

**Z8400/Z84C00 NMOS/CMOS
Z80® CPU
Central Processing Uni**

January 1989

FEATURES

- The extensive instruction 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 microprocessors and associated family of peripherals can be linked by a vectored interrupt system. This system can be daisy-chained to allow implementation of a priority interrupt scheme.
- Duplicate set of both general-purpose and flag registers.
- Two sixteen bit index registers.
- Three modes of maskable interrupts:
Mode 0—8080A similar;
Mode 1 -Non-Z80 environment, location 38H;
Mode 2—Z80 family peripherals, vectored interrupts.
- On-chip dynamic memory refresh counter.

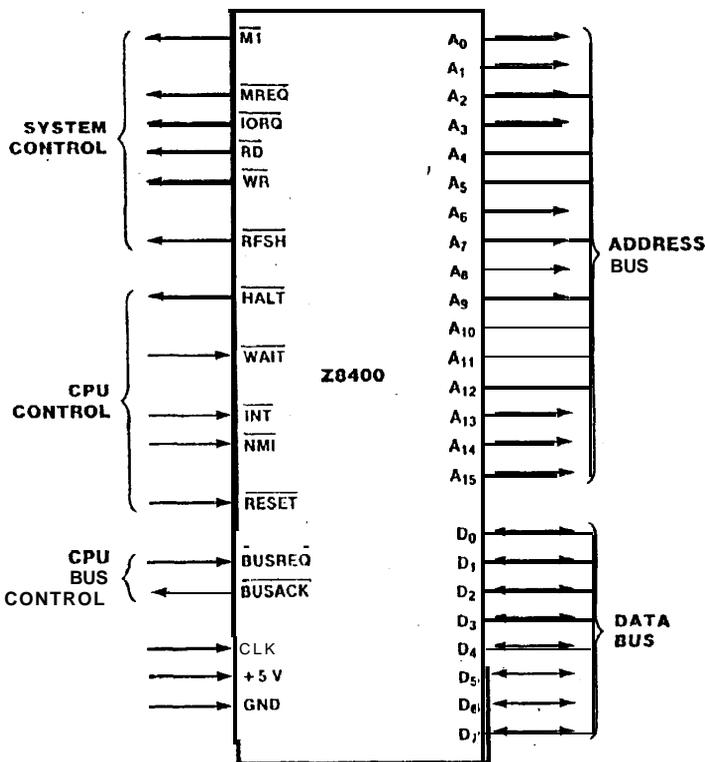


Figure 1. Pin Functions

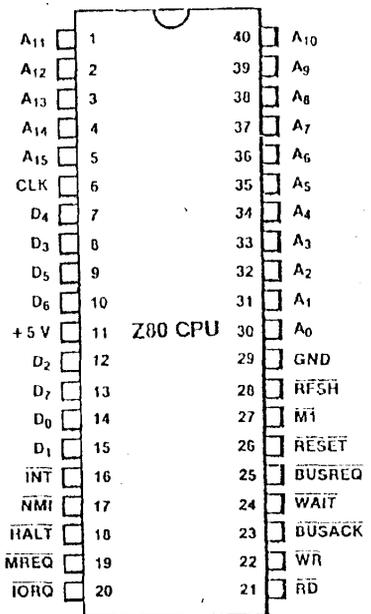
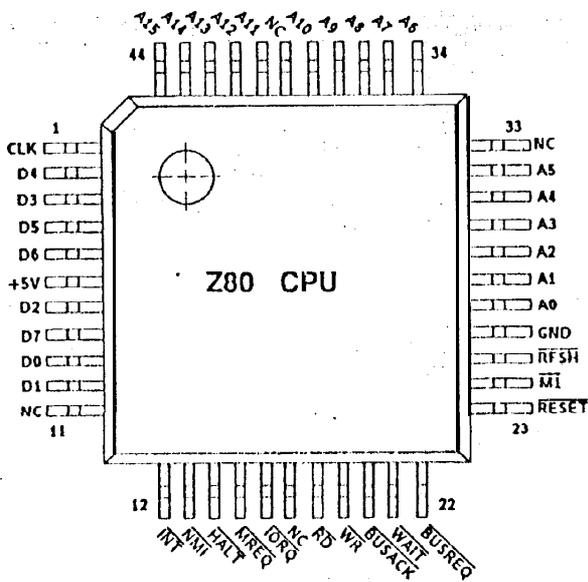


Figure 2. 40-pin Dual-In-Line (DIP), Pin Assignments



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

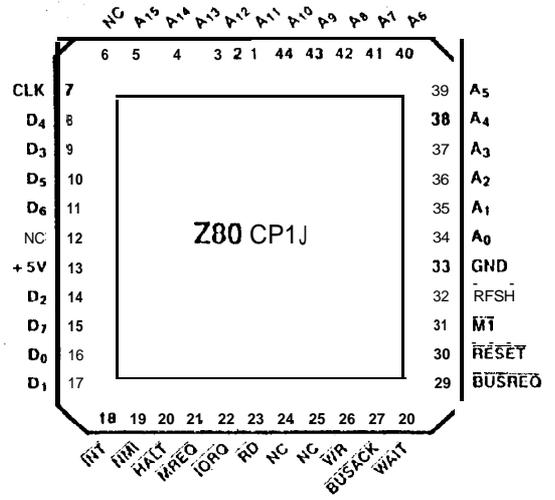


Figure 2b. 44-Pin Chip Carrier Pin Assignments

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.

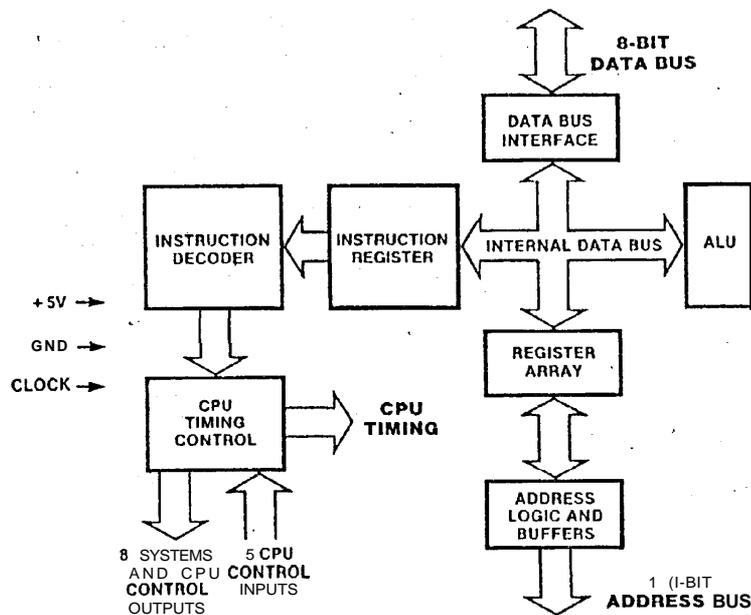


Figure 3. Z80C CPU Block Diagram

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.

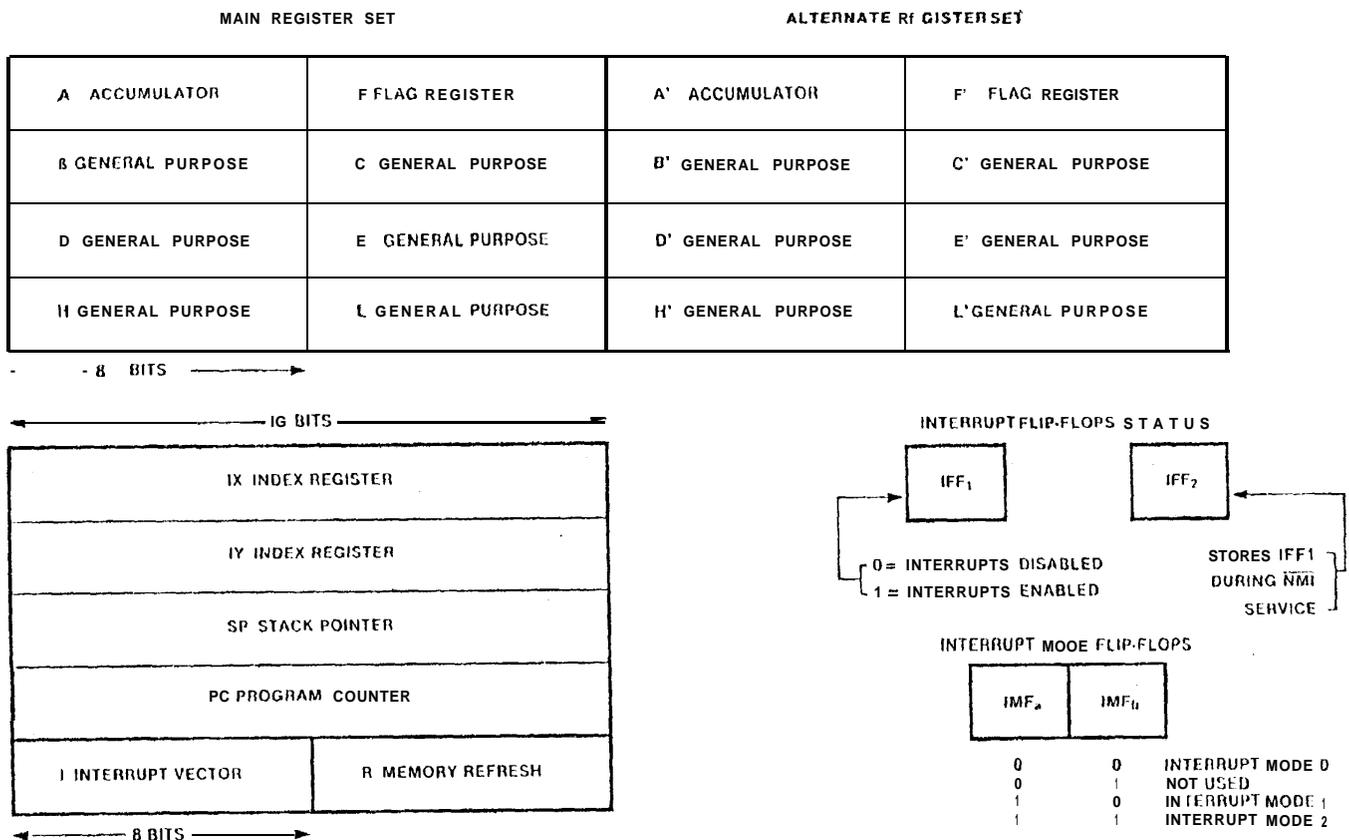


Figure 4. CPU Registers

INTERRUPTS: GENERAL OPERATION

The CPU accepts two interrupt input signals: $\overline{\text{NMI}}$ and $\overline{\text{INT}}$. The $\overline{\text{NMI}}$ is a non-maskable interrupt and has the highest priority. $\overline{\text{INT}}$ is a lower priority interrupt and it requires that interrupts be enabled in software in order to operate. $\overline{\text{INT}}$ 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, $\overline{\text{INT}}$, 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 280 Family and compatible peripheral devices.

The CPU services interrupts by sampling the $\overline{\text{NMI}}$ and $\overline{\text{INT}}$ 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 ($\overline{\text{NMI}}$). The nonmaskable interrupt cannot be disabled by program control and therefore will be accepted at all times by the CPU. $\overline{\text{NMI}}$ 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)	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 B.
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 D.
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 H.
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 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 procedure. 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 280 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 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, IFF₁ and IFF₂, 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 → Parity flag
LD A,R instruction execution	•	•	IFF ₂ → Parity flag
Accept \overline{NMI}	0	•	Maskable interrupt \overline{INT} disabled
RETN instruction execution	IFF ₂	•	IFF ₂ → 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.

- Bit set, reset, and test operations
- Jumps
- Calls, returns, and restarts
- Input and output operations

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 Reference Guide (03-0012-03)*, and *Assembly Language Programming Manual (03-0002-01)* contain significantly more details for programming use.

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:

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

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

PIN DESCRIPTIONS

A₀-A₁₅. *Address Bus* (output, active High, 3-state). A₀-A₁₅ 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.

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

BUSREQ. *Bus Request* (input, active Low). Bus Request has a higher priority than NMI and is always recognized at the end of the current machine 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 devices can control these lines. BUSREQ is normally wired-OR and requires an external pullup for these applications. Extended BUSREQ periods due to extensive DMA operations can prevent the CPU from properly refreshing dynamic RAMs.

D₀-D₇. *Data Bus* (input/output, active High, 3-state). D₀-D₇ constitute an 8-bit bidirectional data bus, used for data exchanges with memory and I/O.

HALT. *Halt State* (output, active Low). 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.

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. INT is normally wired-OR and requires an external pullup for these applications.

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

M1. *Machine Cycle One* (output, active Low). M1, together with MREQ, indicates that the current machine cycle is the opcode fetch cycle of an instruction execution. M1, together with IORQ, indicates an interrupt acknowledge 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 write operation.

NMI. *Non-Maskable Interrupt* (input, negative edge-triggered). NMI has a higher priority than INT. 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.

RD. *Read* (output, active Low, 3-state). 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.

RESET. *Reset* (input, active Low). 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 RESET must be active for a minimum of three full clock cycles before the reset operation is complete.

RFSH. *Refresh* (output, active Low). RFSH, together with 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.

WAIT. *Wait* (input, active Low). 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 WAIT periods can prevent the CPU from properly refreshing dynamic memory.

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

CPU TIMING

The Z80 CPU executes instructions by proceeding through a specific sequence of operations:

- Memory read or write
- I/O device read or write
- Interrupt acknowledge

The basic clock period is referred to as a T time or cycle, and three or more T cycles make up a machine cycle (M1, M2 or M3 for instance). Machine cycles can be extended either by the CPU automatically inserting one or more Wait states or by the insertion of one or more Wait states by the user.

Instruction **Opcode Fetch**. The CPU places the contents of the Program Counter (PC) on the address bus at the start of the cycle (Figure 5). Approximately one-half clock cycle later, \overline{MREQ} goes active. When active, \overline{RD} indicates that the memory data can be enabled onto the CPU data bus.

The CPU samples the WAIT input with the falling edge of clock state T_2 . During clock states T_3 and T_4 of an $\overline{M1}$ cycle, dynamic RAM refresh can occur while the CPU starts decoding and executing the instruction. When the Refresh Control signal becomes active, refreshing of dynamic memory can take place.

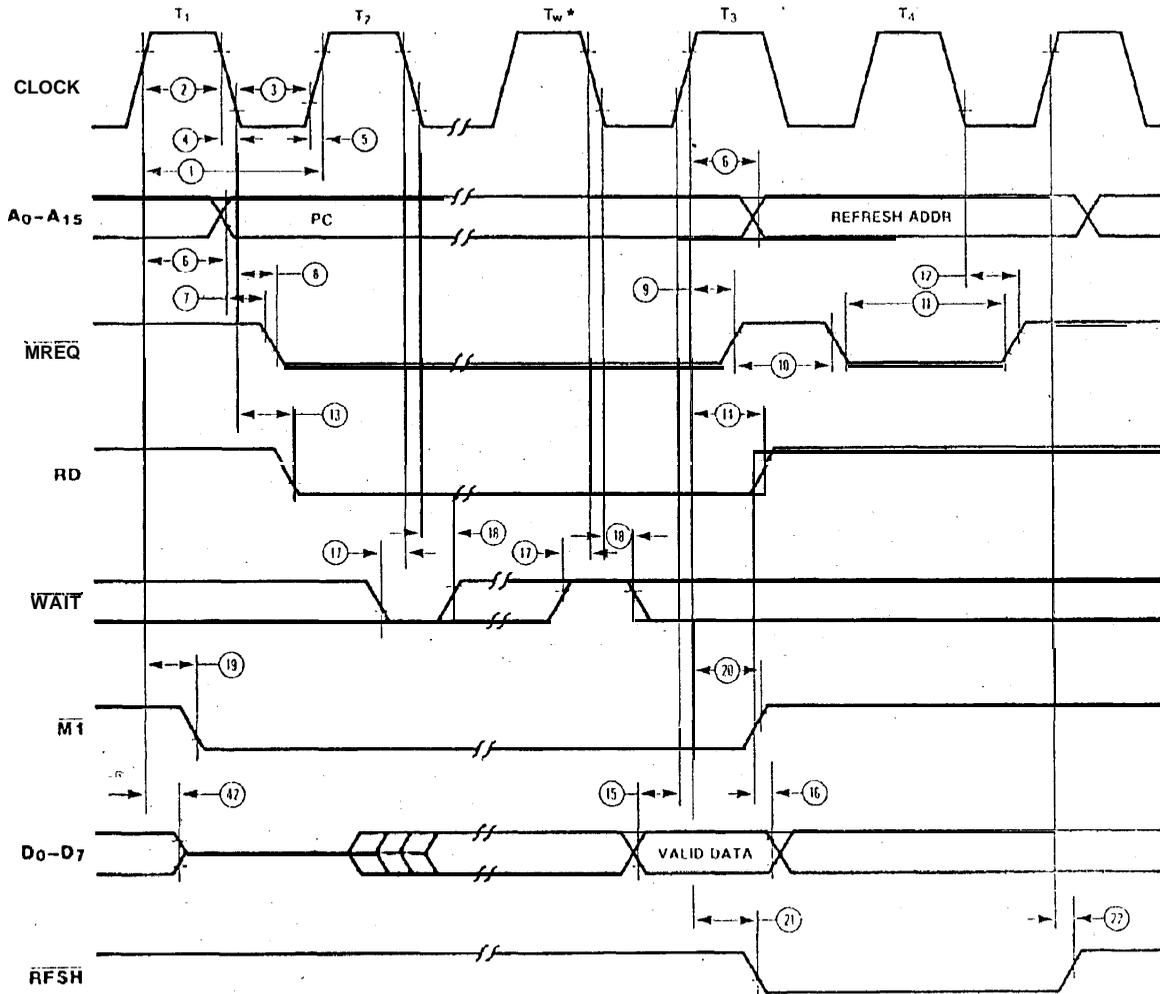


Figure 5. Instruction Opcode Fetch

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 \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.

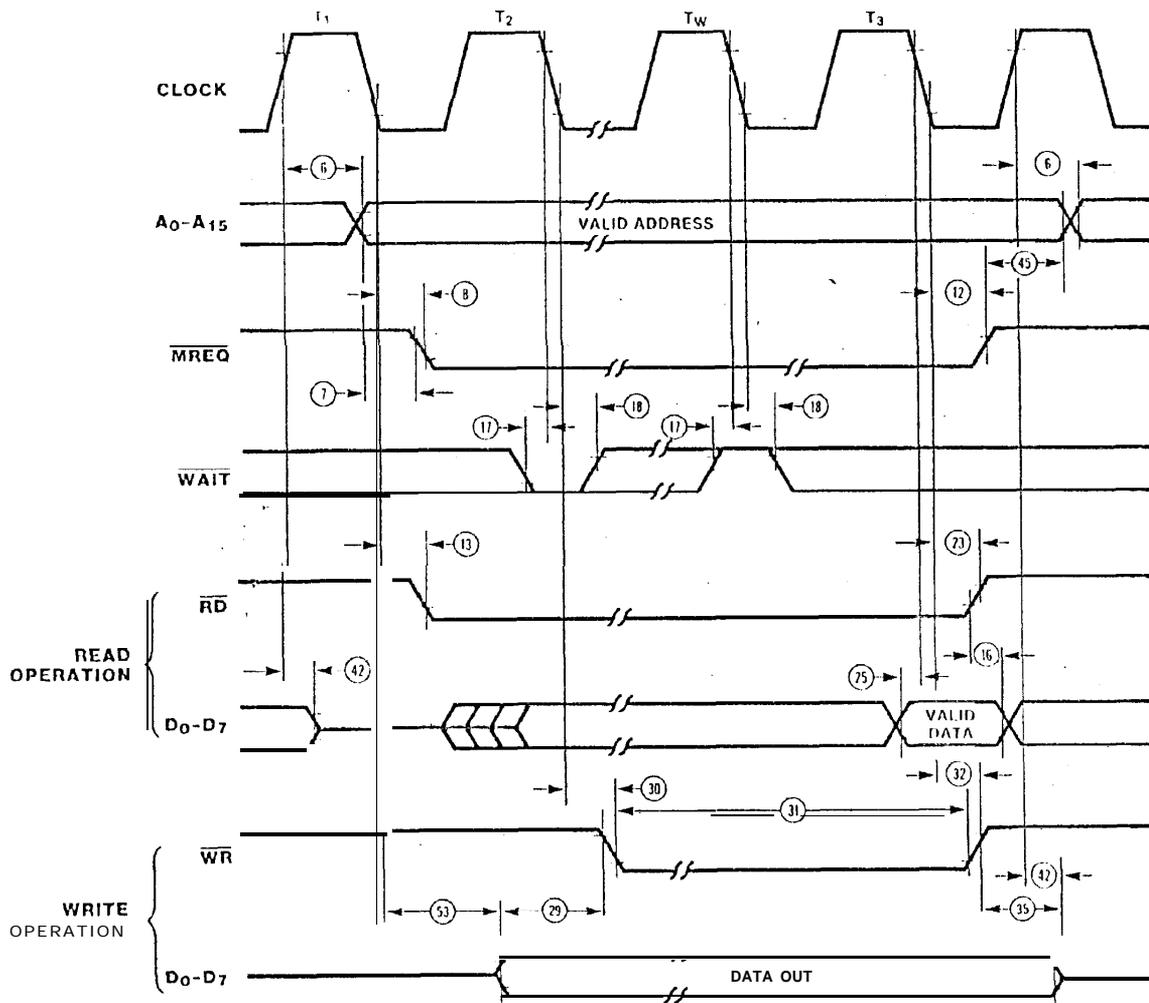
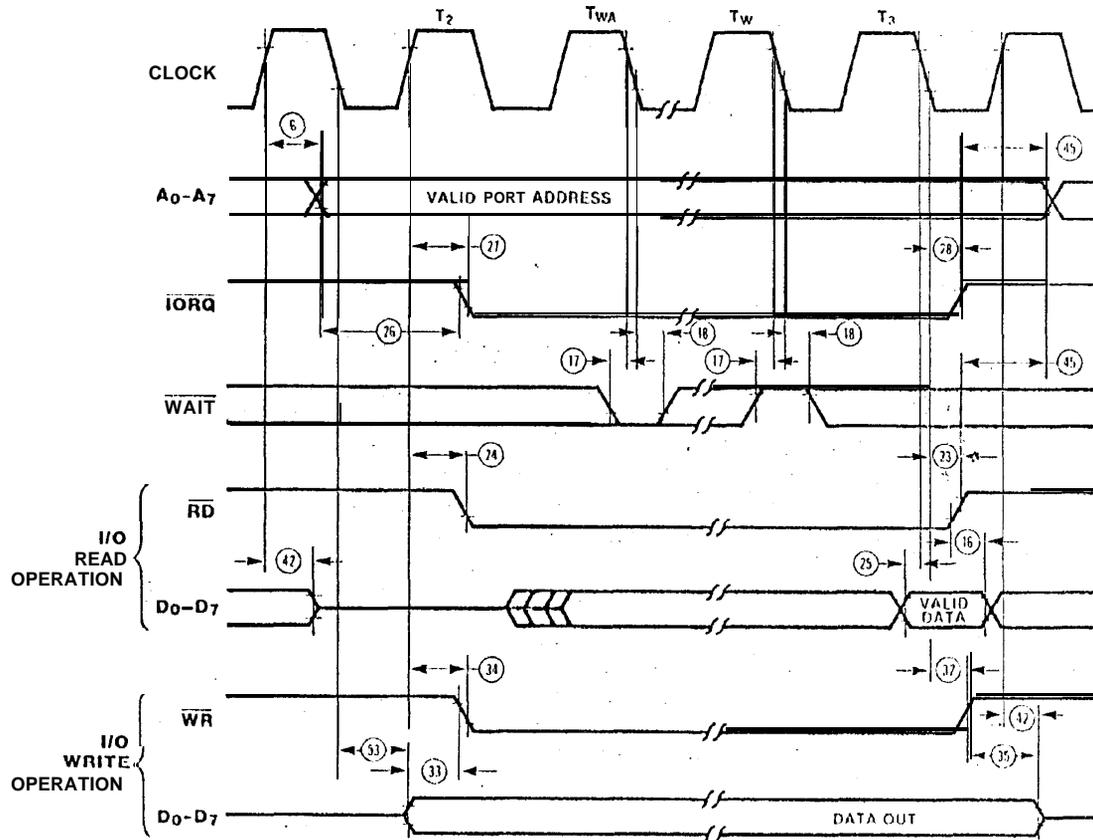


Figure 6. Memory Read or Write Cycles

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.

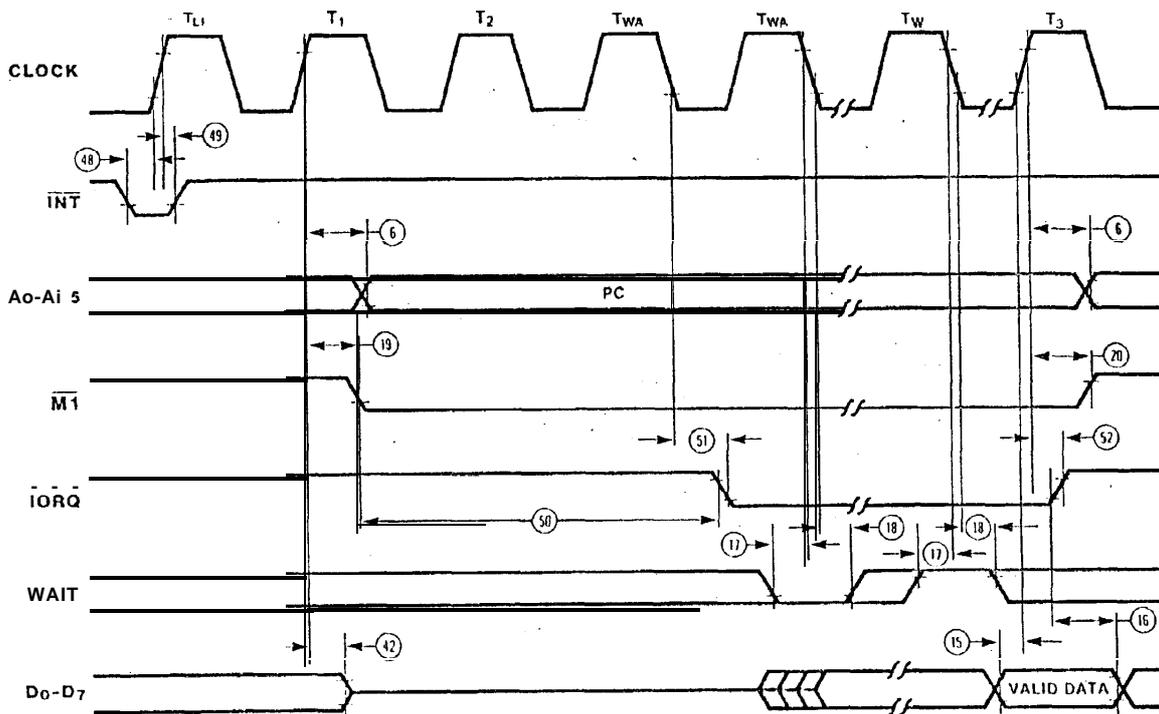


T_{WA} = One wait cycle automatically inserted by CPU

Figure 7. Input or Output Cycles

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, \overline{IORQ} becomes active (instead of \overline{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.

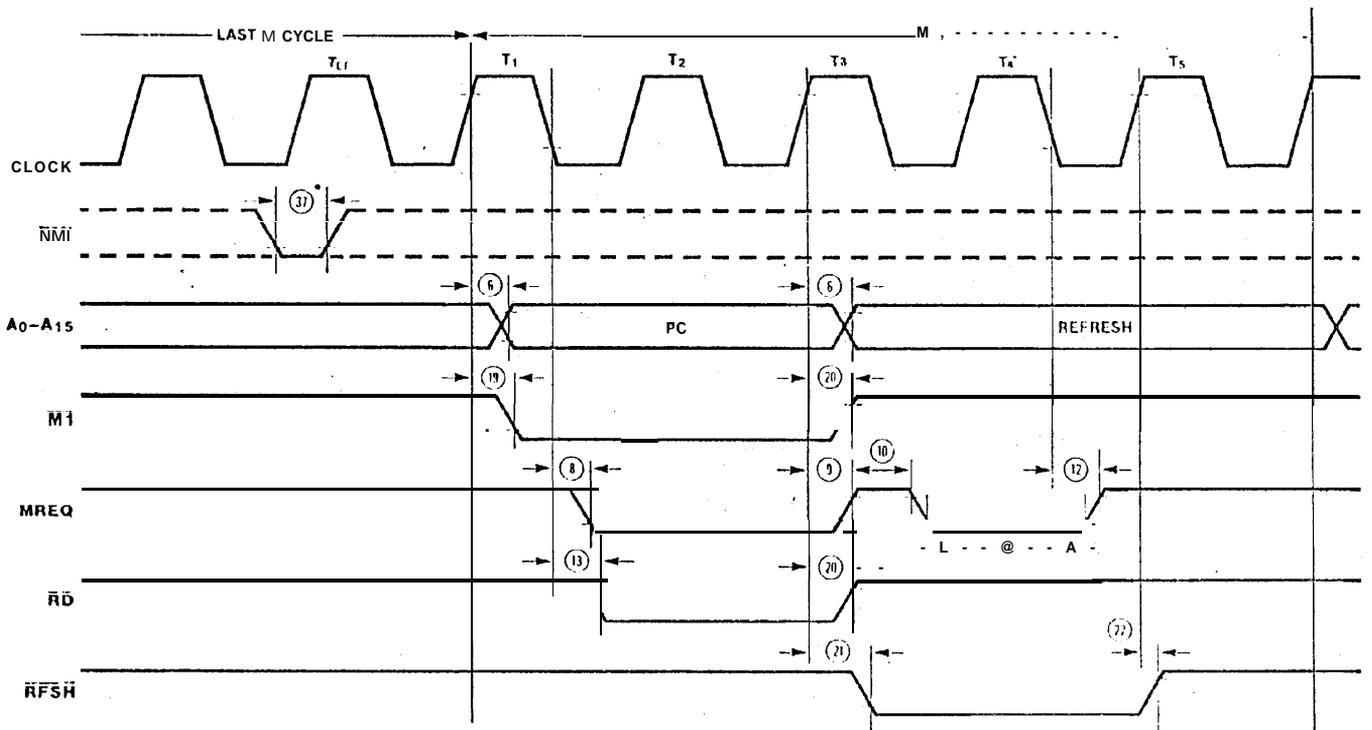


NOTES: 1) T_{L1} = Last state of any instruction cycle.
 2) T_{WA} = Wait cycle automatically inserted by CPU.

Figure 8. Interrupt Request/Acknowledge Cycle

Non-Maskable Interrupt Request 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 NMI service routine located at address 0066H (Figure 9).

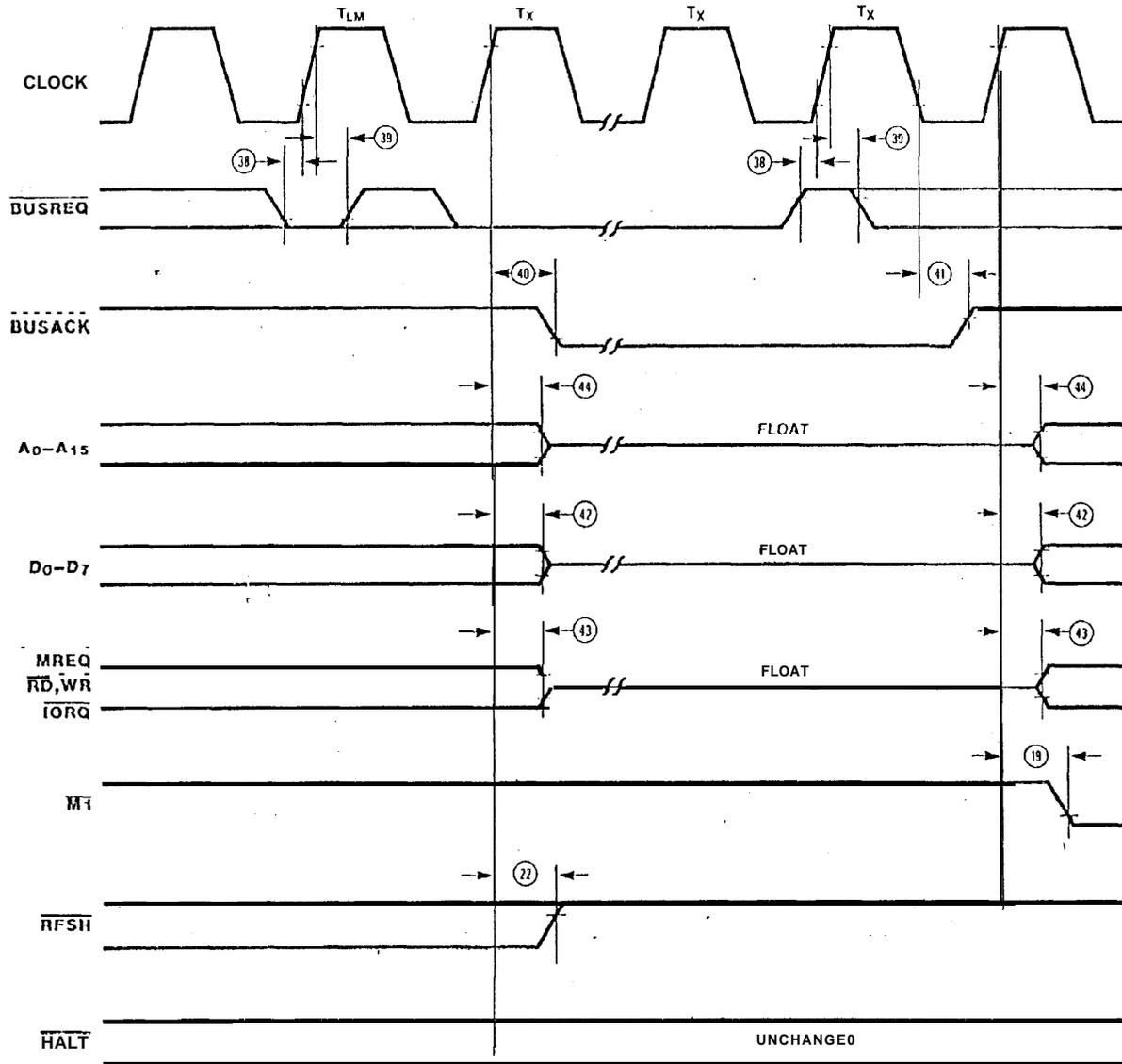


*Although $\overline{\text{NMI}}$ is an asynchronous input, to guarantee its being recognized on the following machine cycle, $\overline{\text{NMI}}$'s falling edge must occur no later than the rising edge of the clock cycle preceding the last state of any instruction cycle (T_{LI}).

Figure 9. Non-Maskable Interrupt Request Operation

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

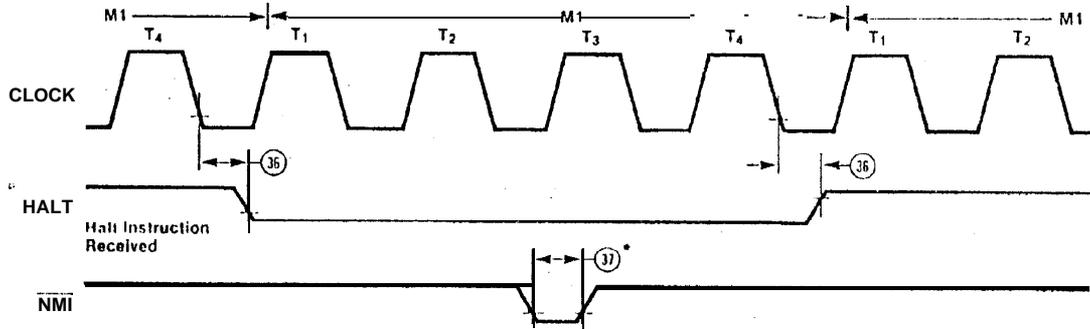


NOTES: 1) T_{LM} = Last state of any M cycle.
 2) T_x = An arbitrary clock cycle used by requesting device.

Figure 10. BUS Request/Acknowledge Cycle

Halt Acknowledge Cycle. When the CPU receives a $\overline{\text{HALT}}$ instruction, it executes NOP states until either an $\overline{\text{INT}}$ or $\overline{\text{NMI}}$ input is received. When in the Halt state, the $\overline{\text{HALT}}$ output is

active and remains so until an interrupt is received (Figure 11). $\overline{\text{INT}}$ will also force a Halt exit.



*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_{L1}).

Figure 11. Halt Acknowledge

Reset Cycle. $\overline{\text{RESET}}$ must be active for at least three clock cycles for the CPU to properly accept it. As long as $\overline{\text{RESET}}$ remains active, the address and data buses float, and the control outputs are inactive. Once $\overline{\text{RESET}}$ goes inactive, two

internal T cycles are consumed before the CPU resumes normal processing operation. $\overline{\text{RESET}}$ clears the PC register, so the first opcode fetch will be to location 0000H (Figure 12).

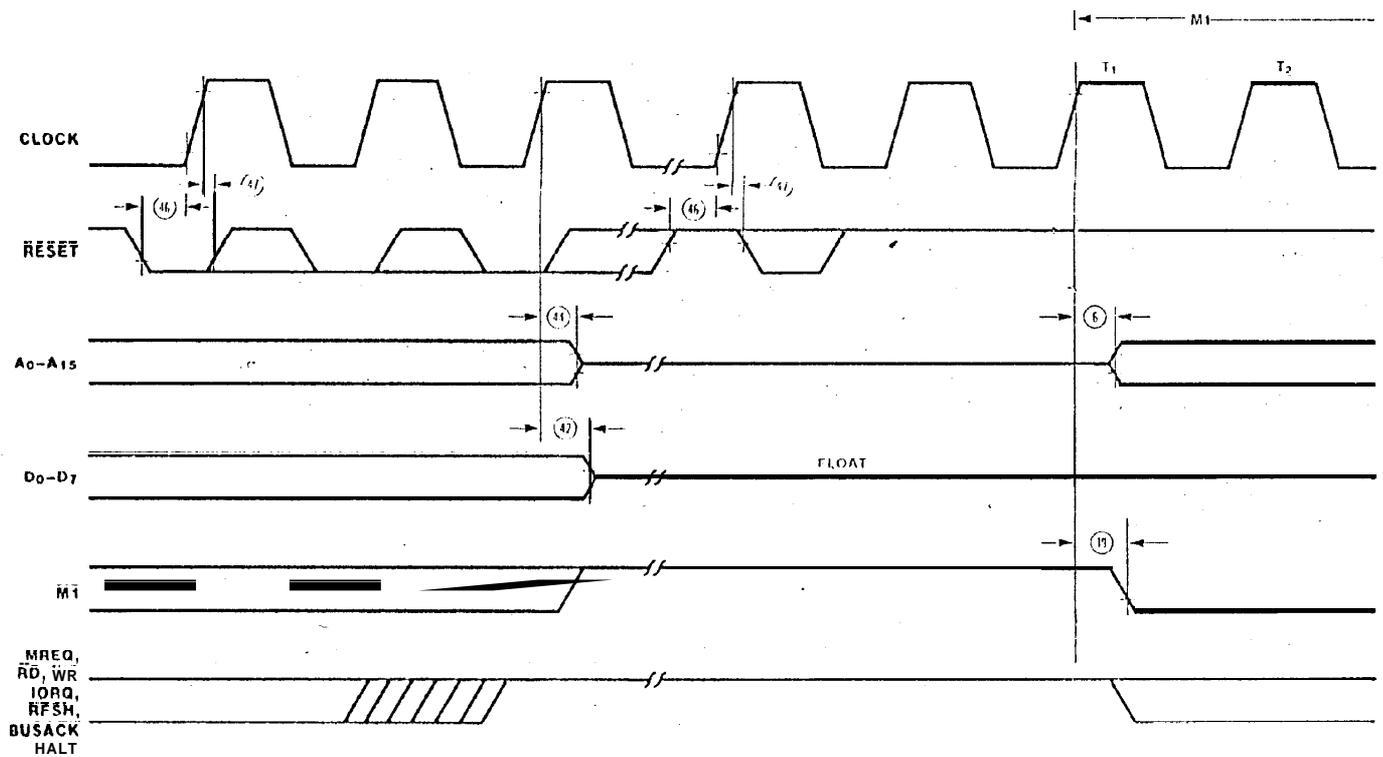


Figure 12. Reset Cycle

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

CMOS Z80 CPU supports Power-Down mode of Operation.

This mode is also referred to as the "standby mode", and supply current for the CPU goes down as low as 10 μA (Where specified as I_{cc2}).

Power-Down Acknowledge Cycle. When the clock input to the CPU is stopped at either a High or Low level, the CPU stops its operation and maintains all registers and control signals. However, I_{cc2} (standby supply current) is guaranteed only when the system clock is stopped at a Low

level during T_4 of the machine cycle 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.

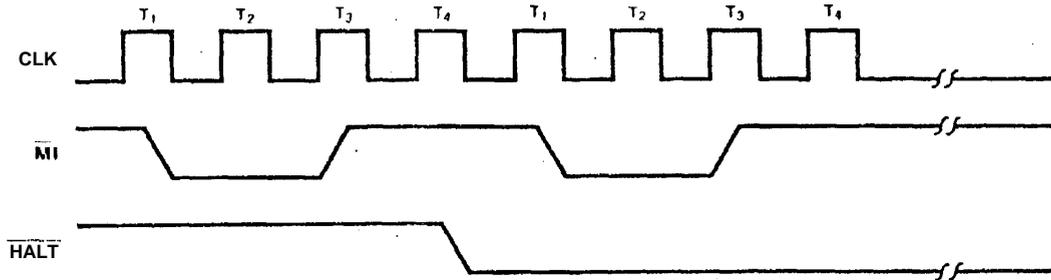


Figure 13. Power-Down Acknowledgc

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 $\overline{\text{NMI}}$ 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.

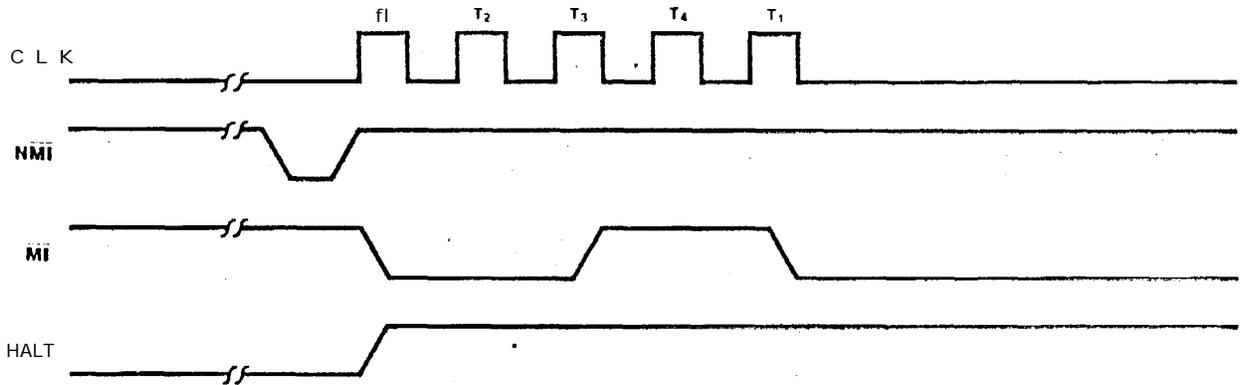


Figure 14a.

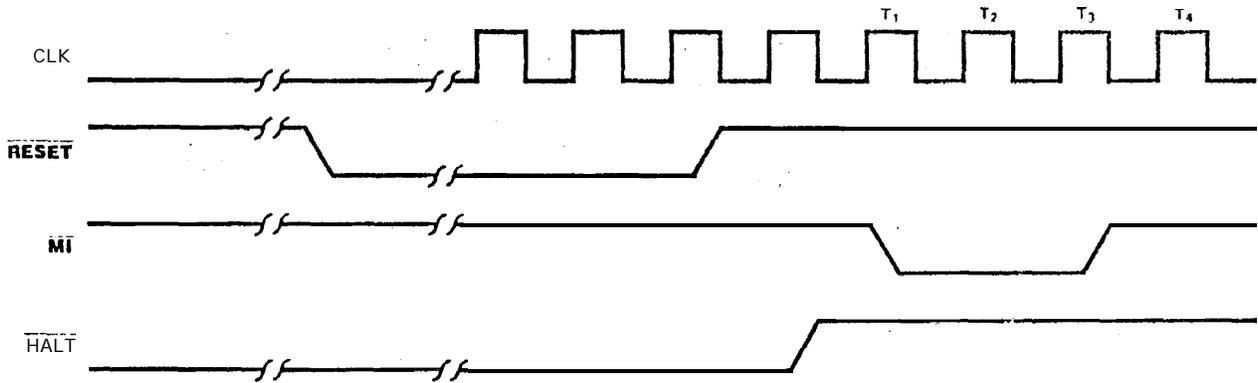


Figure 14b.

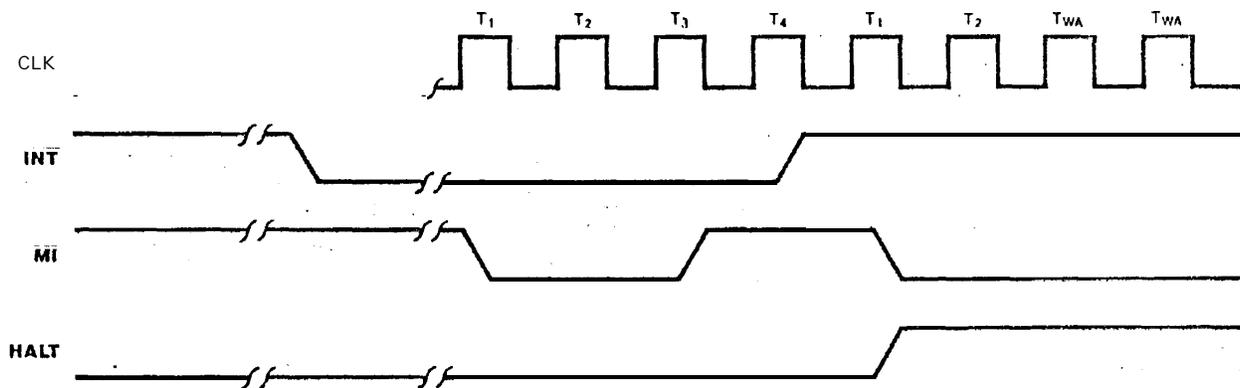


Figure 14c.

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 $-65^{\circ}C$ to $+150^{\circ}C$

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:

■ $s = 0^{\circ}C$ to $+70^{\circ}C$

Voltage Supply Range:

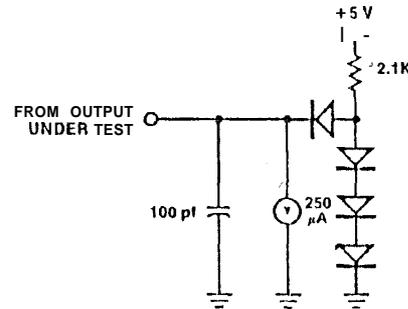
NMOS: $+4.75V \geq V_{CC} \geq +5.25V$

CMOS: $+4.50V \geq V_{CC} \geq 5.50V$

■ $E = -40^{\circ}C$ to $+100^{\circ}C$, $+4.50V \geq V_{CC} > +5.50V$

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} - 0.6	V _{CC} + 0.3	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 _{OH1}	Output High Voltage	2.4		V	I _{OH} = -1.6 mA
V _{OH2}	Output High Voltage	V _{CC} - 0.8		V	I _{OH} = -250 μA
I _{CC1}	Power Supply Current				V _{CC} = 5V
	4 MHz		20	mA	
	6 MHz		30	mA	V _{IH} = V _{CC} - 0.2V
	8 MHz		40	mA	V _{IL} = 0.2V
	10 MHz		50	mA	
I _{CC2}	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	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. A₁₅-A₀, D₇-D₀, MREQ, IORQ, RD, and WR.

3. I_{CC2} standby supply current is guaranteed only when the supplied clock is stopped at a low level during T₄ of the machine cycle immediately following the execution of a HALT instruction.

CAPACITANCE

Symbol	Parameter	Min	Max	Unit
CCLOCK	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)

Number	Symbol	Parameter	Z84C0004		Z84C0006		Z84C0008		Z84C0010	
			Min	Max	Min	Max	Min	Max	Min	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 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 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 WR ↓	80*		25*		5'		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† (Z84C00/CMOS Z80 CPU; Continued)

Number	Symbol	General Parameter	Z84C0004		Z84C0006		Z84C0008		Z84C0010	
			Min	Max	Min	Max	Min	Max	Min	Max
30	TdCf(WRf)	Clock ↓ to \overline{WR} ↓ Delay		80		70		60		55
31	TwWR	\overline{WR} Pulse Width	220*		1 3 5		100*			75*
3 2	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		6 5		6 0		55		5 0
3 5	TdWRr(D)	Data Stable from \overline{WR} ↑	60*		30*		15*		12*	
36	TdCf(HALT)	Clock ↓ to HALT ↑ or ↓		300		260		2 2 5		90
3 7	TwNMI	NMI Pulse Width	80		70		60		60	
38	TsBUSREQ(Cr)	BUSREQ Setup Time to Clock ↑	50		50		40		3 0	
3 9	ThBUSREQ(Cr)	BUSREQ Hold Time after Clock ↑	10		10		10		10	
40	TdCr(BUSACKf)	Clock ↑ to BUSACK ↓ Delay		100		90		8 0		75
41	TdCf(BUSACKr)	Clock ↓ to 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 (MREQ, IORQ, RD, and WR)		80		70		60		60
44	TdCr(Az)	Clock ↑ to Address Float Delay		90		8 0		70		65
4 5	TdCTr(A)	\overline{MREQ} ↑, \overline{IORQ} ↑, \overline{RD} ↑, and \overline{WR} ↑ to Address Hold Time	80*		35*		20*		32*	
46	TsRESET(Cr)	RESET to Clock ↑ Setup Time	60		60		45		40	
47	ThRESET(Cr)	RESET to Clock ↑ Hold Time	10		10		10		10	
4 8	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	
5 0	TdM1f(IORQf)	\overline{MI} ↓ to \overline{IORQ} ↓ Delay	565*		365*		270*		222*	
51	TdCf(IORQf)	Clock ↓ to \overline{IORQ} ↓ Delay		8 5		7 0		60		55
52	TdCf(IORQr)	Clock ↑ to \overline{IORQ} ↑ Delay		85		70		60		55
53	TdCf(D)	Clock ↓ to Data Valid Delay				150		130		110

• For clock periods other than the minimums shown, calculate parameters using the following table. Calculated values above assumed $T_{rC} = T_{fC} = 20$ ns.

†Units in nanoseconds (ns).

FOOTNOTES TO AC CHARACTERISTICS

Number	Symbol	General Parameter	Z84C0004	Z84C0006	Z84C0008	Z84C0010
1	TcC	$T_{wCh} + T_{wCl} + T_{rC} + T_{fC}$				
7	TdA(MREQf)	$T_{wCh} + T_{fC}$	-65	-50	-4 5	-45
10	TwMREQh	$T_{wCh} + T_{fC}$	-20	-20	-20	-20
11	TwMREQl	TcC	-30	-30	-25	-25
2 6	TdA(IORQf)	TcC	-70	-5 5	-50	-50
29	TdD(WRf)	TcC	-170	-140	1120	-60
31	TwWR	TcC	-30	-30	-25	-25
33	TdD(WRf)	$T_{wCl} + T_{rC}$	-140	-140	-120	-60
35	TdWRr(D)	$T_{wCl} + T_{rC}$	-7 0	-55	-5 0	-40
45	TdCTr(A)	$T_{wCl} + T_{rC}$	-50	-50	-45	-30
50	TdM1f(IORQf)	$2T_{cC} + T_{wCh} + T_{fC}$	-65	-50	-45	-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

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} - 0.6	V _{CC} + 0.3	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. A₁₅-A₀, D₇-D₀, MREQ, IORQ, RD, and 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		Z0840006		Z0840008	
			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 $\overline{\text{MREQ}} \downarrow$ Delay	65'		35*		20'	
8	TdCl(MREQf)	Clock ↓ to $\overline{\text{MREQ}} \downarrow$ Delay		85		70		60
9	TdCr(MREQr)	Clock ↑ to $\overline{\text{MREQ}} \uparrow$ Delay		85		70		60
10	TwMREQh	$\overline{\text{MREQ}}$ Pulse Width (High)	110*††		65*††		45*††	
11	TwMREQl	$\overline{\text{MREQ}}$ Pulse Width (Low)	220*††		135*††		100*††	
12	TdCl(MREQr)	Clock ↓ to $\overline{\text{MREQ}} \uparrow$ Delay		85		70		60
13	TdCl(RDf)	Clock ↓ to $\overline{\text{RD}} \downarrow$ Delay		35		80		70
14	TdCr(RDr)	Clock ↑ to $\overline{\text{RD}} \uparrow$ Delay		85		70		60
15	TsD(Cr)	Data Setup Time to Clock ↑	35		30		30	
16	ThD(RDr)	Data Hold Time to $\overline{\text{RD}} \uparrow$		0		0		0
17	TsWAIT(Cf)	$\overline{\text{WAIT}}$ Setup Time to Clock ↓	70		60		50	
18	ThWAIT(Cf)	$\overline{\text{WAIT}}$ Hold Time after Clock ↓		0		0		0
19	TdCr(M1f)	Clock ↑ to $\overline{\text{M1}} \downarrow$ Delay		100		80		70
20	TdCr(M1r)	Clock ↑ to $\overline{\text{M1}} \uparrow$ Delay		100		80		70
21	TdCr(RFSHf)	Clock ↑ to $\overline{\text{RFSH}} \downarrow$ Delay		130		110		95
22	TdCr(RFSHr)	Clock ↑ to $\overline{\text{RFSH}} \uparrow$ Delay		120		100		85
23	TdCl(RDr)	Clock ↓ to $\overline{\text{RD}} \uparrow$ Delay		85		70		60
24	TdCr(RDf)	Clock ↑ to $\overline{\text{RD}} \downarrow$ 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 $\overline{\text{IORQ}} \downarrow$	180*		110"		75'	
27	TdCr(IORQf)	Clock ↑ to $\overline{\text{IORQ}} \downarrow$ Delay		75		65		55
28	TdCl(IORQr)	Clock ↓ to $\overline{\text{IORQ}} \uparrow$ Delay		85		70		60
29	TdD(WRf)	Data Stable prior to $\overline{\text{WR}} \downarrow$	80"		25*		5*	
30	TdCl(WRf)	Clock ↓ to $\overline{\text{WR}} \downarrow$ Delay		80		70		60
31	TwWR	$\overline{\text{WR}}$ Pulse Width	220'		135*		100'	
32	TdCl(WRr)	Clock ↓ to $\overline{\text{WR}} \uparrow$ Delay		ao.		70		60
33	TdD(WRf)	Data Stable prior to $\overline{\text{WR}} \downarrow$	-10'		-55*		55"	
34	TdCr(WRf)	Clock ↑ to $\overline{\text{WR}} \downarrow$ Delay		65		60		55
35	TdWRr(D)	Data Stable from $\overline{\text{WR}} \uparrow$	60*		30'		15"	
36	TdCl(HALT)	Clock ↓ to HALT ↑ or ↓		300		260		225
37	TwNMI	NMI Pulse Width.	80		70		60'	
38	TsBUSREQ(Cr)	$\overline{\text{BUSREQ}}$ 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.

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

Number	Symbol	Parameter	20840004		Z0840006		20840008		
			Min	Max	Min	Max	Min	Max	
3	9	ThBUSREQ(Cr)	BUSREQ Hold Time after Clock ↑						0
4	0	TdCr(BUSACKf)	Clock ↑ to BUSACK ↓ Delay						100
4	1	TdCl(BUSACKr)	Clock ↓ to BUSACK ↑ Delay						100
42		TdCr(Dz)	Clock ↑ to Data Float Delay						90
43		TdCr(CTz)	Clock ↑ to Control Outputs Float Delay (MREQ, IORQ, RD, and WR)						80
4	4	TdCr(Az)	Clock ↑ to Address Float Delay						90
45		TdCr(A)	MREQ ↑, IORQ ↑, RD ↑, and WR ↑ to Address Hold Time						80*
46		TsRESET(Cr)	RESET to Clock ↑ Setup Time						60
47		ThRESET(Cr)	RESET to Clock ↑ Hold Time						0
48		TsINTf(Cr)	INT to Clock ↑ Setup Time						80
49		ThINTr(Cr)	INT to Clock ↑ Hold Time						0
50		TdM1f(IORQf)	M1 ↓ to IORQ ↓ Delay						565*
51		TdCl(IORQf)	Clock ↓ to IORQ ↓ Delay						85
52		TdCl(IORQr)	Clock ↑ IORQ ↑ Delay						85
53		TdCl(D)	Clock 3 to Data Valid Delay						150

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

†Units in nanoseconds (ns).

FOOTNOYES YO AC CHARACTERISTICS

Number	Symbol	General Parameter	20840004	Z0840006	Z0840008
1	TcC	TwCh + TwCl + TrC + TIC			
7	TdA(MREQf)	TwCh + TIC	-65	-50	-45
10	TwMREQh	TwCh + TIC	-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 + TIC	-65	-50	-45

AC Test Conditions:

V_{IH} = 2.0 V V_{OH} = 1.5 V
V_{IL} = 0.8 V V_{OL} = 1.5 V
V_{IHC} = V_{CC} - 0.6 V FLOAT = ±0.5 V
V_{ILC} = 0.45 V