165220

Zilog

ADVANCED INFORMATION
Product Specification

Z8400/Z84C00 NMOS/CMOS Z80® CPU Central Processiny Uni

January 1989

FEATURES

- The extonsiveinstruction set contains 158 instructions, including the 8080A instruction set as a subset.
- Single 5 volt power supply.
- NMOS version for low cost high Performance solutions, CMOS version for high Performance low power designs.
- NMOS 20840004 - 4 MHz, 20840006 - 6.17 MHz, 20840008 - 8 MHz.

  CMOS Z84C0004 - DC to 4 MHz, Z84C0006 - DC to 6.17 MHz, Z84C0008 - DC to 8 MHz, Z84C0010 - DC to 10 MHz.
- 6 MHz Version can be operated at 6.144 MHz clock.
- The Z80 microproccssors and associatcd family of peripherals can be linked by a vectored interrupt system. This system can bsdaisy-chained to allow implementation of a priority interrupt schemo.
- Duplicate set of both general-purpose and flag rcgistcrs.
- Two sixtccn bit index rcgistcrs.
- Thrcc modes of maskable interrupts:
  Mode 0—8080A similar;
  Mode 1 -Non-Z80 environment,location38H;
  Mode 2—Z80 family peripherals, vectorcd interrupts.
- On-chip dynamic memory refresh counter.

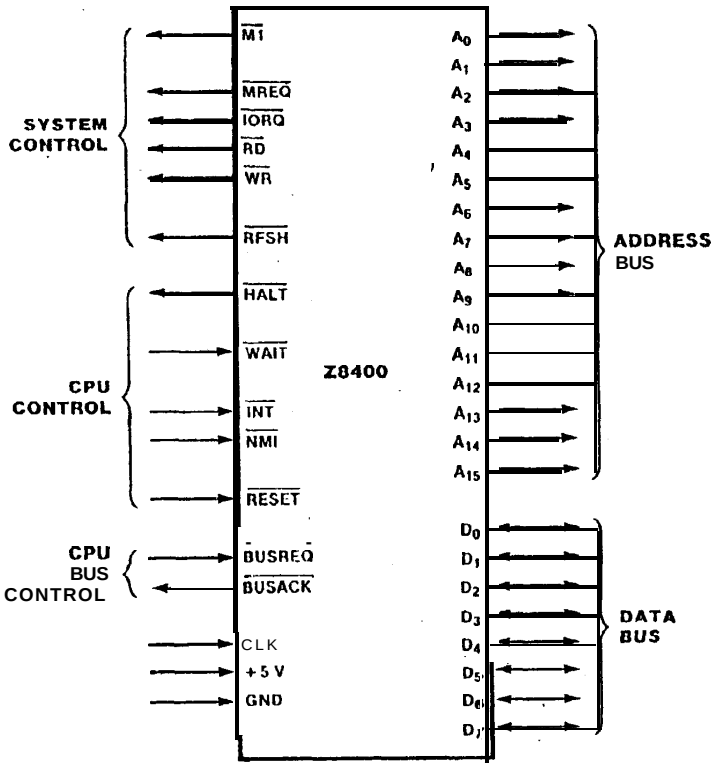


Figure 1. Pin Functions

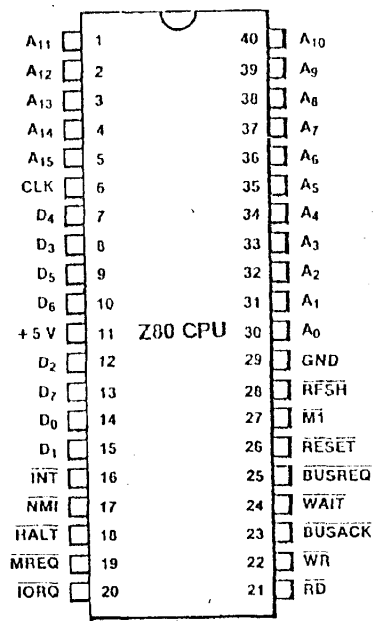
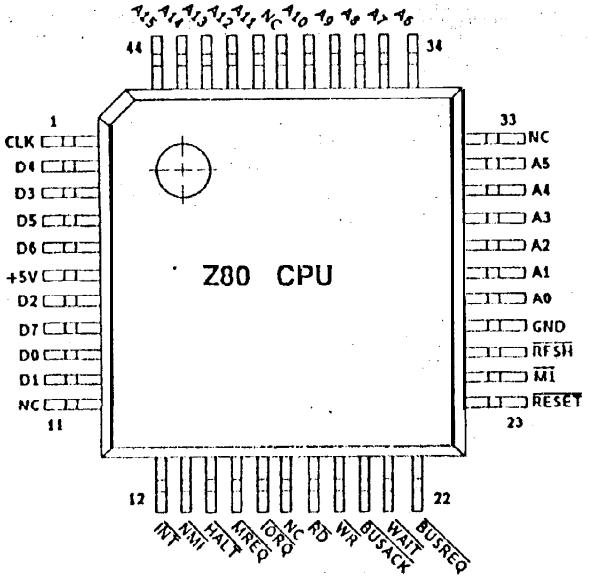
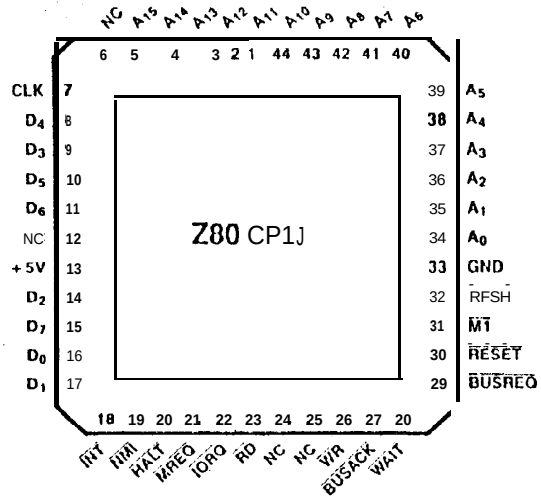


Figure 2. 40-pin Dual-In-Lim (DIP), Pin Assignments



44 pin Quad Flat Pack (QFP), Pin Assignments (Only available for 84C00)



Fiaurc 2b. 44-Pin Chip Cnrrier Pin Assianmcnts

GENERAL DESCRIPTION

The CPUs arc fourth-gcncration enhanced microproccssors with exccptional computational power. They offer higher system throughput and morc elficient memory utilization than comparablc second- and third-generation microproccssors. The internal rcgislers contain 208 bits of read/writc mcmory that arc accessible to thc programmer. These rcgislers include two sets of six general-purpose registers which may be uscd individually as cithcr 8-bit rcgisters or as 1 G-bit register pairs. In addition, thcrc are two sets of accumulator and flag rcgisters. A group of "Exchange" instructions makes either set of main or alternate registers accessibie to thc programmcr. The alternate sct allows Operation in foreground-background mode or it may be reservcd for vcry fast interrupt rcsponse.

The CPU also contains a Stack Pointer, Program Counter, two index rcgisters, a Refresh rcgistcr (countcr), and an Interrupt rcgistcr. The CPU is easy to incorporatc into a system since it rcquircs only a single +5V power source. All output Signals arc fully clecodcd and timed to conirol Standard mcmory or peripheral circuits; thc CPU is supportod by an cxtcnsivc family of peripheral controllers. The internal block diagram (Figure 3) shows thc prirnary functions of thc processors. Subsequent text provides morc detail on the I/O controller family, registers, instruction sct, interrupts and daisy chaining, and CPU timing.

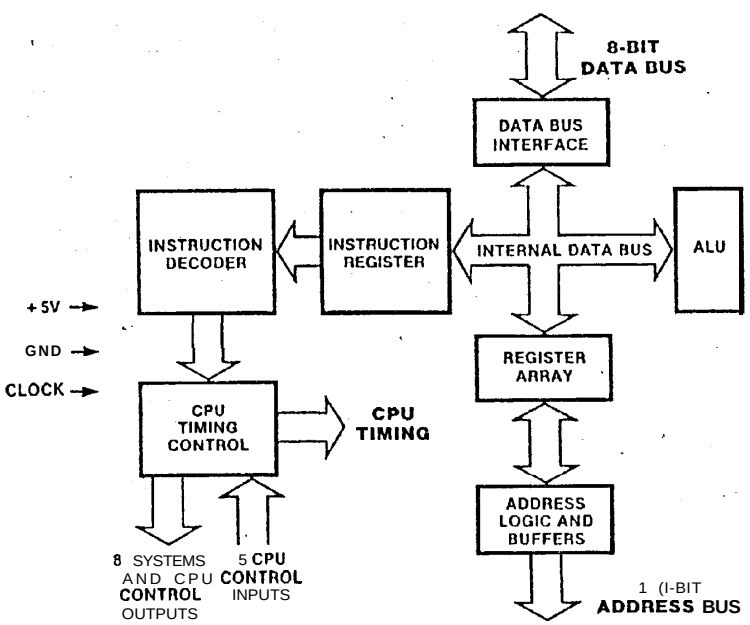


Figure 3. Z80C CPU Block Diagram

CPU REGISTERS

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

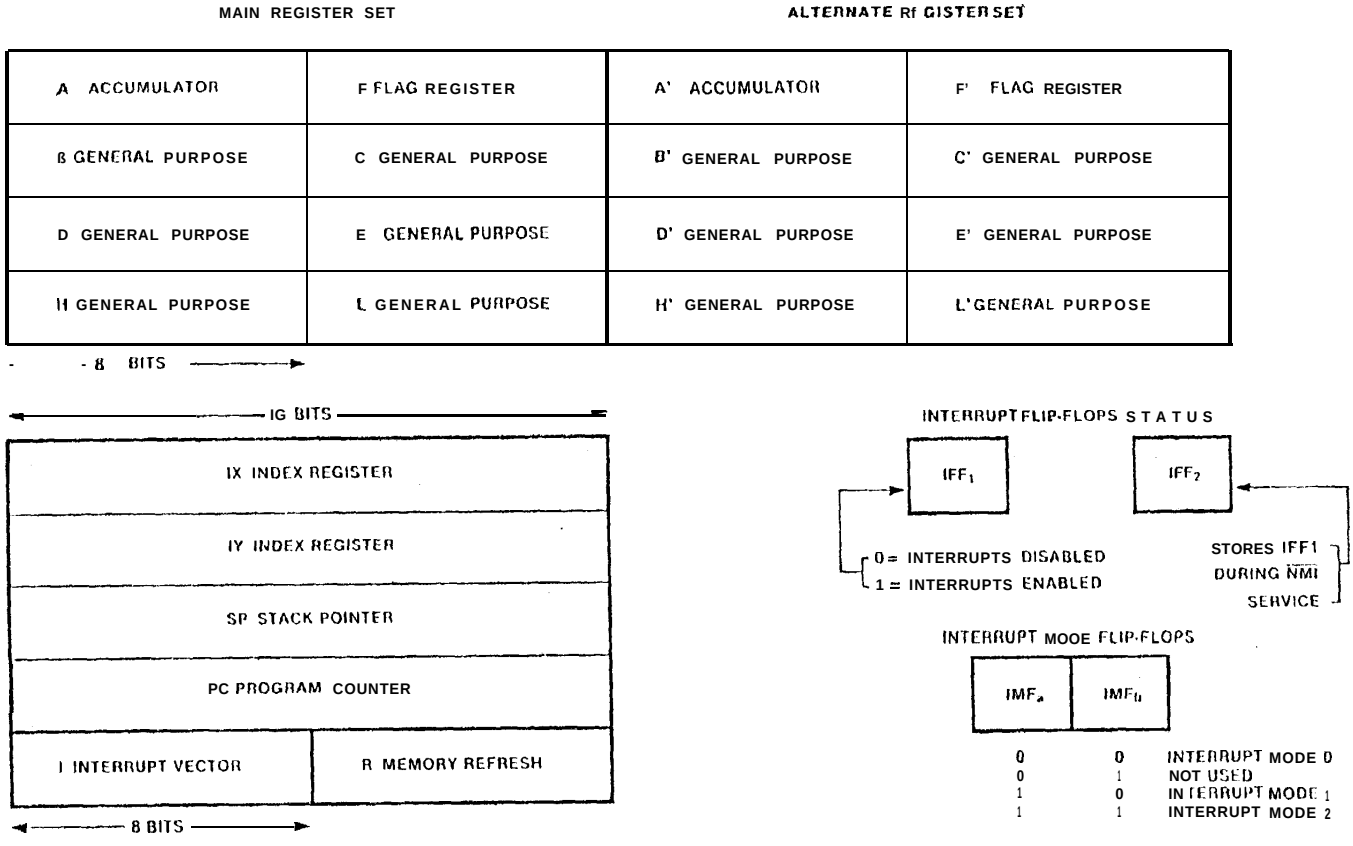


Figure 4. CPU Registers

INTERRUPTS: GENERAL OPERATION

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

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

- Mode 0 — similar to the 8080 microproccssor.
- 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 280 Family and compatibtc peripheral doviccs.

The CPU services interrupts by sampling thc $\overline{NMI}$ and $\overline{INT}$ Signals at the rising edge of the last clock of an instruction. Further interrupt service processing depcnds upon thc type of interrupt that was dctected. Details on interrupt responscs are shown in thc CPU Timing Scction.

Non-Maskable Interrupt **($\overline{NMI}$).** The nonmaskable interrupt cannot be disabled by program control and therefore will bc 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

Table 1. Z80C CPU Registers

Register		Size (Bits)	Rcmarks
A, A′	Accumulator	8	Stores an operand or the results of an opcration.
F, F'	Flags	8	See Instruction Set.
ß, B'	General Purpose	a	Can be uscd separately or as a 16-bit registcr with C.
c. C′	General Purpose	8	Can be uscd separately or as a 1 G-bit register with C.
D, D′	General Purpose	a	Can bc used scparately or as a 1 G-bit rcgistcr with E.
E, E'	General Purpose	a	Can be uscd scparatcly or as a 16-bit rcgistcr with E.
H, H'	General Purpose	8	Can bc uscd separately or as a 16-bit register with L.
L, L'	General Purpose	8	Can be uscd separately or as a 1 G-bit register with L.
			Note: The (B,C), (D,E), and (H,L) sets arc combincd 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 uppcr cight bits of memory addrcss for vcctored interrupl proccssing.
R	Rcfresh Register	8	Provides user-transparent dynamic memory rcfrcsh. Automatically incremented and placed on the addrcss bus during each instruction fetch cyclc.
IX	Index Register	16	Uscd for indexed addressing.
IY	Index Register	16	Uscd for indexed addressing
SP	Stack Pointer	1 6	Holds address of the top of the stack. See Push or Pop in instruction set.
PC	Program Counter	16	Holds addrcss of next instruction.
IFF_1-IFF_2	Interrupt Enable	Flip-Flops	Set or reset to indicatc interrupt status (sec Figure 4).
IMFa-IMFb	Interrupt Mode	Flip-Flops	Reflect Interrupt mode (see Figure 4).

failure has been detected. After recognition of the NMI signal (providing $\overline{BUSREQ}$ is not activc), 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 inpul follows a common timing cycle. After the interrupt has been dctectcd by the CPU (providcd that interrupts are enabled and $\overline{BUSREQ}$ is not active) a spccial interrupt proccssing cycle bcgins. This is a special fetch ($\overline{M1}$) cycle in which $\overline{IORQ}$ becomes active rather than MREQ, as in a normal $\overline{M1}$ cycle. In addition, this spccial $\overline{M1}$ cyclc 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 proccdcrres. The interrupting device places an instruction on the data bus. This is normally a Restart instruction, which will initiatc a call to thc selected one of oight restart locations in page zero of memory. Unlike the 8080, the 280 CPU responds to the Call instruction with only one interrupt acknowledge cyclc followed by two memory read cycles.

Mode 1 Interrupt Operation. Mode 1 opcration is very similar to that for the NMI. The principal difference is that the Mode 1 interrupt has only onc rostart 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 dcvice sclects the starting address of the interrupt service routine. It does this by placing an 8-bit vector on the data bus during the interrupt acknowlcdgc cycle. The CPU forms a pointcr using this byte as the lower 8 bits and the contents of the I registcr as the upper 8 bits. This points to an entry in a table of addrosses 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 *Programmerk 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

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 ior memory ciata bus cxchanges (up to 64K bytes) and for I/O dcvice exchanges.

BUSACK. *Bus Acknowledge* (output, active Low). Bus Acknowledge indicates to the requesting dcvicc that the CPU address bus, data bus, and control signals MREQ, IORQ, RD, and WR have entered their high-impedance states. The extemal circuitry can now control thcsc lines.

BUSREQ. Bus *Request* (input, active Low). Bus Request has a higher priority than NMI and is always recognized at thc end of the current machinc cycle. BUSREQ forces the CPU address bus, data bus, and control signals MREQ, IORQ, RD, and WR to go to a high-impedance state so that other dcviccs can control thcse lines. BUSREQ is normally wired-OR and requires an external pullup for thcsc applications. Extendcd BUSREQ periods due lo extensive DMA operations can prevent the CPU from properly rcfreshing dynamic RAMs.

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

HALT. *Halt State* (output, activc Low). HALT indicates that thc CPU has cxecuted a Halt instruction and is awaiting cithcr a nonmaskablc or a maskable interrupt (with the mask enabled) bcforc operation can resume. While halted, thc CPU cxccutes NOPs to maintain memory refresh.

INT. *Interrupt Request* (input, active Low). Interrupt Requcst is gcncrated by I/O devices. The CPU honors a request at thc end of thc currcnt instruction if the internal software-controlled interrupt enable flip-flop (IFF) is enabled. INT is normally wired-OR and requires an external pullup for thcse applications.

IORQ. *Input/Output Request* (output, active Low, 3-state). IORQ indicatcs that the lower half of the addrcss bus holds a valid I/O address for an I/O read or writc Operation. IORQ is also generated concurrently with M1 during an interrupt acknowledgc cycle to indicatc that an interrupt rcsponsc vector can bc placcd on the data bus.

M1. *Machine Cycle One* (output, active Low). M1, together with MHEQ, indicatcs that the current machino cycle is the opcodc fetch cycle of an instruction cxccution. M1, togcthcr with IORQ, indicatcs an interrupt acknowledgc cycle.

MREQ. *Memory Request* (output, active Low, 3-state). MREQ indicates that the address bus holds a valid address for a memory read or memory writc Operation.

NMI. *Non-Maskable Interrupt* (input, negative edge-triggered). NMI has a higher priority than INT. NMI is always rccognized at the cnd of the currcnt instruction, indcpcndent of thc status of thc interrupt enable flip-flop, and automatically forccs thc CPU to restart at location 0066H.

RD. *Read* (output, activc Low, 3-state). RD indicatcs that the CPU wants to read data from memory or an I/O dcvicc. The addressed I/O dcvicc or memory should use this signal to gate data onto the CPU data bus.

RESET. ßesct (input, activc Low). RESET initializcs thc 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, thc address and data bus go to a high-impcdancc state, and all control output Signals go to thc inactive state. Note that RESET must be active for a minimum of three full clock cycles bcfore thc reset Operation is completc.

RFSH. *Refresh* (output, active Low). RFSH, together with MREQ, indicates that the lower seven bits of the system's address bus can bc used as a rcfrcsh address to the system's dynamic memories.

WAIT. *Wait* (input, activc Low). WAIT indicates to the CPU that the addressed rncmory or I/O dcviccs arc not rcady for a data transfer. The CPU continucs to cntcr a Wait state as long as this signal is active. Extended WAIT periods can prevent the CPU from properly rcfrcshing dynamic memory.

WR. *Write* (output, activc Low, 3-state). WR indicates that thc CPU data bus holds valid data to bc stored at fhe addresscd memory or I/O location.

The Z80 CPU executes instructions by procecding through a specific sequence of operalions:

- Memory read or write
- I/O dcvicc read or writc
- Interrupt acknowledge

The basic clock pcriod is referred to as a T time or cyclc, and three or more T cyclcs make up a machinc cyclc (M 1, M2 or M3 for instancc). Machinc cyclcs can bc extended either by thc CPU automatically inserting onc or more Wait states or by thc inscrtion of one or more Wait states by the user.

Instruction Opcode Fetch. The CPU placcs thc contcnts of the Program Countcr (PC) on thc address bus at the start of thc cyclc (Figure 5). Approximately onc-half clock cyclc later, $\overline{MREQ}$ goes activc. When active, $\overline{RD}$ indicates that thc memory data can be enablcd onto the CPU data bus.

The CPU samplcs thc WAIT input with thc falling edge of clock state T_2. During clock states T_3 and T_4 of an $\overline{M1}$ cyclc, dynamic RAM refresh can occur whilc the CPU starts decoding and executing the instruction. Whcn the Refresh Control signal becomes activc, rcfrcshing of dynamic mctnory can take place.

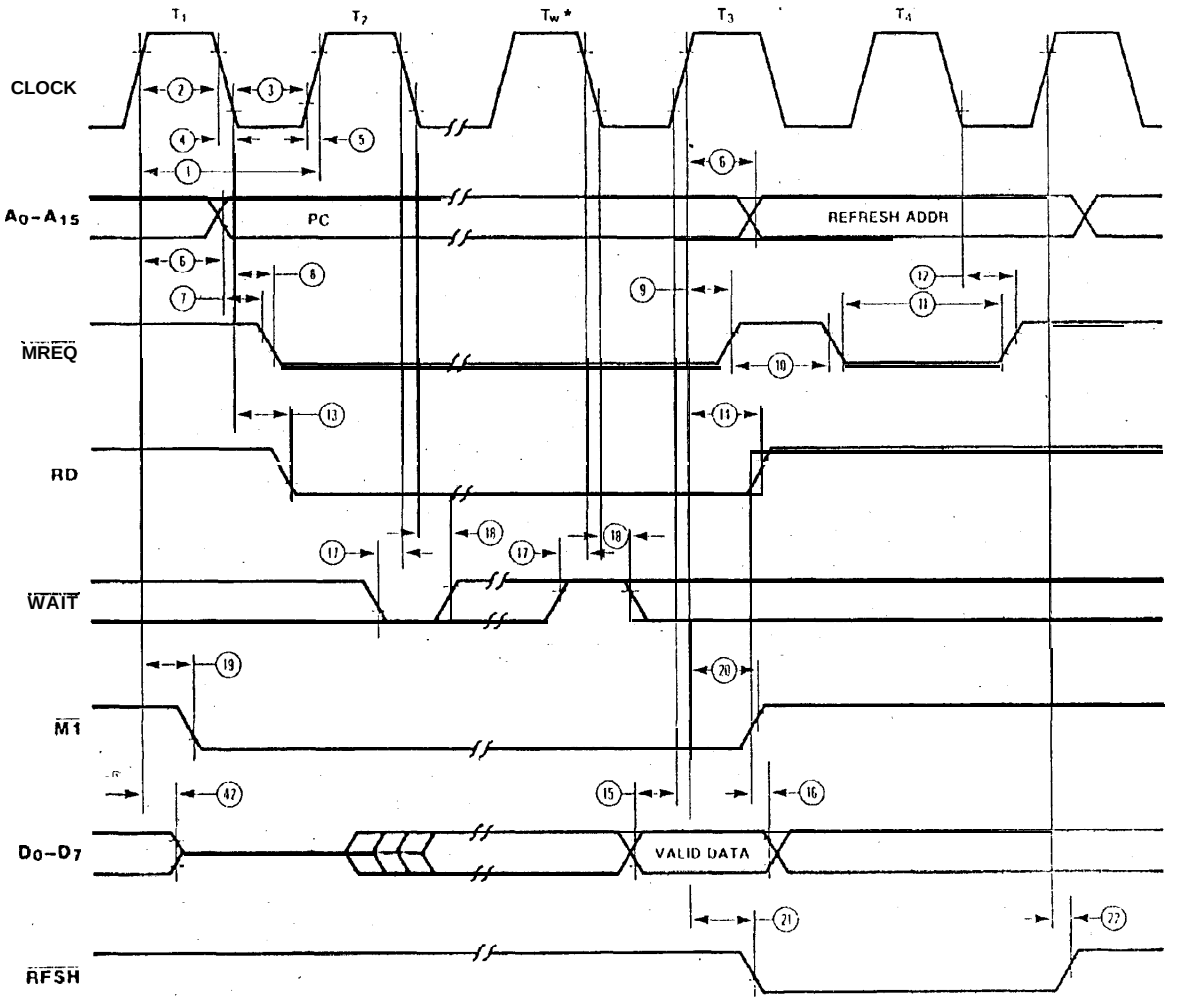


Figure 5. Instruction Opcodc Fetch

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 MREQ and $\overline{RD}$ Signals function exactly as in the fetch cycle. In a memory write cycle, 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.

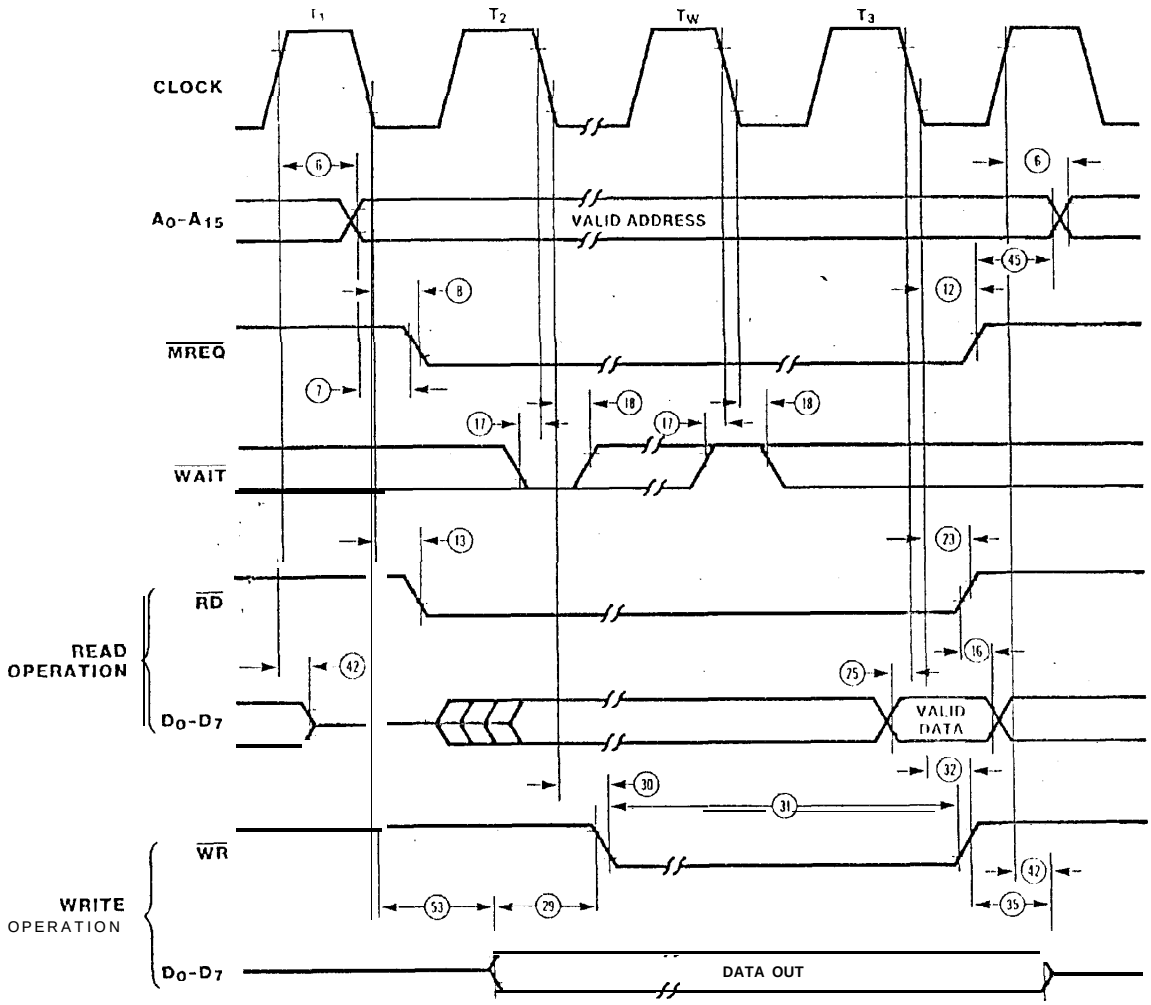
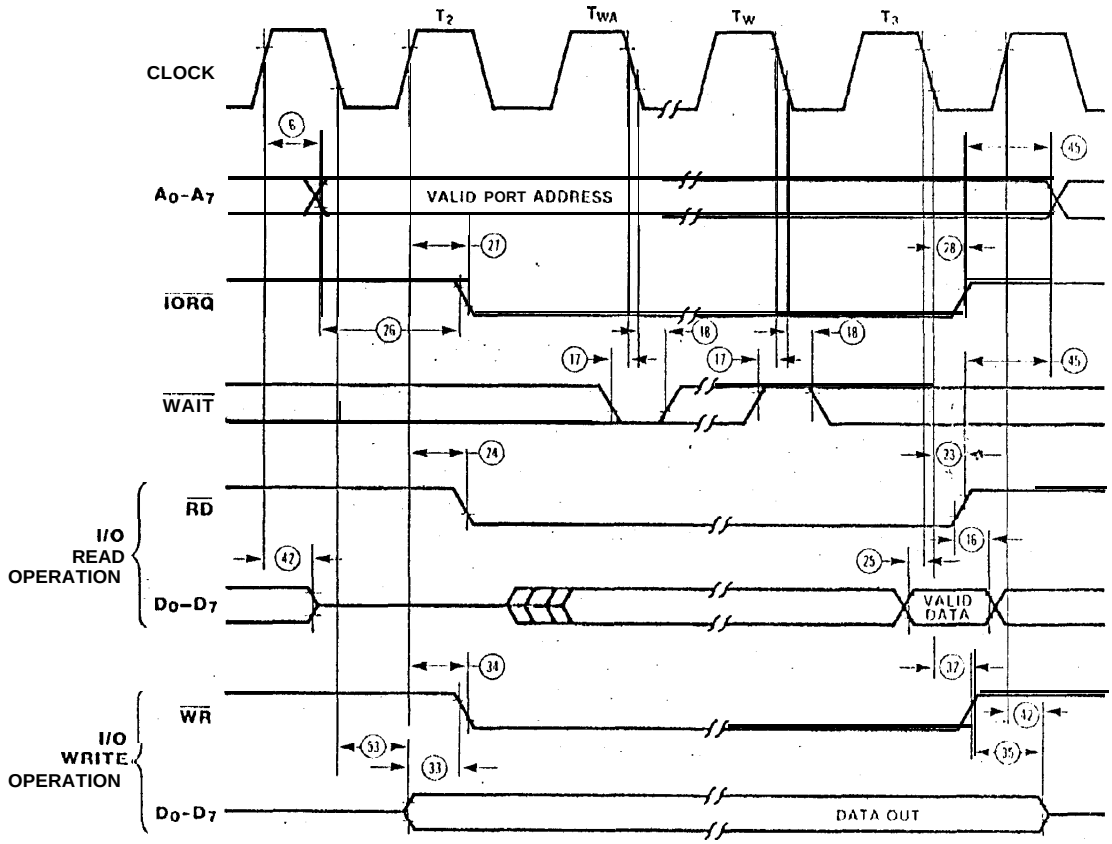


Figure 6. Memory Read or Writc Cycles

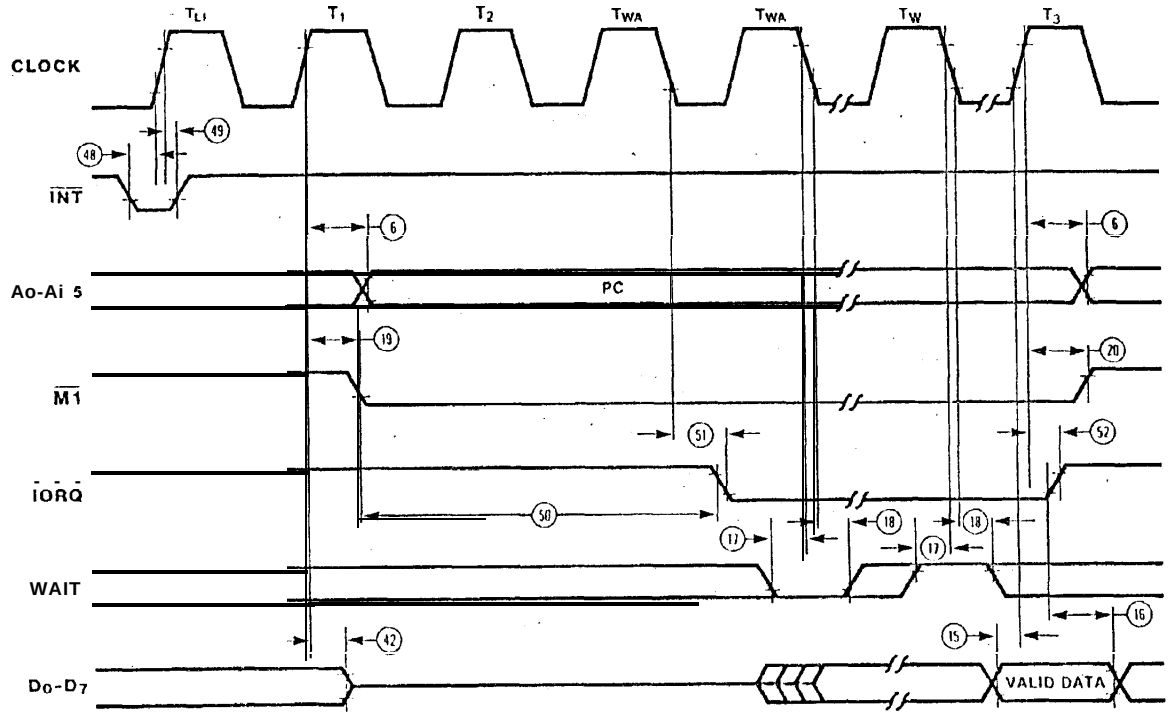
Input or Output Cycles. Figure 7 shows the timing for an I/O read or I/O write Operation. During I/O operations, the CPU automatically inserts a single Wait state (T_{WA}). This extra Wait state allows sufficient time for an I/O port to decode the address from the port address lines.



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

Figure 7. Input or Output Cycles

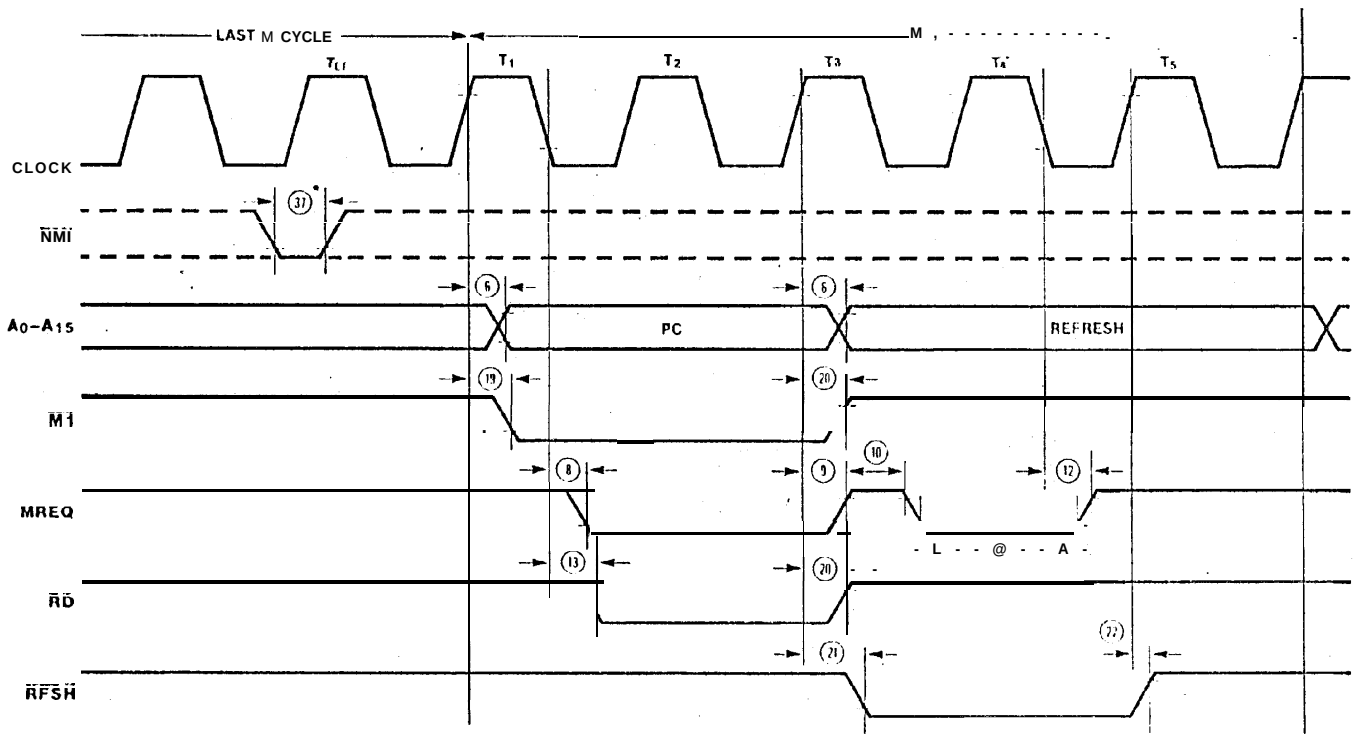
Interrupt Request/Acknowledge Cycle. The CPU samples the interrupt signal with the rising edge of the last clock cycle at the end of any instruction (Figure 8). When an interrupt is accepted, a special $\overline{M1}$ cycle is generated. During this $\overline{M1}$ cycle, IORQ becomes active (instead of 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.



NOTES: 1) T_{LI} = Last state of any instruction cycle.
2) T_{WA} = Wait cycle automatically inserted by CPU.

Figure 8. Interrupt Request/Acknowledge Cycle

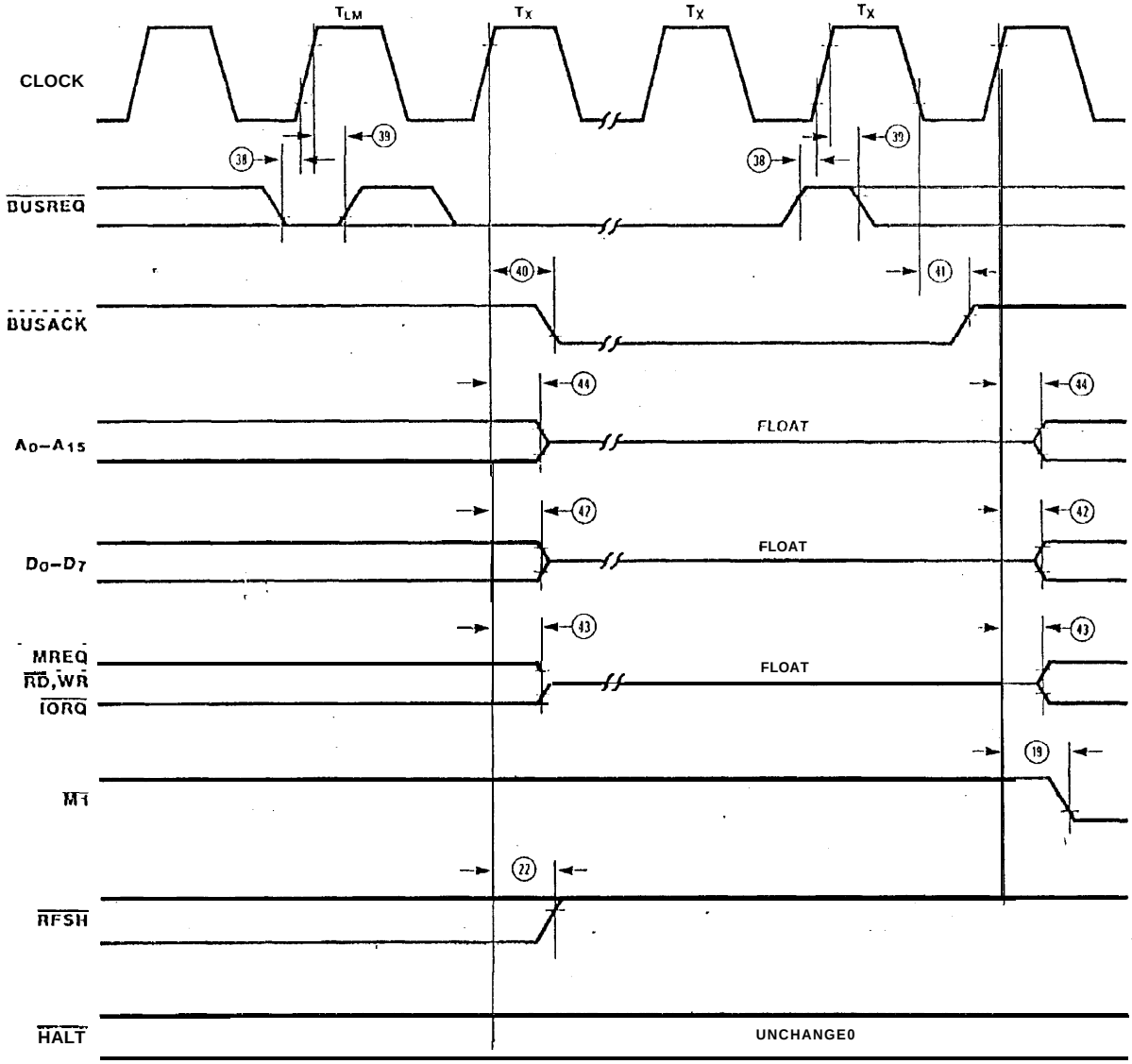
Non-Maskable Interrupt **Request** Cyclc. $\overline{NMI}$ is samplcd at the same time as the maskable inlerrupt input $\overline{INT}$ but has higher priority and cannot be disablcd undcr softwar-c control. The subsequent timing is similar to that of a normal memory read Operation cxccpt that data put on the bus by the memory is ignored. The CPU instead cxecutcs a rcstart (RST) Operation and jumps to the NMI service routine locatcd at address 0066H (Figure 9).



*Although $\overline{NMI}$ is an asynchronous input, lo guarantee ils being rccognizcd onthe lollowing machine cyclc, NMI's falling cdgc must occur no later than the rising edge of the clock cycle preceding the last state of any instruction cyclc (T_{LI}).

Figure 9. Non-Maskable Interrupt Rcqucst Operation

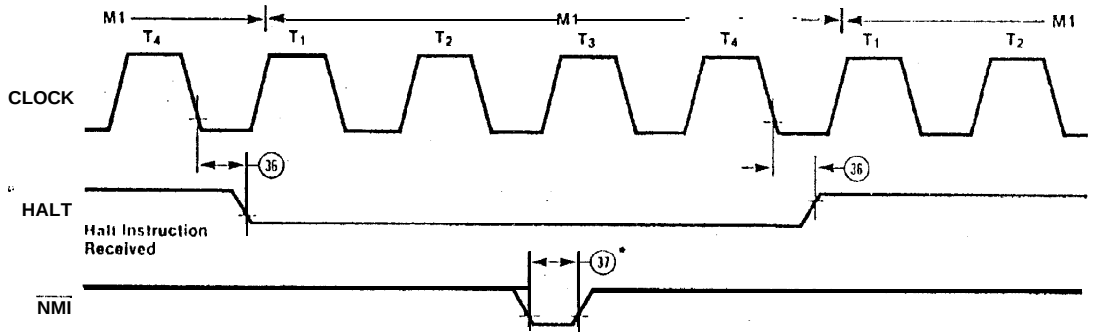
BUS Request/Acknowledge Cycle. The CPU samples $\overline{BUSREQ}$ with the rising edge of the last clock period of any machine cycle (Figure 10). If $\overline{BUSREQ}$ is active, the CPU sets its address, data, and $\overline{MREQ}$, $\overline{IORQ}$, RD, and $\overline{WR}$ lines to a high-impedance 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

Halt Acknowledge Cycle. When the CPU receives a HALT instruction, it executes NOP states until eilher an $\overline{INT}$ **or** $\overline{NMI}$ input is receivcd. When in the Halt state, the $\overline{HALT}$ output is active and rcmains so until an interrupt is rcceived (Figure 11). $\overline{INT}$ will also forcc a Hall exit.



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

Figure 11. Halt Acknowledge

Reset Cycle. $\overline{RESET}$ must be active for at **least** three clock cycles for the CPU to propcrly accept it. As long as $\overline{RESET}$ rcmains activc, the addrcss and data buses float, and thc control Outputs are inactivc. Once $\overline{RESET}$ goes inactivc, two internal T cycles are consumed before the CPU rcsumes normal processing Operation. $\overline{RESET}$ clears the PC register, so thc first opcode fetch will bc to location 0000H (Figure 12).

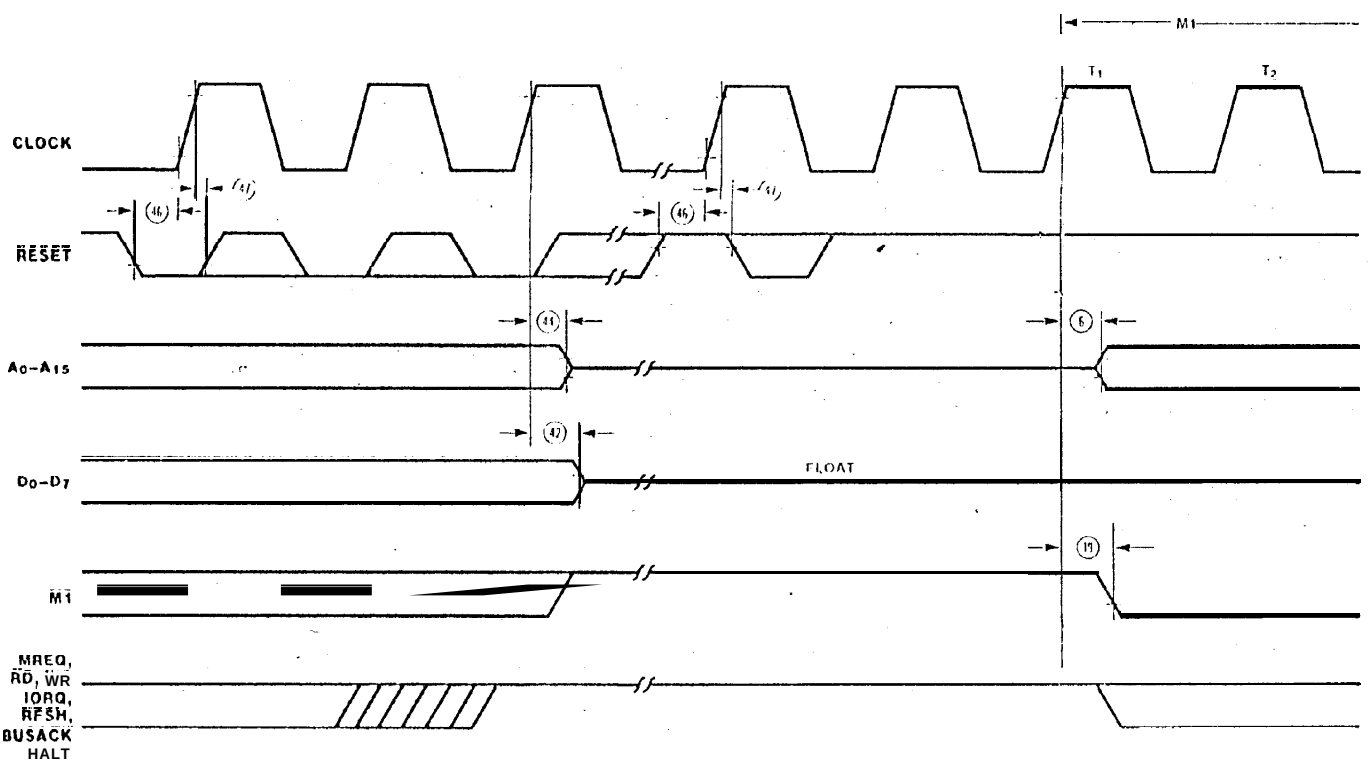


Figure 12. Reset Cycle

Power-Down modo of **Operation (Only applies to** CMOS Z80 CPU).

CMOS Z80 CPU supports Power-Down mode of Operation. This mode is also referrod to as the "siandby **mode",** and supply **current for** thc CPU goes down as low **as 10 uA** (Whcre specified **as** Icc_2).

Power-Down Acknowledge Cycle. When thc clock inpul to the CPU is stopped at eilher a High or Low level, the CPU stops its operation and maintains all registers and control signals. However, I_{cc2} (standby supply currcnt) is guaranteed only when the system clock is stopped at a Low level during T_4 of the machine cyclc following the execution of the H A L T instruction. The timing diagram for the power-down function, when implemented with the HALT instruction, is shown in Figure 13.

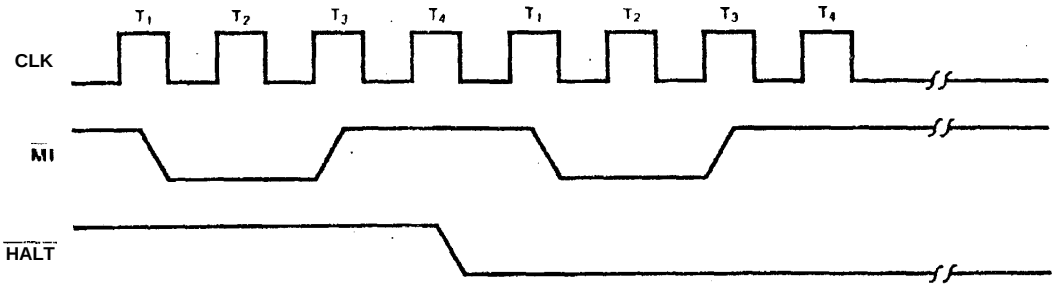


Figure 13. Power-Down Acknowledgc

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.

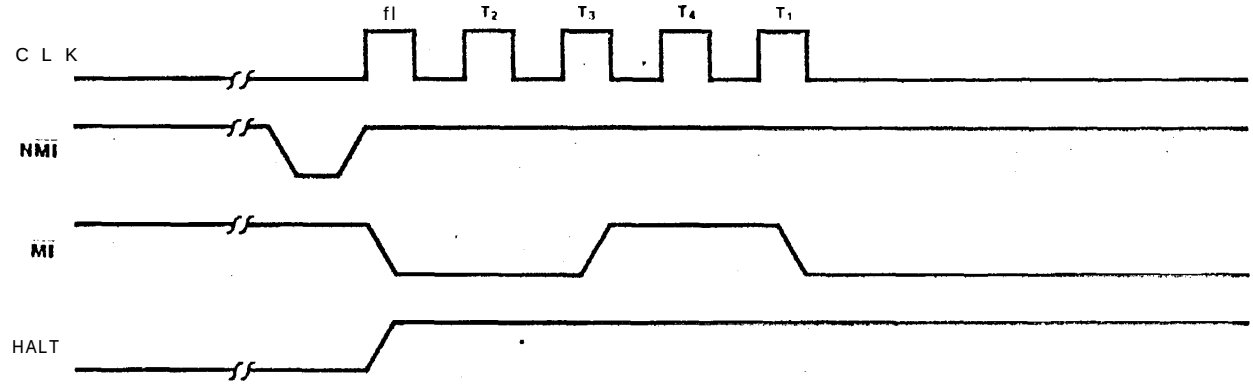


Figure 14a.

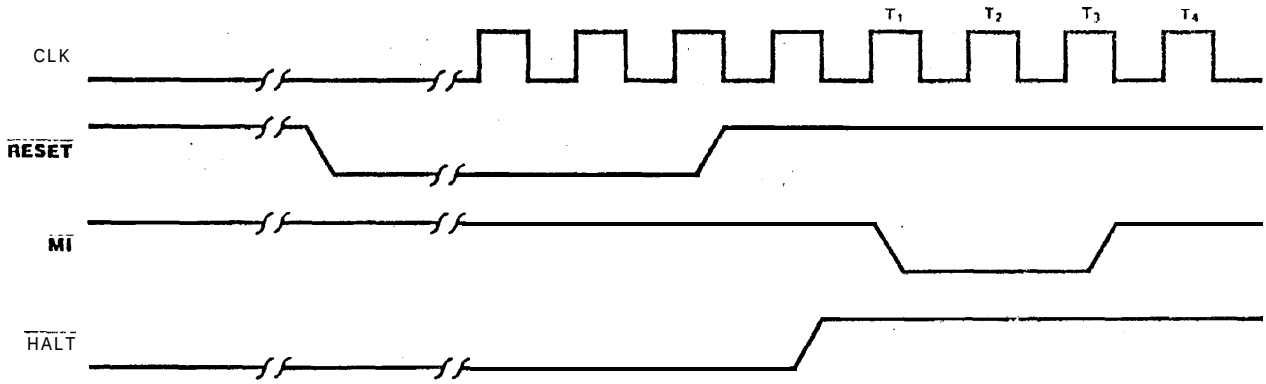


Figure 14b.

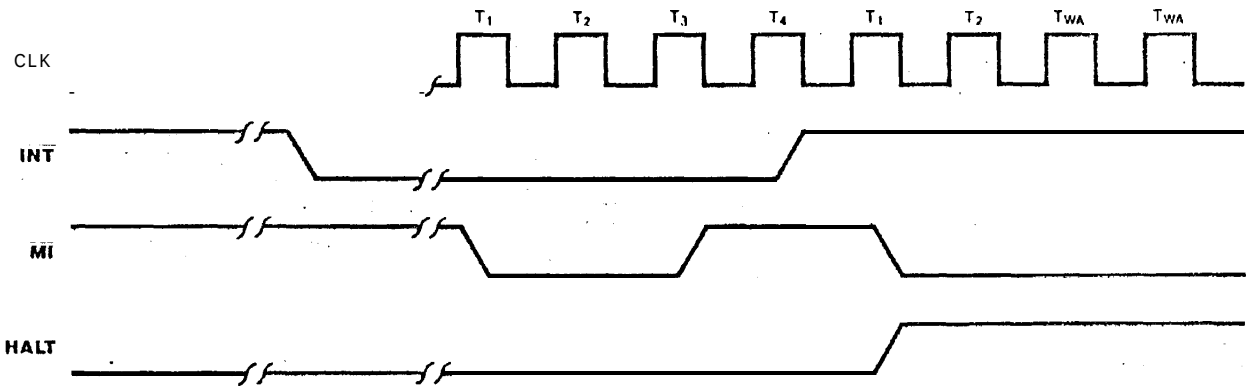


Figure 14c.

Figure 13. Power-Down Release

ABSOLUTE MAXIMUM RATINGS

Voltage on V_{CC} with rcspect 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
Tcmperature . . . . . .See Ordering Information
Storage Temperature . . . . . – 65 °C to + 150°C

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

STANDARD TEST CONDITIONS

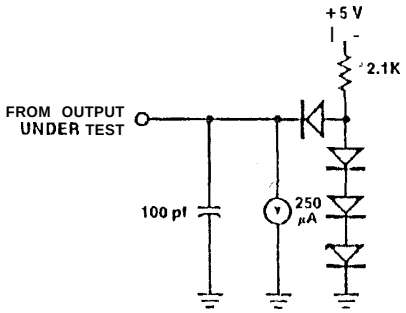
The DC Charactcristics and capacitance scctions below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND (OV). Positive current flows into the referenccd pin.

Available operating tcmperature ranges are:

- s = 0°C to +70°C
  Voltage Supply Range:
  NMOS: $+4.75V \geq V_{cc} \geq +5.25V$
  CMOS: $+4.50V \geq V_{cc} \geq 5.50V$
- E = -40°C to 1 00°C, $+4.50V \geq V_{cc} > +5.50V$

All ac parameters assume a load capacitance of 100 pf. Add 10 ns dclay for each 50 pf increasc in load up to a maximum of 200 pf for thc data bus and 100 pf for address and control lincs. AC timing measurcmcnts are refcrenccd to 1.5 volts (oxcept for clock, which is referenced to thc 10% and 90% points).

The Ordcring Information section lists tcmperature ranges and product numbcrs. Package drawings arc in thc Package Information section. Refcr to thc 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 Vollage	V_{CC} -.6	V_{CC} +.3	V	
V_{IL}	Input Low Vollage	-0.3	0.8	V	
V_{IH}	Input High Vollage	2.2	V_{CC}	V	
VOL	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 Vollage	V_{CC} - 0.8		V	I_{OH} = -250 µA
I_{CC_1}	Power Supply Currenl 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	
I_{CC_2}	Standby Supply Currenl		10	µA	V_{CC} = 5V CLK = (0) V_{IH} = V_{CC} - 0.2V V_{IL} = 0.2V
I_{LI}	Input Leakagc Current		10	µA	V_{IN} = 0.4 10 V_{CC}
I_{LO}	3-State Output Leakage Current in Floal	-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 guaranleed only when the supplied clock is stopped at a low level during T_4 of thc machine cycle immediately following the cxeculion of a HALT instruction.

CAPACITANCE

Symbol	Parameter	Min	Max	Unit
CCLOCK	Clock Capacitance		10	pf
C_{IN}	Input Capacitance		5	pf
COUT	Output Capacitance		15	Pf

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

AC CHARACTERISTICS† (Z84C00/CMOS Z80 CPU)

Number	Symbol	Parameter	Z84C0004 Min	Z84C0004 Max	Z84C0006 Min	Z84C0006 Max	Z84C0008 Min	Z84C0008 Max	Z84C0010 Min	Z84C0010 Max
1	TcC	Clock Cycle Time	250*	DC	162*	DC	125	DC	100	DC
2	TwCh	Clock Pulse Width (High)	110	DC	65	DC	55	DC	42	DC
3	TwCl	Clock Pulse Width (Low)	110	DC	65	DC	55	DC	42	DC
4	TfC	Clock Fall Time		30		20		10		10
5	TrC	Clock Rise Time		30		20		10		10
6	TdCr(A)	Clock ↑ to Address Valid Delay	110		90		80		65	
7	TdA(MREQf)	Address Valid to $\overline{MREQ}$ ↓ Delay	65*		35*		20*		22'	
8	TdCf(MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay	85		70		60		55	
9	TdCr(MREQr)	Clock ↑ to $\overline{MREQ}$ ↑ Delay	85		70		60		55	
10	TwMREQh	$\overline{MREQ}$ Pulse Width, (High)	110*††		65*††		45*††		32*††	
11	TwMREQl	$\overline{MREQ}$ Pulse Width (Low)	220*††		135*††		100*††		75'††	
12	TdCf(MREQr)	Clock ↓ to $\overline{MREQ}$ ↑ Delay	85		70		60		55	
13	TdCf(RDf)	Clock ↓ to $\overline{RD}$ ↓ Delay	95		80		70		65	
14	TdCr(RDr)	Clock ↑ to $\overline{RD}$ ↑ Delay	85		70		60		55	
15	TsD(Cr)	Data Setup Time to Clock ↑	35		30		30		25	
16	ThD(RDr)	Data Hold Time to $\overline{RD}$ ↑	0		0		0		0	
17	TsWAIT(Cf)	$\overline{WAIT}$ Setup Time to Clock ↓	70		60		50		25	
18	ThWAIT (Cf)	$\overline{WAIT}$ Hold Time after Clock ↓	10		10		10		10	
19	TdCr(M1f)	Clock ↑ to $\overline{M1}$ ↓ Delay		100		80		70		65
20	TdCr(M1r)	Clock ↑ to $\overline{M1}$ ↑ Delay		100		80		70		65
21	TdCr(RRSHf)	Clock ↑ to RFSH ↓ Delay		130		110		95		80
22	TdCr(RFSHr)	Clock ↑ to $\overline{RFSH}$ ↑ Delay		120		100		85		80
23	TdCf(RDr)	Clock ↓ to $\overline{RD}$ ↑ Delay		85		70		60		55
24	TdCr(RDf)	Clock ↑ to $\overline{RD}$ ↓ Delay		85		70		60		55
25	TsD(Cf)	Data Setup to Clock ↓ during M_2, M_3, M_4, or M_5 Cycles	50		40		30		25'	
26	TdA(IORQf)	Address Stable prior to IORQ ↓	180*	110*		75*		70'		
27	TdCr(IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Delay		75		65		55		50
28	TdCf(IORQr)	Clock ↓ to IORQ ↑ Delay		85		70		60		55
29	TdD(WRf)Mw	Data Stable prior to $\overline{WR}$ ↓	80*		25*		5'		40"	

*For clock periods other than the minimums shown, calculate Parameters *usrng* 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.

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

Number	Symbol	General Parameter	Z84C0004 Min	Z84C0004 Max	Z84C0006 Min	Z84C0006 Max	Z84C0008 Min	Z84C0008 Max	Z84C0010 Min	Z84C0010 Max
30	TdCf(WRf)	Clock ↓ to $\overline{WR}$ ↓ Delay		80		70		60		55
31	TwWR	$\overline{WR}$ Pulse Width	220*		135		100*		75*	
32	TdCf(WRr)	Clock ↓ to $\overline{WR}$ ↑ Delay		80		70		60		55
33	TdD(WRf)IO	Data Stable prior to $\overline{WR}$ ↓	-10*		-55*		-55*		-8*	
34	TdCr(WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay		65		60		55		50
35	TdWRr(D)	Data Stable from $\overline{WR}$ ↑	60*		30*		15*		12*	
36	TdCf(HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓		300		260		225		90
37	TwNMI	$\overline{NMI}$ Pulse Width	80		70		60		60	
38	TsßUSREQ(Cr)	$\overline{BUSREQ}$ Setup Time to Clock ↑	50		50		40		30	
39	ThBUSREQ(Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	10		10		10		10	
40	TdCr(BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay		100		90		80		75
41	TdCf(ßUSACKr)	Clock ↓ to $\overline{BUSACK}$ ↑ Delay		100		90		80		75
42	TdCr(Dz)	Clock ↑ to Data Float Delay		90		80		70		65
43	TdCr(CTz)	Clock ↑ to Control Outputs Float Delay ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)		80		70		60		60
44	TdCr(Az)	Clock ↑ to Address Float Delay		90		80		70		65
45	TdCTr(A)	$\overline{MREQ}$ ↑, $\overline{IORQ}$ ↑, $\overline{RD}$ ↑, and $\overline{WR}$ ↑ to Address Hold Time	80*		35*		20*		32*	
46	TsRESET(Cr)	$\overline{RESET}$ to Clock ↑ Setup Time	60		60		45		40	
47	ThRESET(Cr)	$\overline{RESET}$ to Clock ↑ Hold Time	10		10		10		10	
48	TsINTf(Cr)	$\overline{INT}$ to Clock ↑ Setup Time	80		70		55		50	
49	ThINTr(Cr)	$\overline{INT}$ to Clock ↑ Hold Time	10		10		10		10	
50	TdM1f(IORQf)	$\overline{M1}$ ↓ to $\overline{IORQ}$ ↓ Delay	565*		365*		270*		222*	
51	TdCf(IORQf)	Clock ↓ to $\overline{IORQ}$ ↓ Delay		85		70		60		55
52	TdCf(IORQr)	Clock ↑ to $\overline{IORQ}$ ↑ Delay		85		70		60		55
53	TdCf(D)	Clock ↓ to Data Valid Delay		150		130		115		110

● For clock periods other than the minimums shown, calculate parameters using the folfowing table. Calculated values above assumed TrC = TfC = 20 ns.

†Units in nanoseconds (ns).

FOOTNOTES TO AC CHARACTERISTICS

Number	Symbol	General Parameter	Z84C0004	Z84C0006	Z84C0008	Z84C0010
1	TcC	TwCh + TwCl + TrC + TfC				
7	TdA(MREQf)	TwCh + TfC	-65	-50	-45	-45
10	TwMREQh	TwCh + TfC	-20	-20	-20	-20
11	TwMREQl	TcC	-30	-30	-25	-25
26	TdA(IORQf)	TcC	-70	-55	-50	-50
29	TdD(WRf)	TcC	-170	-140	-120	-60
31	TwWR	TcC	-30	-30	-25	-25
33	TdD(WRf)	TwCl + TrC	-140	-140	-120	-60
35	TdWRr(D)	TwCl + TrC	-70	-55	-50	-40
45	TdCTr(A)	TwCl + TrC	-50	-50	-45	-30
50	TdM1f(IORQf)	2TcC + TwCh + TfC	-65	-50	-45	-30

AC Test Condilions: 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

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'	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.0 mA
V_{OH}	Output High Voltage	2.4'		V	I_{OH} = -250 μA
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)

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 Widlh (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 Addrcss Valid Delay		110		90		80
7	TdA(MREQf)	Address Valid to $\overline{MREQ}$ ↓ Dclay	65'		35*		20'	
8	TdCf(MREQf)	Clock ↓ to $\overline{MREQ}$ ↓ Delay		85		70		60
9	TdCr(MREQr)	Ciock ↑ to $\overline{MREQ}$ ↑ Dclay		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 $\overline{MREQ}$ ↑ Dclay		85		70		60
13	TdCf(RDf)	Clock ↓ to $\overline{RD}$ ↓ Dclay		35		80		70
14	TdCr(RDr)	Clock ↑ to $\overline{RD}$ ↑ Dclay		85		70		60
15	TsD(Cr)	Data Sctup Time to Clock ↑	35		30		30	
16	ThD(RDr)	Data Hold Time to $\overline{RD}$ ↑		0		0		0
17	TsWAIT(Cf)	$\overline{WAIT}$ Sctup 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		GO
24	TdCr(RDf)	Clock ↑ to $\overline{RD}$ ↓ Dclay		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)	Addrcss Stablc prior to $\overline{IORQ}$ ↓	180*		110"		75'	
27	TdCr(IORQf)	Clock ↑ to $\overline{IORQ}$ ↓ Dclay		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 $\overline{WR}$ ↓ Dclay		80		70		60
31	TwWR	$\overline{WR}$ Pulse Width	220'		135*		100'	
32	TdCf(WRr)	Clock ↓ to $\overline{WR}$ ↑ Dclay		80		70		60
33	TdD(WRf)	Data Stable prior to $\overline{WR}$ ↓	-10'		-55*		55"	
34	TdCr(WRf)	Clock ↑ to $\overline{WR}$ ↓ Delay		65		GO		55
35	TdWRr(D)	Data Stablc from $\overline{WR}$ ↑	60*		30'		15"	
36	TdCf(HALT)	Clock ↓ to $\overline{HALT}$ ↑ or ↓		300		260		225
37	TwNMI	NMI Pulse Width.	80		70		60'	
38	TsBUSREQ(Cr)	$\overline{BUSREQ}$ Sctup 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.

AC CHARACTERISTICS† (Z8400/NMOS Z80 CPU; Continucd)

Number	Symbol	Parameter	20840004 Min	20840004 Max	Z0840006 Min	Z0840006 Mux	20840008 Min	20840008 Max
3	9 ThBUSREQ(Cr)	$\overline{BUSREQ}$ Hold Time after Clock ↑	0		0		0	
4	0 TdCr(BUSACKf)	Clock ↑ to $\overline{BUSACK}$ ↓ Delay		100		90		80
4	1 TdCf(BUSACKr)	Clock ↓ to $\overline{BUSACK}$ ↑ Dclay		100		90		80
42	TdCr(Dz)	Clock ↑ to Data Float Dclay		90		80		70
43	TdCr(CTz)	Clock ↑ to Control Outputs Float Dclay ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$)		80		70		60
4	4 TdCr(Az)	Clock ↑ to Address Float Dclay		90		80		70
45	TdCTr(A)	$\overline{MREQ}$ ↑, $\overline{IORQ}$ ↑, $\overline{RD}$ ↑, and $\overline{WR}$ ↑ to Address Hold Time	80'		35*		20*	
46	TsRESET(Cr)	$\overline{RESET}$ to Clock ↑ Setup Time	60		60		45	
47	ThRESET(Cr)	$\overline{RESET}$ to Clock ↑ Hold Time		0		0		0
48	TsINTf(Cr)	$\overline{INT}$ to Clock ↑ Sctup Time	80		70		55	
49	ThINTr(Cr)	$\overline{INT}$ to Clock ↑ Hold Time		0		0		0
50	TdM1f(IORQf)	$\overline{M1}$ ↓ to $\overline{IORQ}$ ↓ Delay	565*		365'		270'	
51	TdCf(IORQf)	Clock ↓ to $\overline{IORQ}$ ↓ Dclay		85		70		60
52	TdCf(IORQr)	Clock ↑ $\overline{IORQ}$ ↑ Dclay		85		70		60
53	TdCf(D)	Clock 3 to Data Valid Delay		150		130		115

*For clock periods other than the minimums shown, calculate parameters using the following table. Calculated values above assumed TrC = TfC = 20 ns.
†Units in nanoseconds (ns).

FOOTNOYES YO AC CHARACTERISTICS

Number	Symbol	General Parameter	20840004	Z0840006	Z0840008
1	TcC	TwCh + TwCl + TrC + TfC			
7	TdA(MREQf)	TwCh + TfC	– 65	-50	-45
10	TwMREQh	TwCh + TfC	– 20	-20	-20
11	TwMREQl	TcC	– 30	-30	-25
26	TdA(IORQf)	TcC	– 70	-55	-50
29	TdD(WRf)	TcC	– 170	-140	-120
31	TwWR	TcC	– 30	-30	-25
33	TdD(WRf)	TwCl + TrC	– 140	-140	-120
35	TdWRr(D)	TwCl + TrC	– 70	-55	-50
45	TdCTr(A)	TwCl + TrC	– 50	-50	-45'
50	TdM1f(IORQf)	2TcC + TwCh + TfC	– 65	-50	-45

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