY+d)_b$	X	‡	X	X	0	•	11	111	101	FD	4	5	20	b Bit Tested
								11	001	011	CB				000 0
									←d→						001 1
								01	b	110					010 2
SET b, r	$r_b \leftarrow 1$	•	•	X	X	•	•	11	001	011	CB	2	2	8	011 3
								[11]	b	r					100 4
SET b, (HL)	$(HL)_b \leftarrow 1$	•	•	X	X	•	•	11	001	011	CB	2	4	15	101 5
								[11]	b	110					110 6
SET b, (IX+d)	$(IX+d)_b \leftarrow 1$	•	•	X	X	•	•	11	011	101	DD	4	6	23	111 7
								11	001	011	CB				
									←d→						
								[11]	b	110					
SET b, (IY+d)	$(IY+d)_b \leftarrow 1$	•	•	X	X	•	•	11	111	101	FD	4	6	23	
								11	001	011	CB				
									←d→						
								[11]	b	110					
RES b, m	$m_b \leftarrow 0$ m≡r, (HL), (IX+d), (IY+d)	•	•	X	X	•	•	[10]							To form new opcode replace [11] of SET b, s with [10]. Flags and time states for SET instruction.

NOTE: The notation m_b indicates location m, bit b (0 to 7).

JUMP GROUP

Mnemonic	Symbolic Operation	S	Z	H	P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T 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.

INPUT AND OUTPUT GROUP

Mnemonic	Symbolic Operation	S	Z		H		P/V	N	C	76	543	210	Hex	No. of Bytes	No. of M Cycles	No. of T States	Comments
IN A, (n)	A ← (n)	•	•	X	•	X	•	•	•	11	011	01	DB	2	3	11	n to $A_0 \sim A_7$
											← n →						Acc. to $A_8 \sim A_{15}$
IN r, (C)	r ← (C)	‡	‡	X	‡	X	P	0	•	11	101	101	ED	2	3	12	C to $A_0 \sim A_7$
	if r = 110 only									01	r	000					B to $A_8 \sim A_{15}$
	the flags will																
	be affected																
			①														
INI	(HL) ← (C)	X	‡	X	X	X	X	1	X	11	101	101	ED	2	4	16	C to $A_0 \sim A_7$
	B ← B − 1									10	100	010	A2				B to $A_8 \sim A_{15}$
	HL ← HL + 1		②														
INIR	(HL) ← (C)	X	1	X	X	X	X	1	X	11	101	101	ED	2	5	21	C to $A_0 \sim A_7$
	B ← B − 1									10	110	010	B2		(If B≠0)		B to $A_8 \sim A_{15}$
	HL ← HL + 1													2	4	16	
	Repeat until														(If B = 0)		
	B = 0																
			①														
IND	(HL) ← (C)	X	‡	X	X	X	X	1	X	11	101	101	ED	2	4	16	C to $A_0 \sim A_7$
	B ← B − 1									10	101	010	AA				B to $A_8 \sim A_{15}$
	HL ← HL − 1		②														
INDR	(HL) ← (C)	X	1	X	X	X	X	1	X	11	101	101	ED	2	5	21	C to $A_0 \sim A_7$
	B ← B − 1									10	111	010	BA		(If B≠0)		B to $A_8 \sim A_{15}$
	HL ← HL − 1													2	4	16	
	Repeat until														(If B = 0)		
	B = 0																
OUT (n), A	(n) ← A	•	•	X	•	X	•	•	•	11	010	011	D3	2	3	11	n to $A_0 \sim A_7$
											← n →						Acc. to $A_8 \sim A_{15}$
OUT (C), r	(C) ← r	•	•	X	•	X	•	•	•	11	101	101	ED	2	3	12	C to $A_0 \sim A_7$
										01	r	001					B to $A_8 \sim A_{15}$
			①														
OUTI	(C) ← (HL)	X	‡	X	X	X	X	1	X	11	101	101	ED	2	4	16	C to $A_0 \sim A_7$
	B ← B − 1									10	100	011	A3				B to $A_8 \sim A_{15}$
	HL ← HL + 1		②														
OTIR	(C) ← (HL)	X	1	X	X	X	X	1	X	11	101	101	ED	2	5	21	C to $A_0 \sim A_7$
	B ← B − 1									10	110	011	B3		(If B≠0)		B to $A_8 \sim A_{15}$
	HL ← HL + 1													2	4	16	
	Repeat until														(If B = 0)		
	B = 0																
			①														
OUTD	(C) ← (HL)	X	‡	X	X	X	X	1	X	11	101	101	ED	2	4	16	C to $A_0 \sim A_7$
	B ← B − 1									10	101	011	AB				B to $A_8 \sim A_{15}$
	HL ← HL − 1																
			②														
OTDR	(C) ← (HL)	X	1	X	X	X	X	1	X	11	101	101	ED	2	5	21	C to $A_0 \sim A_7$
	B ← B − 1									10	111	011			(If B≠0)		B to $A_8 \sim A_{15}$
	HL ← HL − 1													2	4	16	
	Repeat until														(If B = 0)		
	B = 0																

NOTES: ① If the result of B − 1 is zero, the Z flag is set; otherwise it is reset.
② Z flag is set upon instruction completion only.

CPU REGISTERS

Figure 4 shows three groups of registers within the CPU. The first group consists of duplicate sets of 8-bit registers: a principal set and an alternate set [designated by ′ (prime), e.g., A′]. Both sets consist of the Accumulator register, the Flag register, and six general-purpose registers. Transfer of data between these duplicate sets of registers is accomplished by use of "Exchange" instructions. The result is faster response to interrupts and easy, efficient implementation of such versatile programming techniques as background-foreground data processing. The second set of registers consists of six registers with assigned functions. These are the I (Interrupt register), the R (Refresh register), the IX and IY (Index registers), the SP (Stack Pointer), and the PC (Program Counter). The third group consists of two interrupt status flip-flops, plus an additional pair of flip-flops which assists in identifying the interrupt mode at any particular time. Table 1 provides further information on these registers.

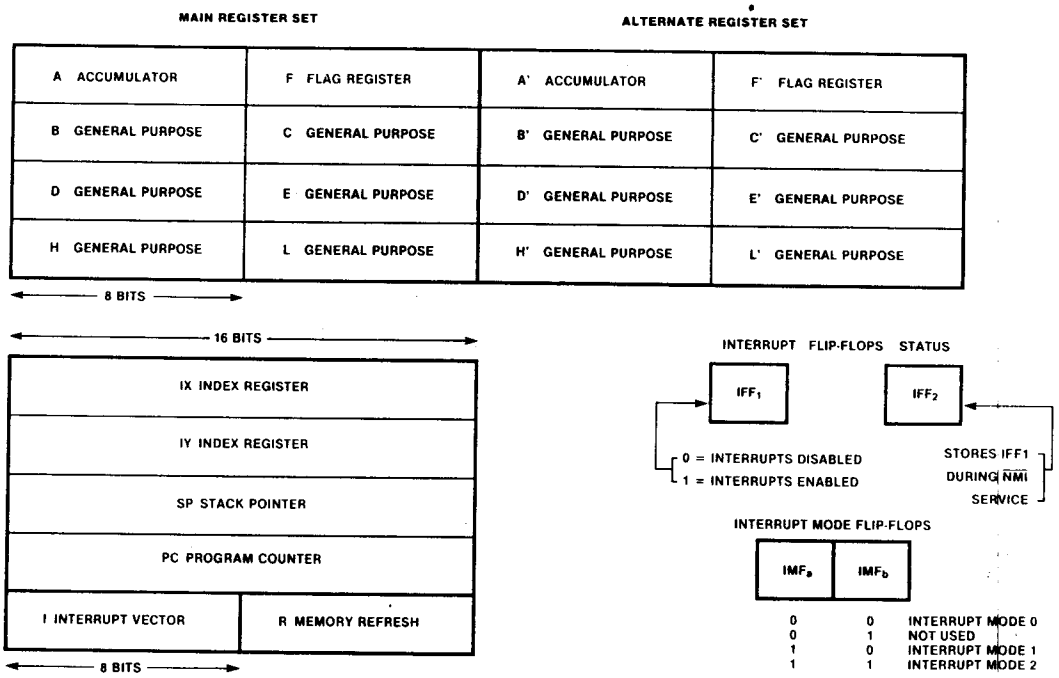


Figure 4. CPU Registers

INTERRUPTS: GENERAL OPERATION

The CPU accepts two interrupt input signals: $\overline{NMI}$ and $\overline{INT}$. The $\overline{NMI}$ is a non-maskable interrupt and has the highest priority. $\overline{INT}$ is a lower priority interrupt and it requires that interrupts be enabled in software in order to operate. $\overline{INT}$ can be connected to multiple peripheral devices in a wired-OR configuration.

The Z80 has a single response mode for interrupt service on the non-maskable interrupt. The maskable interrupt, $\overline{INT}$, has three programmable response modes available. These are:

- Mode 0 — similar to the 8080 microprocessor.
- Mode 1 — Peripheral Interrupt service, for use with non-8080/Z80 systems.
- **Mode 2 - a vectored interrupt scheme, usually daisy-chained, for use with the Z80 Family and compatible peripheral devices.**

The CPU services interrupts by sampling the $\overline{NMI}$ and $\overline{INT}$ signals at the rising edge of the last clock of an instruction. Further interrupt service processing depends upon the type of interrupt that was detected. Details on interrupt responses are shown in the CPU Timing Section.

Non-Maskable Interrupt ($\overline{NMI}$). The nonmaskable interrupt cannot be disabled by program control and therefore will be accepted at all times by the CPU. $\overline{NMI}$ is usually reserved for servicing only the highest priority type interrupts, such as that for orderly shutdown after power

Memory Read or Write Cycles. Figure 6 shows the timing of memory read or write cycles other than an opcode fetch ($\overline{M1}$) cycle. The $\overline{MREQ}$ and $\overline{RD}$ signals function exactly as in the fetch cycle. In a memory write cycle, $\overline{MREQ}$ also becomes active when the address bus is stable. The $\overline{WR}$ line is active when the data bus is stable, so that it can be used directly as an R/$\overline{W}$ pulse to most semiconductor memories.

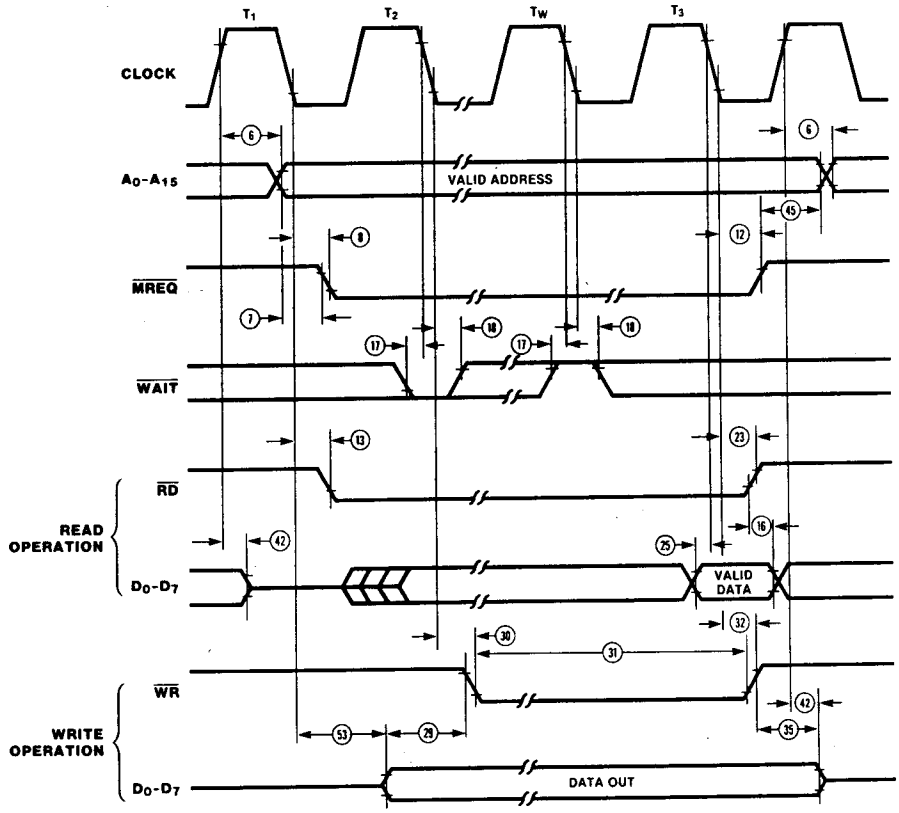
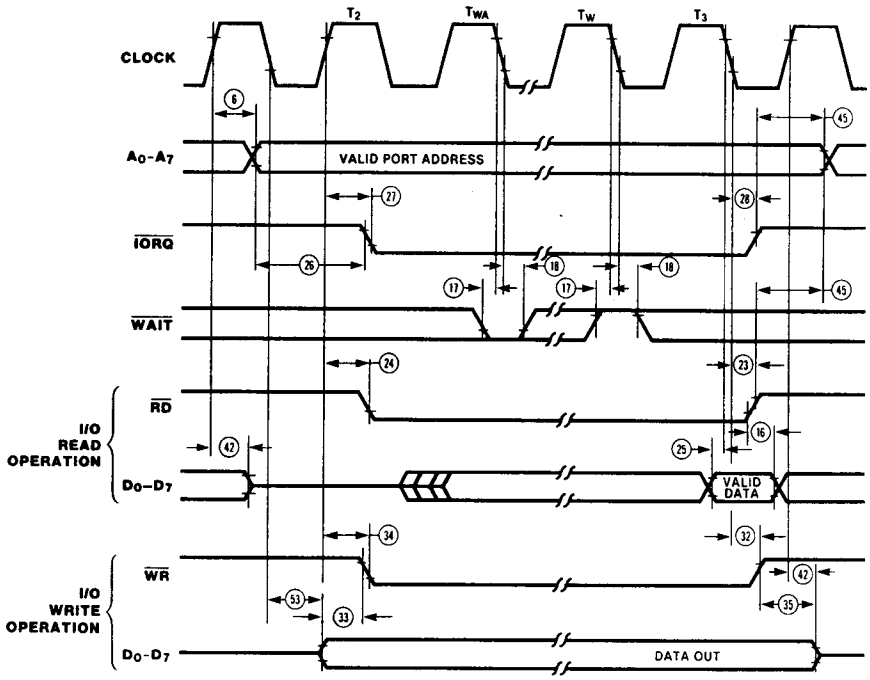


Figure 6. Memory Read or Write Cycles

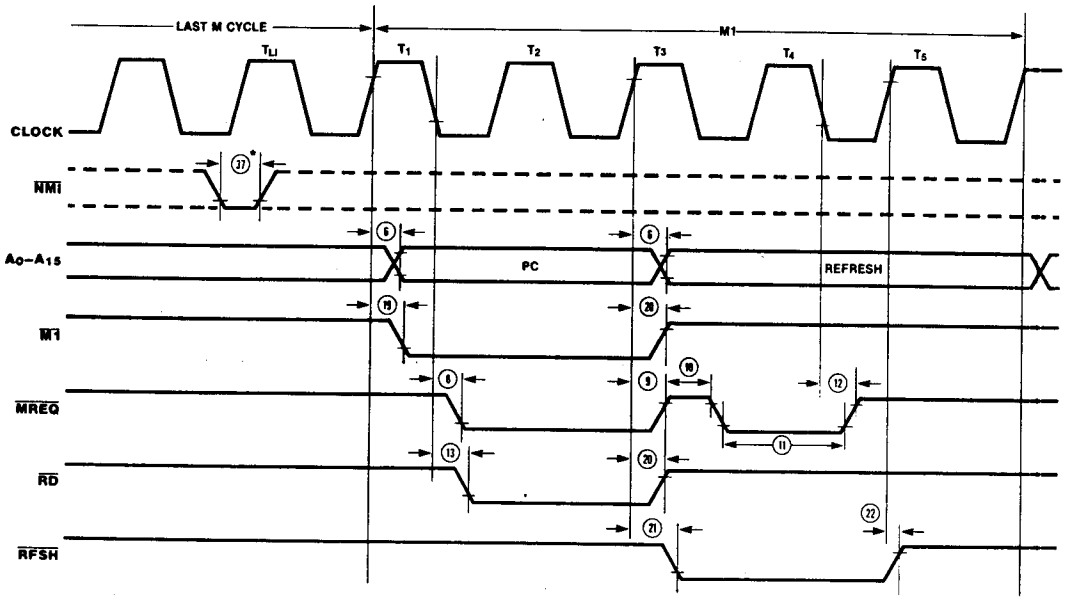
Input or Output Cycles. Figure 7 shows the timing for an I/O read or I/O write operation. During I/O operations, the CPU automatically inserts a single Wait state (T_{WA}). This extra Wait state allows sufficient time for an I/O port to decode the address from the port address lines.



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

Figure 7. Input or Output Cycles

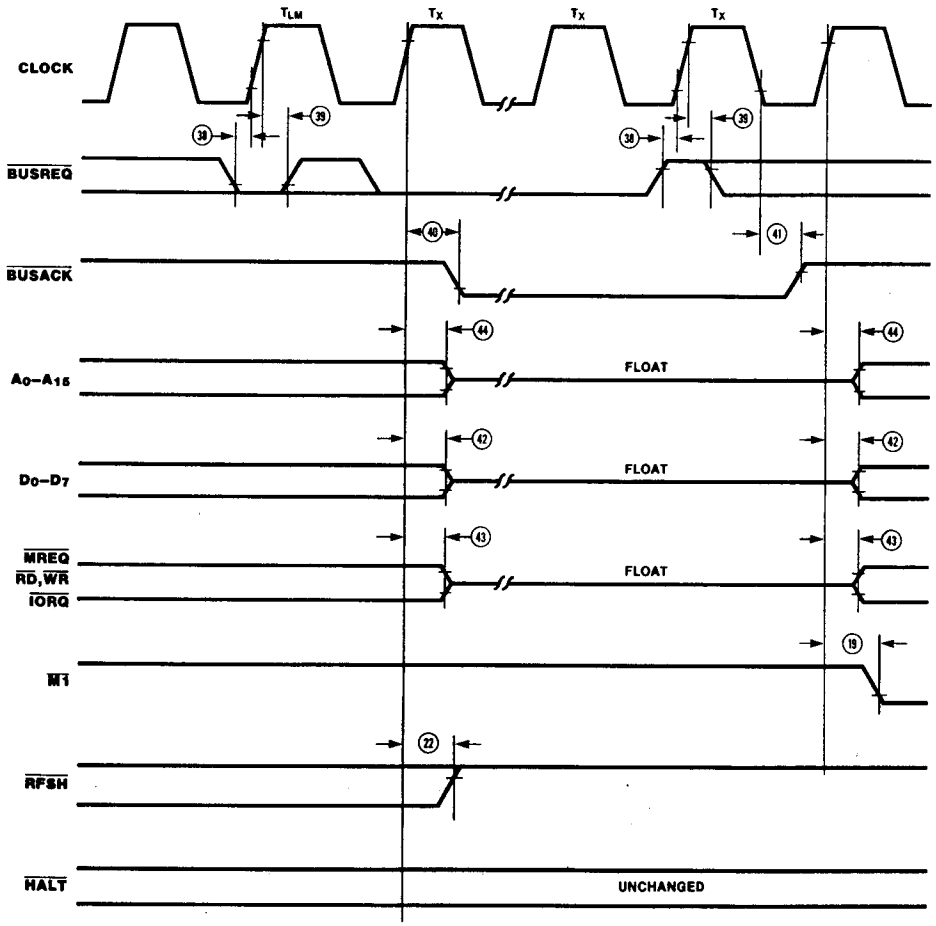
Non-Maskable Interrupt Request Cycle. $\overline{NMI}$ is sampled at the same time as the maskable interrupt input $\overline{INT}$ but has higher priority and cannot be disabled under software control. The subsequent timing is similar to that of a normal memory read operation except that data put on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



*Although $\overline{NMI}$ is an asynchronous input, to guarantee its being recognized on the following machine cycle, $\overline{NMI}$'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 $\overline{BUSREQ}$ with the rising edge of the last clock period of any machine cycle (Figure 10). If $\overline{BUSREQ}$ is active, the CPU sets its address, data, and $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$ 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 $\overline{NMI}$ or $\overline{INT}$) or a $\overline{RESET}$ signal must be applied to the CPU after the system clock is supplied in order to release the power-down state.

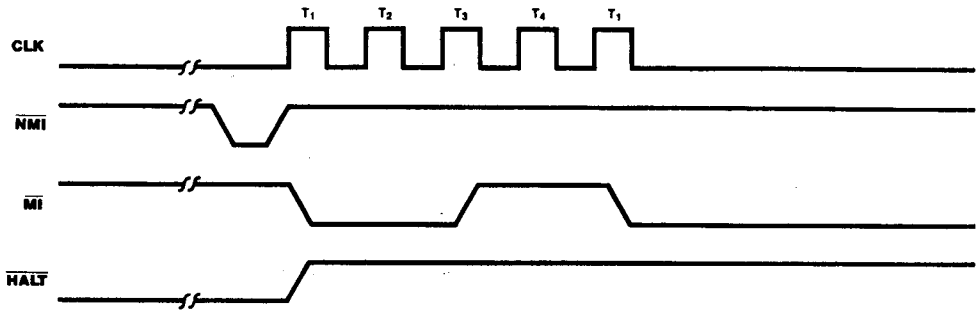


Figure 14a.

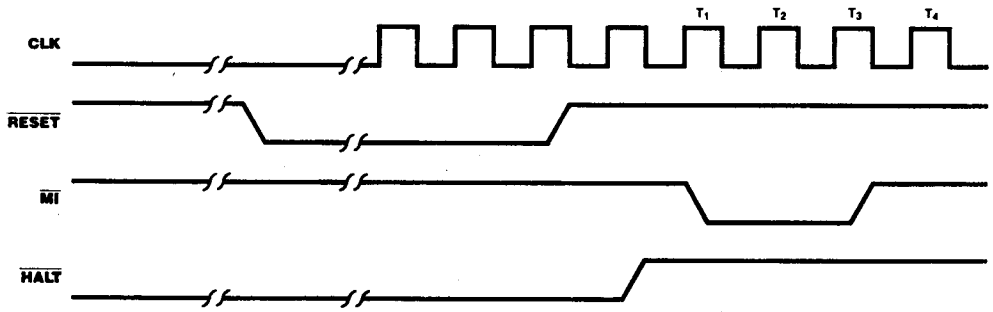


Figure 14b.

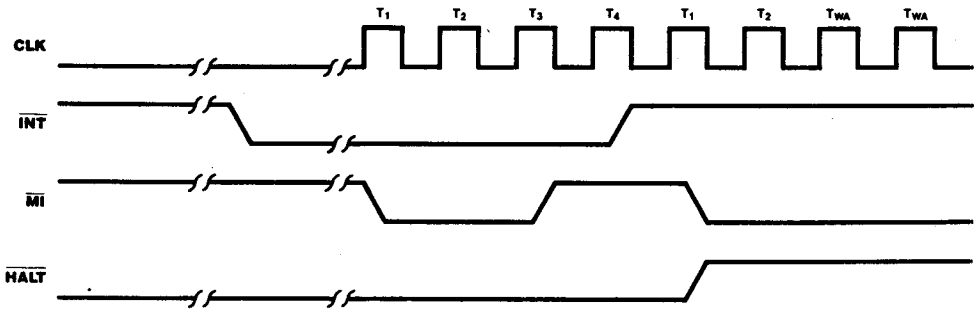


Figure 14c.

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

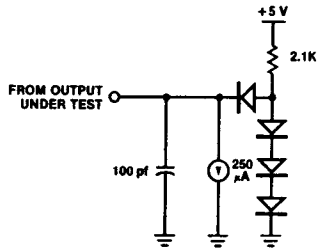
The DC Characteristics and capacitance sections below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND (0V). Positive current flows into the referenced pin.

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 μA
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
					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.

AC CHARACTERISTICS† (Z84C00/CMOS Z80 CPU)

V_{cc}=5.0V ± 10%, unless otherwise specified

No	Symbol	Parameter	Z84C0004** Min	Z84C0004** Max	Z84C0006 Min	Z84C0006 Max	Z84C0008 Min	Z84C0008 Max	Z84C0010 Min	Z84C0010 Max	Z84C0020[1] Min	Z84C0020[1] Max	Unit	Note
1	TcC	Clock Cycle time	250*	DC	162*	DC	125*	DC	100*	DC	50*	DC	nS	
2	TwCh	Clock Pulse width (high)	110	DC	65	DC	55	DC	40	DC	20	DC	nS	
3	TwCl	Clock Pulse width (low)	110	DC	65	DC	55	DC	40	DC	20	DC	nS	
4	TfC	Clock Fall time		30		20		10		10		10	nS	
5	TrC	Clock Rise time		30		20		10		10		10	nS	
6	TdCr(A)	Address vaild from Clock Rise		110		90		80		65		57	nS	[2]
7	TdA(MREQf)	Address valid to /MREQ Fall	65*		35*		20*		5*		-15*		nS	
8	TdCf(MREQf)	Clock Fall to /MREQ Fall delay		85		70		60		55		40	nS	
9	TdCr(MREQr)	Clock Rise to /MREQ Rise delay		85		70		60		55		40	nS	
10	TwMREQh	/MREQ pulse width (High)	110*		65*		45**		30*		10*		nS	[3]
11	TwMREQl	/MREQ pulse width (low)	220*		132*		100*		75*		25*		nS	[3]
12	TdCf(MERQr)	Clock Fall to /MREQ Rise delay		85		70		60		55		40	nS	
13	TdCf(RDf)	Clock Fall to /RD Fall delay		95		80		70		65		40	nS	
14	TdCr(RDr)	Clock Rise to /RD Rise delay		85		70		60		55		40	nS	
15	TsD(Cr)	Data setup time to Clock Rise	35		30		30		25		12		nS	
16	ThD(RDr)	Data hold time after /RD Rise	0		0		0		0		0		nS	
17	TsWAIT(Cf)	/WAIT setup time to Clock Fall	70		60		50		20		7.5		nS	
18	ThWAIT(Cf)	/WAIT hold time after Clock Fall	10		10		10		10		10		nS	
19	TdCr(M1f)	Clock Rise to /M1 Fall delay		100		80		70		65		45	nS	
20	TdCr(M1r)	Clock Rise to /M1 Rise delay		100		80		70		65		45	nS	
21	TdCr(RFSHf)	Clock Rise to /RFSH Fall delay		130		110		95		80		60	nS	
22	TdCr(RFSHr)	Clock Rise to /RFSH Rise delay		120		100		85		80		60	nS	
23	TdCf(RDr)	Clock Fall to /RD Rise delay		85		70		60		55		40	nS	
24	TdCr(RDf)	Clock Rise to /RD Fall delay		85		70		60		55		40	nS	
25	TsD(Cf)	Data setup to Clock Fall during M2, M3, M4 or M5 cycles	50		40		30		25		12		nS	
26	TdA(IORQf)	Address stable prior to /IORQ Fall	180*		107*		75*		50*		0*		nS	
27	TdCr(IORQf)	Clock Rise to /IORQ Fall delay		75		65		55		50		40	nS	
28	TdCf(IORQr)	Clock Fall to /IORQ Rise delay		85		70		60		55		40	nS	
29	TdD(WRf)Mw	Data stable prior to /WR Fall	80*		22*		5*		40*		-10*		nS	
30	TdCf(WRf)	Clock Fall to /WR Fall delay		80		70		60		55		40	nS	
31	TwWR	/WR pulse width	220*		132*		100*		75*		25*		nS	
32	TdCf(WRr)	Clock Fall to /WR Rise delay		80		70		60		55		40	nS	
33	TdD(WRf)IO	Data stable prior to /WR Fall	-10*		-55*		-55*		-10*		-30*		nS	
34	TdCr(WRf)	Clock Rise to /WR Fall delay		65		60		60		50		40	nS	
35	TdWRr(D)	Data stable from /WR Rise	60*		30*		15*		10*		0*		nS	
36	TdCf(HALT)	Clock Fall to /HALT 'L' or 'H'		300		260		225		90		70	nS	
37	TwNMI	/NMI pulse width	80		60		60		60		60		nS	
38	TsBUSREQ (Cr)	/BUSREQ setup time to Clock Rise	50		50		40		30		15		nS	

*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.**

**4 MHz CMOS Z80 is obsoleted and replaced by 6 MHz

AC CHARACTERISTICS† (Z84C00/CMOS Z80 CPU; Continued)

V_{CC}=5.0V ± 10%, unless otherwise specified

No	Symbol	Parameter	Z84C0004** Min	Z84C0004** Max	Z84C0006 Min	Z84C0006 Max	Z84C0008 Min	Z84C0008 Max	Z84C0010 Min	Z84C0010 Max	Z84C0020[1] Min	Z84C0020[1] Max	Unit	Note
39	ThBUSREQ (Cr)	/BUSREQ hold time after Clock Rise	10		10		10		10		10		nS	
40	TdCr (BUSACKf)	Clock Rise to /BASACK Fall delay		100		90		80		75		40	nS	
41	TdCf (BUSACKr)	Clock Fall to /BASACK Rise delay		100		90		80		75		40	nS	
42	TdCr(Dz)	Clock Rise to Data float delay		90		80		70		65		40	nS	
43	TdCr(CTz)	Clock Rise to Control Outputs Float Delay (/MREQ, /IORQ, /RD and /WR)		80		70		60		65		40	nS	
44	TdCr(Az)	Clock Rise to Address float delay		90		80		70		75		40	nS	
45	TdCTr(A)	Address Hold time from /MREQ, /IORQ, /RD or /WR	80*		35*		20*		20*		0*		nS	
46	TsRESET(Cr)	/RESET to Clock Rise setup time	60		60		45		40		15		nS	
47	ThRESET(Cr)	/RESET to Clock Rise Hold time	10		10		10		10		10		nS	
48	TsINTf(Cr)	/INT Fall to Clock Rise Setup Time	80		70		55		50		15		nS	
49	ThINTr(Cr)	/INT Rise to Clock Rise Hold Time	10		10		10		10		10		nS	
50	TdM1f (IORQf)	/M1 Fall to /IORQ Fall delay	565*		359*		270*		220*		100*		nS	
51	TdCf(IORQf)	/Clock Fall to /IORQ Fall delay		85		70		60		55		45	nS	
52	TdCf(IORQr)	Clock Rise to /IORQ Rise delay		85		70		60		55		45	nS	
53	TdCf(D)	Clock Fall to Data Valid delay		150		130		115		110		75	nS	

Notes:

* For Clock periods other than the minimum shown, calculate parameters using the following table. Calculated values above assumed TrC = TfC = maximum.

** 4 MHz CMOS Z80 is obsoleted and replaced by 6 MHz

[1] Z84C0020 parameters are guuaranteed with 50pF load Capacitance.

[2] If Capacitive Load is other than 50pF, please use Figure 1. to calculate the value.

[3] Increasing delay by 10nS for each 50pF increase in loading, 200pF max for data lines, and 100pF for control lines.

FOOTNOTES TO AC CHARACTERISTICS

No	Symbol	Parameter	Z84C0004**	Z84C0006	Z84C0008	Z84C0010	Z84C0020
1	TcC	TwCh + TwCl + TrC + TfC					
7	TdA(MREQf)	TwCh + TfC	-65	-50	-45	-45	-45
10	TwMREQh	TwCh + TfC	-20	-20	-20	-20	-20
11	TwMREQl	TcC	-30	-30	-25	-25	-25
26	TdA(IORQf)	TcC	-70	-55	-50	-50	-50
29	TdD(WRf)	TcC	-170	-140	-120	-60	-60
31	TwWR	TcC	-30	-30	-25	-25	-25
33	TdD(WRf)	TwCl + TrC	-140	-140	-120	-60	-60
35	TdWRr(D)	TwCl + TrC	-70	-55	-50	-40	-25
45	TdCTr(A)	TwCl + TrC	-50	-50	-45	-30	-30
50	TdM1f(IORQf)	2TcC + TwCh + TfC	-65	-50	-45	-30	-30

AC Test Conditions: V_{IH} = 2.0 V, V_{IL} = 0.8 V; V_{OH} = 1.5 V, V_{OL} = 1.5 V; V_{IHC} = V_{CC} −0.6 V, V_{ILC} = 0.45 V; FLOAT = ±0.5 V

AC CHARACTERISTICS† (Z8400/NMOS Z80 CPU)

Number	Symbol	Parameter	Z0840004 Min	Z0840004 Max	Z0840006 Min	Z0840006 Max	Z0840008 Min	Z0840008 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 $\overline{MREQ}$ ↓ Delay	65*		35*		20*	
8	TdCf(MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay		85		70		60
9	TdCr(MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay		85		70		60
10	TwMREQh	$\overline{MREQ}$ Pulse Width (High)	110*‡‡		65*‡‡		45*‡‡	
11	TwMREQl	$\overline{MREQ}$ Pulse Width (Low)	220*‡‡		135*‡‡		100*‡‡	
12	TdCf(MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay		85		70		60
13	TdCf(RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay		95		80		70
14	TdCr(RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay		85		70		60
15	TsD(Cr)	Data Setup Time to Clock ↑	35		30		30	
16	ThD(RDr)	Data Hold Time to $\overline{RD}$ ↑		0		0		0
17	TsWAIT(Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	70		60		50	
18	ThWAIT(Cf)	$\overline{WAIT}$ Hold Time after Clock ↓		0		0		0
19	TdCr(M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay		100		80		70
20	TdCr(M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay		100		80		70
21	TdCr(RFSHf)	Clock ↑ to $\overline{RFSH}$ ↓ Delay		130		110		95
22	TdCr(RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay		120		100		85
23	TdCf(RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay		85		70		60
24	TdCr(RDf)	Clock ↑ to $\overline{RD}$ ↓ 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 $\overline{IORQ}$ ↓	180*		110*		75*	
27	TdCr(IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay		75		65		55
28	TdCf(IORQr)	Clock ↓ to $\overline{IORQ}$ ↑ Delay		85		70		60
29	TdD(WRf)	Data Stable prior to $\overline{WR}$ ↓	80*		25*		5*	
30	TdCf(WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay		80		70		60
31	TwWR	$\overline{WR}$ Pulse Width	220*		135*		100*	
32	TdCf(WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay		80		70		60
33	TdD(WRf)	Data Stable prior to $\overline{WR}$ ↓	−10*		−55*		55*	
34	TdCr(WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay		65		60		55
35	TdWRr(D)	Data Stable from $\overline{WR}$ ↑	60*		30*		15*	
36	TdCf(HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓		300		260		225
37	TwNMI	$\overline{NMI}$ Pulse Width	80		70		60*	
38	TsBUSREQ(Cr)	$\overline{BUSREQ}$ 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.

Customer Support

For answers to technical questions about the product, documentation, or any other issues with Zilog's offerings, please visit Zilog's Knowledge Base at http://www.zilog.com/kb.

For any comments, detail technical questions, or reporting problems, please visit Zilog's Technical Support at http://support.zilog.com.