FUNCTIONAL DESCRIPTION

Microprocessors like the 8080 are MOS devices and are generally capable of driving a single TTL load. The same is true for MOS memory devices. While this type of drive is sufficient in small systems with few components, quite often it is necessary to buffer the microprocessor and memories when adding components or expanding to a multi-board system.

The 8216/8226 is a four bit bi-directional bus driver specifically designed to buffer microcomputer system components.

Bi-Directional Driver

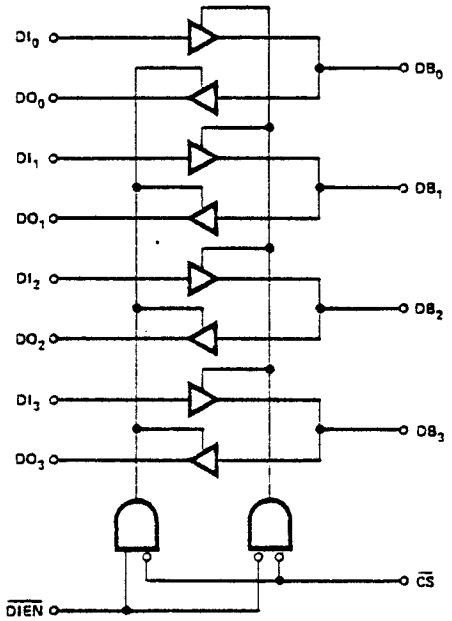
Each buffered line of the four bit driver consists of two separate buffers that are tri-state in nature to achieve direct bus interface and bi-directional capability. On one side of the driver the output of one buffer and the input of another are tied together (DB), this side is used to interface to the system side components such as memories, I/O, etc., because its interface is direct TTL compatible and it has high drive (50mA). On the other side of the driver the inputs and outputs are separated to provide maximum flexibility. Of course, they can be tied together so that the driver can be used to buffer a true bi-directional bus such as the 8080 Data Bus. The DO outputs on this side of the driver have a special high voltage output drive capability (3.65V) so that direct interface to the 8080 and 8008 CPUs is achieved with an adequate amount of noise immunity (350mV worst case).

Control Gating $\overline{DIEN}$, $\overline{CS}$

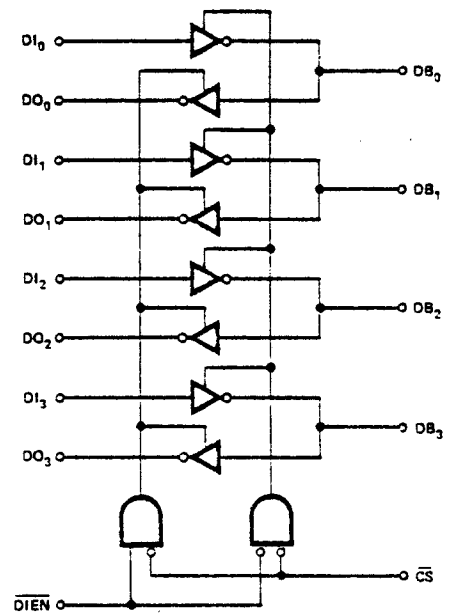
The $\overline{CS}$ input is actually a device select. When it is "high" the output drivers are all forced to their high-impedance state. When it is at "zero" the device is selected (enabled) and the direction of the data flow is determined by the $\overline{DIEN}$ input.

The $\overline{DIEN}$ input controls the direction of data flow (see Figure 1) for complete truth table. This direction control is accomplished by forcing one of the pair of buffers into its high impedance state and allowing the other to transmit its data. A simple two gate circuit is used for this function.

The 8216/8226 is a device that will reduce component count in microcomputer systems and at the same time enhance noise immunity to assure reliable, high performance operation.



(a) 8216



(b) 8226

$\overline{DIEN}$	$\overline{CS}$	
0	0	DI → DB
1	0	DB → DO
0	1	– HIGH IMPEDANCE
1	1	

Figure 1. 8216/8226 Logic Diagrams

D.C. AND OPERATING CHARACTERISTICS

ABSOLUTE MAXIMUM RATINGS*

Temperature Under Bias . . . . . 0°C to 70°C
Storage Temperature . . . . . -65°C to +150°C
All Output and Supply Voltages . . . . . -0.5V to +7V
All Input Voltages . . . . . -1.0V to +5.5V
Output Currents . . . . . 125 mA

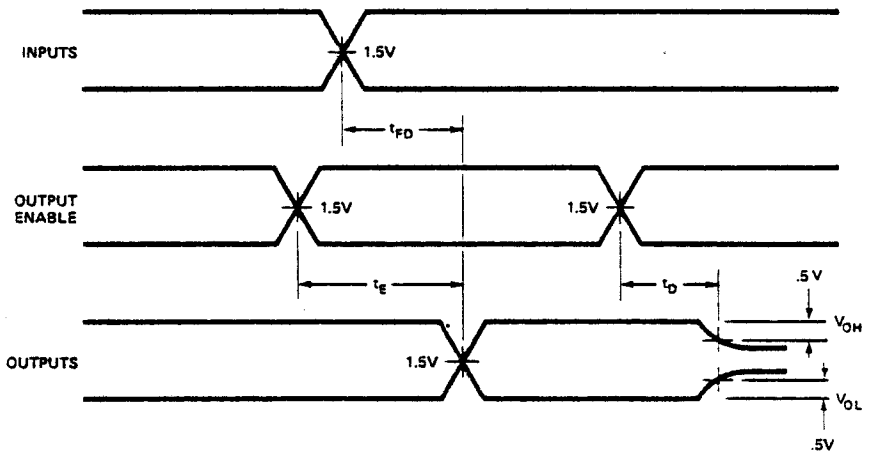
*COMMENT: Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or at any other condition above those indicated in the operational sections of this specification is not implied.

T_A = 0°C to +70°C, V_{CC}=+5V ±5%

Symbol	Parameter		Limits Min.	Typ.	Max.	Unit	Conditions
I_{F1}	Input Load Current $\overline{DIEN}$, $\overline{CS}$			-0.15	-.5	mA	V_F = 0.45
I_{F2}	Input Load Current All Other Inputs			-0.08	-.25	mA	V_F = 0.45
I_{R1}	Input Leakage Current $\overline{DIEN}$, $\overline{CS}$				20	μA	V_R = 5.25V
I_{R2}	Input Leakage Current DI Inputs				10	μA	V_R = 5.25V
V_C	Input Forward Voltage Clamp				-1	V	I_C = -5mA
V_{IL}	Input "Low" Voltage				.95	V	
V_{IH}	Input "High" Voltage		2.0			V	
$\|I_O\|$	Output Leakage Current (3-State)	DO DB			20 100	μA	V_O = 0.45V/5.25V
I_{CC}	Power Supply Current	8216		95	130	mA	
		8226		85	120	mA	
V_{OL1}	Output "Low" Voltage			0.3	.45	V	DO Outputs I_{OL}=15mA DB Outputs I_{OL}=25mA
V_{OL2}	Output "Low" Voltage	8216		0.5	.6	V	DB Outputs I_{OL}=55mA
		8226		0.5	.6	V	DB Outputs I_{OL}=50mA
V_{OH1}	Output "High" Voltage		3.65	4.0		V	DO Outputs I_{OH} = -1mA
V_{OH2}	Output "High" Voltage		2.4	3.0		V	DB Outputs I_{OH} = -10mA
I_{OS}	Output Short Circuit Current		-15 -30	-35 -75	-65 -120	mA mA	DO Outputs $V_O \cong 0V$, DB Outputs V_{CC}=5.0V

NOTE: Typical values are for T_A = 25°C, V_{CC} = 5.0V.

WAVEFORMS



A.C. CHARACTERISTICS

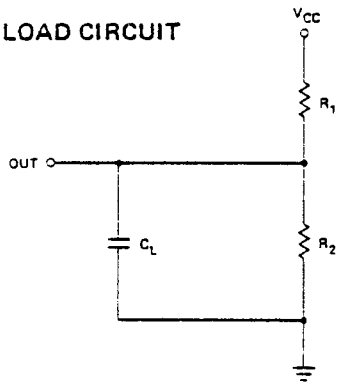
T_A = 0°C to +70°C, V_{CC} = +5V ±5%

Symbol	Parameter	Limits Min.	Typ.[1]	Max.	Unit	Conditions
T_{PD1}	Input to Output Delay DO Outputs		15	25	ns	C_L=30pF, R_1=300Ω R_2=600Ω
T_{PD2}	Input to Output Delay DB Outputs					
	8216		20	30	ns	C_L=300pF, R_1=90Ω
	8226		16	25	ns	R_2 = 180Ω
T_E	Output Enable Time					
	8216		45	65	ns	(Note 2)
	8226		35	54	ns	(Note 3)
T_D	Output Disable Time		20	35	ns	(Note 4)

TEST CONDITIONS:

Input pulse amplitude of 2.5V.
Input rise and fall times of 5 ns between 1 and 2 volts.
Output loading is 5 mA and 10 pF.
Speed measurements are made at 1.5 volt levels.

TEST LOAD CIRCUIT



Capacitance[5]

Symbol	Parameter	Limits Min.	Typ.[1]	Max.	Unit
C_{IN}	Input Capacitance		4	8	pF
C_{OUT1}	Output Capacitance		6	10	pF
C_{OUT2}	Output Capacitance		13	18	pF

TEST CONDITIONS: V_{BIAS} = 2.5V, V_{CC} = 5.0V, T_A = 25°C, f = 1 MHz.

NOTES:
1. Typical values are for T_A = 25°C, V_{CC} = 5.0V.
2. DO Outputs, C_L = 30pF, R_1 = 300/10 KΩ, R_2 = 180/1KΩ; DB Outputs, C_L = 300pF, R_1 = 90/10 KΩ, R_2 = 180/1 KΩ.
3. DO Outputs, C_L = 30pF, R_1 = 300/10 KΩ, R_2 = 600/1K; DB Outputs, C_L = 300pF, R_1 = 90/10 KΩ, R_2 = 180/1 KΩ.
4. DO Outputs, C_L = 5pF, R_1 = 300/10 KΩ, R_2 = 600/1 KΩ; DB Outputs, C_L = 5pF, R_1 = 90/10 KΩ, R_2 = 180/1 KΩ.
5. This parameter is periodically sampled and not 100% tested.

Schottky Bipolar 8224

CLOCK GENERATOR AND DRIVER FOR 8080A CPU

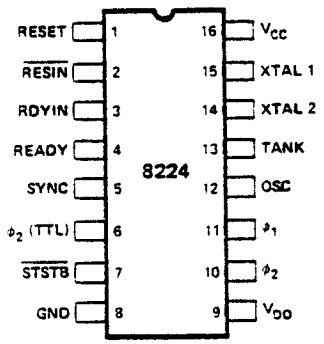
- **Single Chip Clock Generator/Driver for 8080A CPU**
- **Power-Up Reset for CPU**
- **Ready Synchronizing Flip-Flop**
- **Advanced Status Strobe**
- **Oscillator Output for External System Timing**
- **Crystal Controlled for Stable System Operation**
- **Reduces System Package Count**

The 8224 is a single chip clock generator/driver for the 8080A CPU. It is controlled by a crystal, selected by the designer, to meet a variety of system speed requirements.

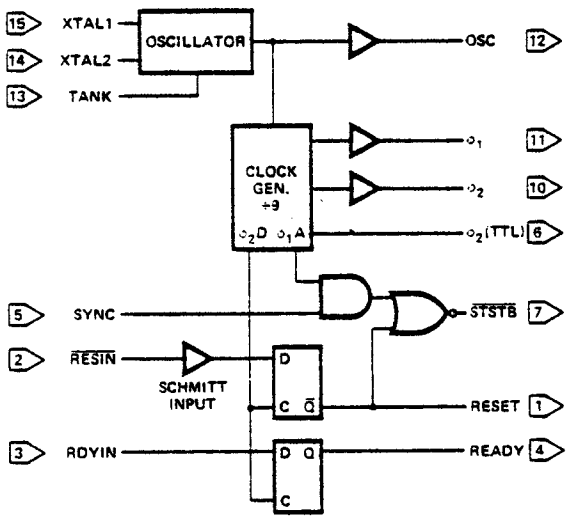
Also included are circuits to provide power-up reset, advance status strobe and synchronization of ready.

The 8224 provides the designer with a significant reduction of packages used to generate clocks and timing for 8080A.

PIN CONFIGURATION



BLOCK DIAGRAM



PIN NAMES

$\overline{RESIN}$	RESET INPUT
RESET	RESET OUTPUT
RDYIN	READY INPUT
READY	READY OUTPUT
SYNC	SYNC INPUT
$\overline{STSTB}$	STATUS STB (ACTIVE LOW)
ϕ_1	8080 CLOCKS
ϕ_2	8080 CLOCKS

XTAL 1	CONNECTIONS FOR CRYSTAL
XTAL 2	CONNECTIONS FOR CRYSTAL
TANK	USED WITH OVERTONE XTAL
OSC	OSCILLATOR OUTPUT
ϕ_2 (TTL)	ϕ_2 CLK (TTL LEVEL)
V_{CC}	+5V
V_{DD}	+12V
GND	0V

FUNCTIONAL DESCRIPTION

General

The 8224 is a single chip Clock Generator/Driver for the 8080A CPU. It contains a crystal-controlled oscillator, a "divide by nine" counter, two high-level drivers and several auxiliary logic functions.

Oscillator

The oscillator circuit derives its basic operating frequency from an external, series resonant, fundamental mode crystal. Two inputs are provided for the crystal connections (XTAL1, XTAL2).

The selection of the external crystal frequency depends mainly on the speed at which the 8080A is to be run at. Basically, the oscillator operates at 9 times the desired processor speed.

A simple formula to guide the crystal selection is:

$$\text{Crystal Frequency} = \frac{1}{t_{CY}} \text{ times } 9$$

Example 1: (500ns t_{CY})
2mHz times 9 = 18mHz*

Example 2: (800ns t_{CY})
1.25mHz times 9 = 11.25mHz

Another input to the oscillator is TANK. This input allows the use overtone mode crystals. This type of crystal generally has much lower "gain" than the fundamental type so an external LC network is necessary to provide the additional "gain" for proper oscillator operation. The external LC network is connected to the TANK input and is AC coupled to ground. See Figure 4.

The formula for the LC network is:

$$F = \frac{1}{2\pi\sqrt{LC}}$$

The output of the oscillator is buffered and brought out on ØSC (pin 12) so that other system timing signals can be derived from this stable, crystal-controlled source.

*When using crystals above 10mHz a small amount of frequency "trimming" may be necessary to produce the exact desired frequency. The addition of a small selected capacitance (3pF - 10pF) in series with the crystal will accomplish this function.

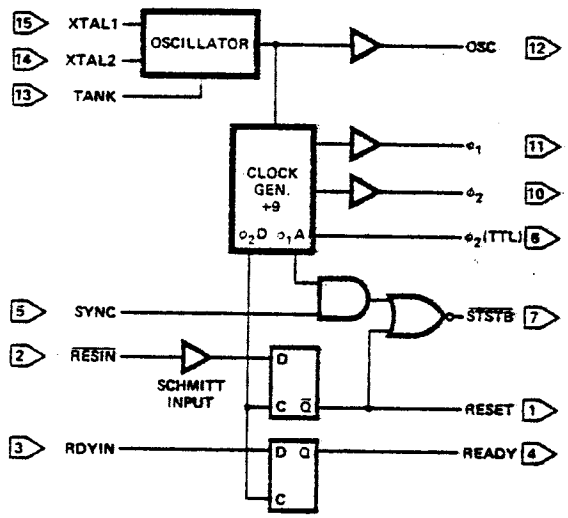
Clock Generator

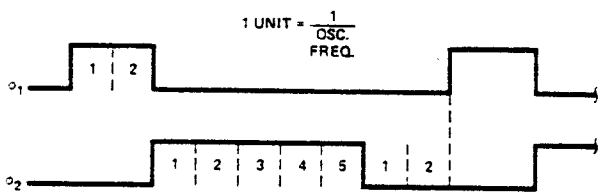
The Clock Generator consists of a synchronous "divide by nine" counter and the associated decode gating to create the waveforms of the two 8080A clocks and auxiliary timing signals.

The waveforms generated by the decode gating follow a simple 2-5-2 digital pattern. See Figure 2. The clocks generated; phase 1 and phase 2, can best be thought of as consisting of "units" based on the oscillator frequency. Assume that one "unit" equals the period of the oscillator frequency. By multiplying the number of "units" that are contained in a pulse width or delay, times the period of the oscillator frequency, the approximate time in nanoseconds can be derived.

The outputs of the clock generator are connected to two high level drivers for direct interface to the 8080A CPU. A TTL level phase 2 is also brought out ϕ_2 (TTL) for external timing purposes. It is especially useful in DMA dependant activities. This signal is used to gate the requesting device onto the bus once the 8080A CPU issues the Hold Acknowledgement (HLDA).

Several other signals are also generated internally so that optimum timing of the auxiliary flip-flops and status strobe ($\overline{STSTB}$) is achieved.





EXAMPLE: (8080 t_{CY} = 500ns)
OSC = 18mHz/55ns
ϕ_1 = 110ns (2 x 55ns)
ϕ_2 = 275ns (5 x 55ns)
ϕ_2-ϕ_1 = 110ns (2 x 55ns)

$\overline{\text{STSTB}}$ (Status Strobe)

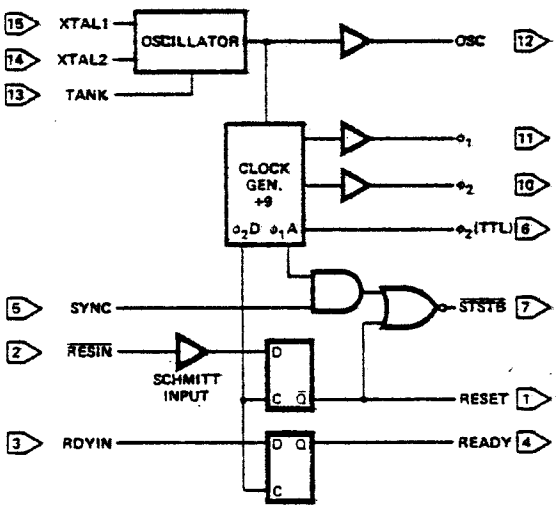
At the beginning of each machine cycle the 8080A CPU issues status information on its data bus. This information tells what type of action will take place during that machine cycle. By bringing in the SYNC signal from the CPU, and gating it with an internal timing signal (ϕ1A), an active low strobe can be derived that occurs at the start of each machine cycle at the earliest possible moment that status data is stable on the bus. The $\overline{\text{STSTB}}$ signal connects directly to the 8228 System Controller.

The power-on Reset also generates $\overline{\text{STSTB}}$, but of course, for a longer period of time. This feature allows the 8228 to be automatically reset without additional pins devoted for this function.

Power-On Reset and Ready Flip-Flops

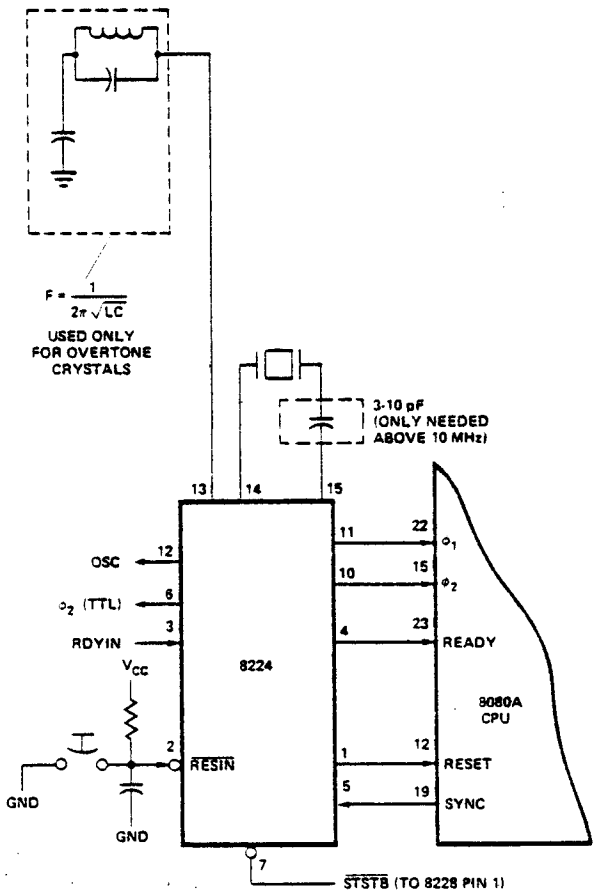
A common function in 8080A Microcomputer systems is the generation of an automatic system reset and start-up upon initial power-on. The 8224 has a built in feature to accomplish this feature.

An external RC network is connected to the $\overline{\text{RESIN}}$ input. The slow transition of the power supply rise is sensed by an internal Schmitt Trigger. This circuit converts the slow transition into a clean, fast edge when its input level reaches a predetermined value. The output of the Schmitt Trigger is connected to a "D" type flip-flop that is clocked with ϕ2D (an internal timing signal). The flip-flop is synchronously reset and an active high level that complies with the 8080A input spec is generated. For manual switch type system Reset circuits, an active low switch closing can be connected to the $\overline{\text{RESIN}}$ input in addition to the power-on RC net-network.



The READY input to the 8080A CPU has certain timing specifications such as "set-up and hold" thus, an external synchronizing flip-flop is required. The 8224 has this feature built-in. The RDYIN input presents the asynchronous "wait request" to the "D" type flip-flop. By clocking the flip-flop with ϕ2D, a synchronized READY signal at the correct input level, can be connected directly to the 8080A.

The reason for requiring an external flip-flop to synchronize the "wait request" rather than internally in the 8080 CPU is that due to the relatively long delays of MOS logic such an implementation would "rob" the designer of about 200ns during the time his logic is determining if a "wait" is necessary. An external bipolar circuit built into the clock generator eliminates most of this delay and has no effect on component count.



D.C. Characteristics

T_A = 0°C to 70°C; V_{CC} = +5.0V ±5%; V_{DD} = +12V ±5%.

Symbol	Parameter	Limits			Units	Test Conditions
		Min.	Typ.	Max.		
I_F	Input Current Loading			-.25	mA	V_F = .45V
I_R	Input Leakage Current			10	μA	V_R = 5.25V
V_C	Input Forward Clamp Voltage			1.0	V	I_C = -5mA
V_{IL}	Input "Low" Voltage			.8	V	V_{CC} = 5.0V
V_{IH}	Input "High" Voltage	2.6 2.0			V	Reset Input All Other Inputs
V_{IH}-V_{IL}	REDIN Input Hysteresis	.25			mV	V_{CC} = 5.0V
V_{OL}	Output "Low" Voltage			.45 .45	V V	(ϕ_1,ϕ_2), Ready, Reset, $\overline{STSTB}$ I_{OL} =2.5mA All Other Outputs I_{OL} = 15mA
V_{OH}	Output "High" Voltage ϕ_1 , ϕ_2 READY, RESET All Other Outputs	 9.4 3.6 2.4			 V V V	 I_{OH} = -100μA I_{OH} = -100μA I_{OH} = -1mA
I_{SC}[1]	Output Short Circuit Current (All Low Voltage Outputs Only)	-10		-60	mA	V_O = 0V V_{CC} = 5.0V
I_{CC}	Power Supply Current			115	mA	
I_{DD}	Power Supply Current			12	mA	

Note: 1. Caution, ϕ_1 and ϕ_2 output drivers do not have short circuit protection

CRYSTAL REQUIREMENTS

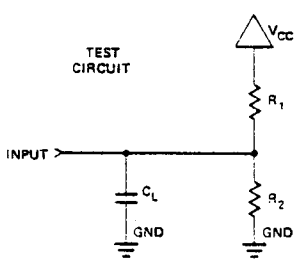
Tolerance: .005% at 0°C -70°C
Resonance: Series (Fundamental)*
Load Capacitance: 20-35pF
Equivalent Resistance: 75-20 ohms
Power Dissipation (Min): 4mW

*With tank circuit use 3rd overtone mode.

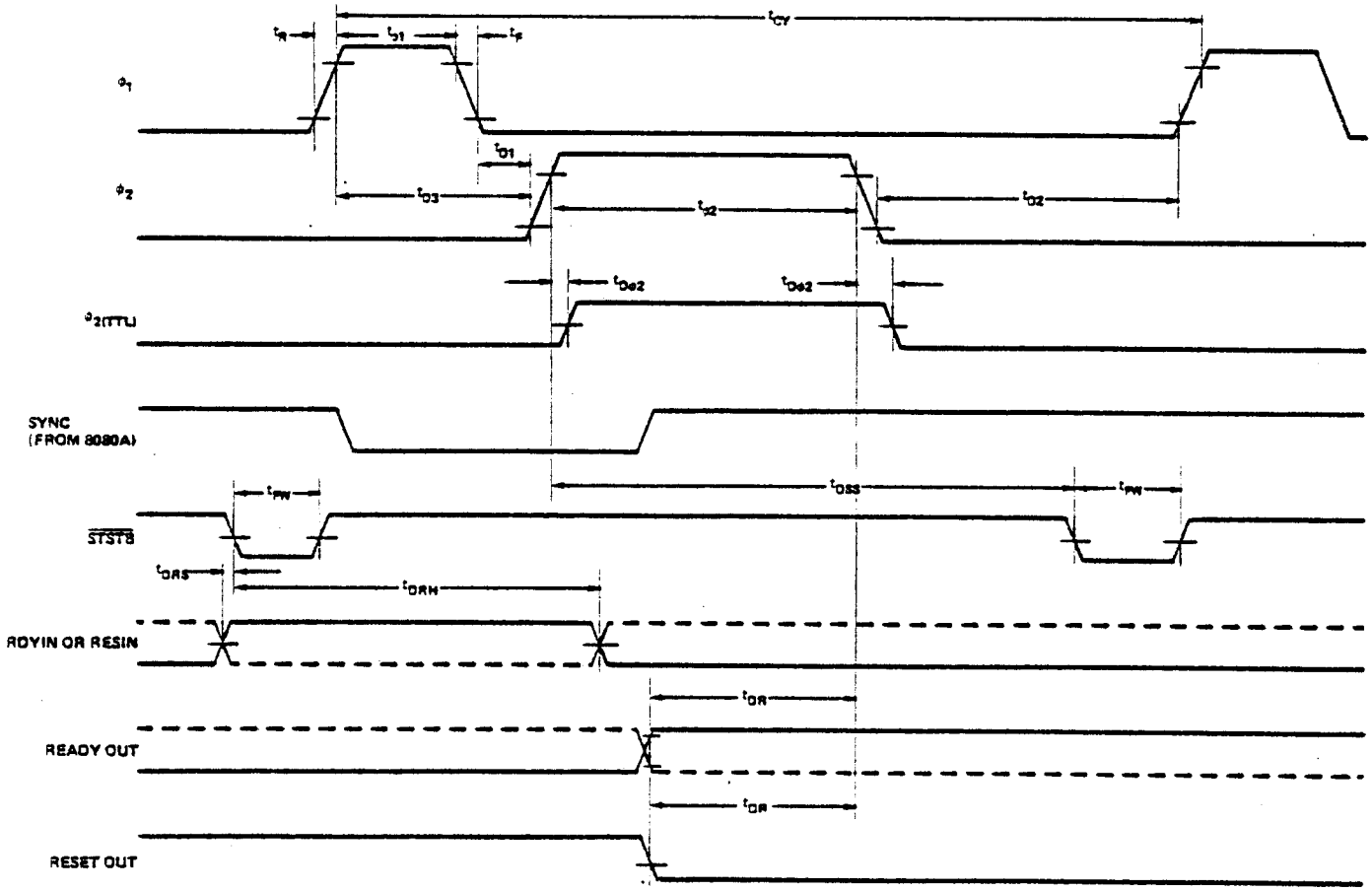
A.C. Characteristics

V_{CC} = +5.0V ± 5%; V_{DD} = +12.0V ± 5%; T_A = 0°C to 70°C

Symbol	Parameter	Limits			Units	Test Conditions
		Min.	Typ.	Max.		
$t_{\phi 1}$	ϕ_1 Pulse Width	$\frac{2tcy}{9}$ - 20ns			ns	C_L = 20pF to 50pF
$t_{\phi 2}$	ϕ_2 Pulse Width	$\frac{5tcy}{9}$ - 35ns				
t_{D1}	ϕ_1 to ϕ_2 Delay	0				
t_{D2}	ϕ_2 to ϕ_1 Delay	$\frac{2tcy}{9}$ - 14ns				
t_{D3}	ϕ_1 to ϕ_2 Delay	$\frac{2tcy}{9}$		$\frac{2tcy}{9}$ + 20ns		
t_R	ϕ_1 and ϕ_2 Rise Time			20		
t_F	ϕ_1 and ϕ_2 Fall Time			20		
$t_{D\phi 2}$	ϕ_2 to ϕ_2 (TTL) Delay	-5		+15	ns	ϕ_2TTL,CL=30 R_1=300Ω R_2=600Ω
t_{DSS}	ϕ_2 to $\overline{STSTB}$ Delay	$\frac{6tcy}{9}$ - 30ns		$\frac{6tcy}{9}$		$\overline{STSTB}$, CL=15pF R_1 = 2K R_2 = 4K
t_{PW}	$\overline{STSTB}$ Pulse Width	$\frac{tcy}{9}$ - 15ns				
t_{DRS}	RDYIN Setup Time to Status Strobe	50ns - $\frac{4tcy}{9}$				
t_{DRH}	RDYIN Hold Time After $\overline{STSTB}$	$\frac{4tcy}{9}$				
t_{DR}	RDYIN or RESIN to ϕ_2 Delay	$\frac{4tcy}{9}$ - 25ns				Ready & Reset CL=10pF R_1=2K R_2=4K
t_{CLK}	CLK Period		$\frac{tcy}{9}$			
f_{max}	Maximum Oscillating Frequency	27			MHz	
C_{in}	Input Capacitance			8	pF	V_{CC}=+5.0V V_{DD}=+12V V_{BIAS}=2.5V f=1MHz



WAVEFORMS



VOLTAGE MEASUREMENT POINTS: ϕ_1, ϕ_2 Logic "0" = 1.0V, Logic "1" = 8.0V. All other signals measured at 1.5V.

EXAMPLE:

A.C. Characteristics (For t_{CY} = 488.28 ns)

T_A = 0°C to 70°C; V_{DD} = +5V ±5%; V_{DD} = +12V ±5%.

Symbol	Parameter	Limits Min.	Typ.	Max.	Units	Test Conditions
$t_{\phi 1}$	ϕ_1 Pulse Width	89			ns	t_{CY}=488.28ns
$t_{\phi 2}$	ϕ_2 Pulse Width	236			ns	
t_{D1}	Delay ϕ_1 to ϕ_2	0			ns	
t_{D2}	Delay ϕ_2 to ϕ_1	95			ns	ϕ_1 & ϕ_2 Loaded to C_L = 20 to 50pF
t_{D3}	Delay ϕ_1 to ϕ_2 Leading Edges	109		129	ns	
t_r	Output Rise Time			20	ns	
t_f	Output Fall Time			20	ns	
t_{DSS}	ϕ_2 to $\overline{STSTB}$ Delay	296		326	ns	
$t_{D\phi 2}$	ϕ_2 to ϕ_2 (TTL) Delay	-5		+15	ns	
t_{PW}	Status Strobe Pulse Width	40			ns	Ready & Reset Loaded to 2mA/10pF
t_{DRS}	RDYIN Setup Time to $\overline{STSTB}$	-167			ns	All measurements referenced to 1.5V unless specified otherwise.
t_{DRH}	RDYIN Hold Time after $\overline{STSTB}$	217			ns	
t_{DR}	READY or RESET to ϕ_2 Delay	192			ns	
f_{MAX}	Oscillator Frequency			18.432	MHz	

3-4. SCHEMATIC REFERENCING

The detailed schematics of the Interface circuit, CPU circuit, and Display/Control panel are provided to aid in determining signal direction and tracing. A solid arrow (→) on the signal line indicates direction, and the tracing of the signal through the schematics is referenced as it leaves the page. The reference is shown as a number - letter number (e.g. 2-A3), indicating sheet 2 and schematic zone A3. The reference may be shown alone or in a bracket. If the reference is bracketed, the signal is going to another schematic which is referenced outside the bracket. If the reference is shown alone, the signal is going to another page of the multisheet schematic.

3-5. 8800b BLOCK DIAGRAM DESCRIPTION (Figure 3-1)

The 8800b computer contains four basic circuits; the Central Processing Unit (CPU), Memory, an Input/Output (I/O) section, and the Front Panel. The CPU controls the interpretation and execution of software instructions, and Memory stores the software information to be used by the CPU. The I/O section provides a communication link between the CPU and external devices. The Front Panel allows the operator to manually perform various operations with the 8800b. The 8800b basic block diagram and accompanying text (paragraphs 3-6 and 3-7) explain the CPU's communication with the memory (and I/O) circuits and with the front panel. The system clock, power-on operation and run operation are explained in paragraphs 3-8 through 3-10.

3-6. CPU TO MEMORY OR I/O OPERATION

The Memory or I/O section operation repuires several signals that allow transfer of data to and from the CPU. The ADDRESS bus (AØ-A15) consists of sixteen individual lines from the CPU to Memory and I/O devices. The signals on this bus represent a particular

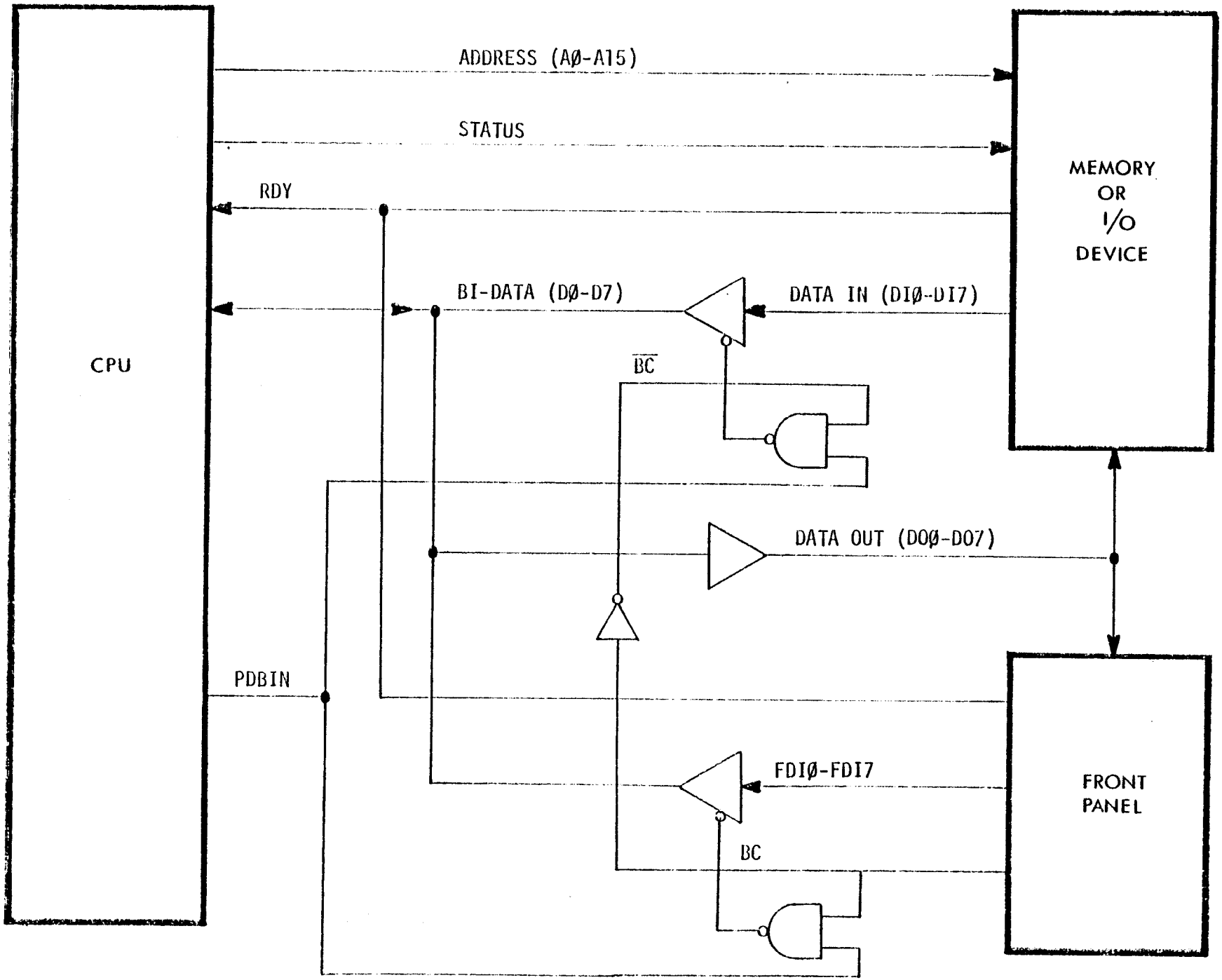


Figure 3-1. 8800b Basic Block.

memory address location or external device number that is needed to establish communications with Memory or I/O devices. Once the address data (AØ-A15) is presented to Memory or I/O devices, the CPU generates various STATUS signals. The STATUS signals enable decoding of a memory address or conditions the I/O device card to send or receive data from the CPU

Data from Memory or I/O devices is presented on the DATA IN lines (DIØ-DI7) and applied to eight non-inverting bus drivers. The drivers are enabled by a PDBIN signal from the CPU and a $\overline{BC}$ (bus control) signal. The BC signal is LOW when the Front Panel is not in operation. The eight non-inverting bus drivers, when enabled, present the input data to BI-DATA lines (DØ-D7) which input the data to the CPU.

Data outputted to Memory or I/O devices is presented to the DATA OUT lines (DOØ-DO7) from the CPU. The RDY (ready) line either forces the CPU to a wait state while data is being transferred or allows the CPU to process data.

3-7. FRONT PANEL OPERATION

The Front Panel Operation is very similar to Memory or I/O section operation. The Front Panel gains control of the CPU by producing a HIGH BC signal. The BC signal disables the DATA IN (DIØ-DI7) lines from a Memory or I/O Device and enables the FDIØ-FDI7 lines. The FDIØ-FDI7 lines contain Front Panel data which is transferred to the CPU upon the occurence of the PDBIN signal. All data from the CPU to the Front Panel is applied to the DATA OUT (DOØ-DO7) lines and displayed on the Front Panel.

3-8. SYSTEM CLOCK

The system clock (F) for the 8800b is located on the CPU circuit card (Figure 3-14, zone B7). The system clock generates phase 1 and phase 2 outputs derived from the external crystal (XTAL 1). The Ø1 and Ø2 outputs operate at a frequency of 2 MHz, which determines the speed at which the 8080 (M) will operate. The Ø1 and Ø2 clock signals are presented to the bus (zone A7) through inverter A and inverter bus driver J, respectively. The Ø1 clock is used by memory and external I/O cards, and the Ø2 clock is applied to the 24-bit counter on the Display/Control card (Figure

3-16, sheet 1, zone D2) through the Interface card (Figure 3-15, sheet 2, zone B3).

3-9. POWER ON CLEAR OPERATION

Positioning the ON/OFF switch to ON causes a power on clear (POC) operation to be performed, resetting the 8800b circuitry. The POC signal is generated on the CPU card (Figure 3-14, zone A3) when VCC is applied. With VCC present, capacitor C4 will charge to the VCC potential in 100 milliseconds because of the RC time constant of C4 and resistor R17. The 100 millisecond delay disables (turns off) transistor Q3, producing a LOW $\overline{POC}$ signal to the bus (pin 99) through inverters S and J (zone A2). The $\overline{POC}$ signal is inverted by U on the Interface card (Figure 3-15, sheet 2, zone B2) and presented to the Display/Control card as a HIGH POC signal (Figure 3-16, sheet 2, zone D6). The POC input is inverted LOW by T1 (zone C6) and applied to three circuits on the Display/Control Card. It clears the M1 flip-flops (zone C7) through NOR gate T1 and inverter J1 (zone C6), insuring that single step operation is disabled. It presets the M1 flip-flop (zone C9) and disables NAND gate P1 (zone B8) to insure that the 8800b is not running. The $\overline{POC}$ signal (zone D9) is also present at NOR gate R1 which inverts it HIGH to reset the PROM counter. The $\overline{POC}$ signal is present to the external input/output (I/O) cards and memory for similar initialization operations. During the POC operation, two other functions are being performed.

On the Display/Control card (Figure 3-16, sheet 1, zone D2), a 24-bit counter is being clocked by Ø2 which will condition circuits on the Display/Control card. The $\overline{C13}$ output (zone D1) from the counter is applied to the clock (CK) input of quad latches C1, F1, H1, G1, N1, U1, Y1, and W1 (zones B9-B1) through non-inverting bus driver K1 (zones A1 and D1) and inverter J1 (zone C1). The $\overline{C13}$ signal clears the quad latches in the following manner to insure all latches are conditioned after POC. The inputs to quad latches C1, F1, H1, and G1 are HIGH because no switches are activated. After the first $\overline{C13}$ clock, all the $\overline{Q}$ outputs are LOW and applied to the inputs of quad latches N1, U1, Y1, and W1 (zones B9-B1).

The occurrence of the next $\overline{C13}$ clock latches the Q outputs LOW and the $\overline{Q}$ outputs HIGH during the POC operation.

When VCC is present in the CPU circuits, another RC time constant affects the clock generator F (Figure 3-14, zone B7). Capacitor C2 will charge to the VCC potential in 33 microseconds which is the time constant of C2 and resistor R10. The 33 microsecond delay allows the RESET output from F (zone B7) to clear the 8080 M internal circuits. The 8080 remains in this state because the READY output (zone B7) is LOW from F. The READY output from F will be affected during the run operation.

3-10. RUN OPERATION

The Run Operation allows the 8080 on the CPU Board to start processing data to and from memory and external devices. The Run Operation is activated when the RUN/STOP switch on the 8800b front panel is momentarily depressed to RUN.

The RUN/STOP circuits are located on the Display/Control card (Figure 3-16, sheet 2, zone A9). When the RUN/STOP switch is momentarily depressed, a LOW is applied to quad latch C1, input D2. The occurrence of the next C13 clock (zone A1) causes the $\overline{Q}$ output at pin 6 of C1 (zone B9) to go HIGH. This HIGH is applied to quad latch N1, input D2. The next C13 clock causes the Q output at pin 2 of N1 (zone B9) to go HIGH and allows NAND gate P1 to clear M1 (zone C9). The Q output of M1 generates a LOW $\overline{RUN}$ signal and LOW $\overline{FRDY}$ signal through NOR gate P1 and inverter R1 (zone D9).

The $\overline{RUN}$ signal is applied to the Interface Card (Figure 3-15, sheet 2, zone D2) to condition the MD input of data latch G (sheet 3, zone A6). With MD enabled, output data from the CPU can be displayed on the 8800b front panel if a STB input is present to G (discussed in Paragraph 3-40).

The $\overline{FRDY}$ signal is applied to the Interface Card (Figure 3-15, sheet 2) to allow the 8080 to start processing data. The FRDY output is applied to pin 58 of the bus through inverter R and non-inverting bus driver H as a HIGH (zone A1). The HIGH on pin 58 of the bus enables NAND gate C, pin 8, LOW on the CPU (Figure 3-14, zone A7) which is inverted HIGH by B (zone B7) and applied

to the clock generator F RYDIN input. The RYDIN signal enables the READY output at F HIGH (zone B7) which allows the 8080 M (zone A8) to start processing data.

3-11. 8800b DATA PROCESSING OPERATION

The 8800b data processing begins when the 8080 IC is enabled (Paragraph 3-10). With the 8080 IC enabled, the program (P) counter in the 8080 starts to increment or begins at a predetermined count established by the operator. The count in the P counter represents a location in memory which is examined by the CPU before the P counter increments to the next location. To examine each memory location, the CPU initiates an instruction cycle operation. Every instruction cycle consists of one, two, three, four, or five machine cycles. In order to perform a data processing operation, basic machine cycles are required.

The Instruction Fetch Machine cycle is a basic machine cycle needed to allow the CPU to fetch an instruction from memory. A memory read machine cycle is also a basic machine cycle that enables the CPU to communicate with a memory or external device for data transfer operations.

The following paragraphs discuss data transfers from an external device to the CPU, from the CPU to memory, from memory to the CPU, and from the CPU to an external device. However, the instruction fetch and memory read machine cycles used in the data transfers are discussed first because their operation is identical in all of the data transfers. It is important to note that there are many variations of data transfer which are dependent on the programmer.

3-12. INSTRUCTION FETCH CYCLE

The Instruction Fetch Cycle is the first machine cycle (M1) to be performed by the CPU in any data transfer operation. The memory location specified by the P counter contains data that the CPU interprets as an instruction. The first cycle must be a fetch cycle because, during the fetch cycle, the CPU is informed as to what operation will be performed next.

3-13. INSTRUCTION FETCH CYCLE OPERATION (Figure 3-2)

The Instruction Fetch Cycle is initiated whenever the P counter is incremented to a new memory address location (e.g. 000 100_8) where an instruction (e.g. 072_8) is stored. In order to fetch the 072_8 data from memory during machine cycle one, several signals are generated by the CPU.

A PSYNC output from the CPU is applied to memory to condition for address decoding. Next the ADDRESS (000 100_8), consisting of sixteen parallel outputs (AØ-A15) from the CPU, is presented to the Display/Control Card and memory. The AØ through A15 signals drive the appropriate address buffers, illuminating the light emitting diodes (LEDs) on the Display/Control Card. The ADDRESS and PSYNC signals present at the memory from the CPU initiate decoding of the memory address (000 100_8).

The CPU then generates three signals, SM1, SMEMR, and Ø1 CLOCK to complete the Instruction Fetch Cycle. The SM1 output is applied to the Display/Control Card through the Interface Card to light the M1 (machine cycle 1) LED on the 8800b front panel. The SMEMR and Ø1 CLOCK outputs are applied to memory to allow decoding of the memory address (000 100_8). With the memory address decoded, the 072_8 data present in that location is transferred to the CPU on the eight DATA IN (DIØ-DI7) lines. The DIG 1 input to the CPU from the Interface Card is enabled when the 8800b is in the run mode (see paragraph 3-10). This permits the memory data to be transferred to the CPU. The SMEMR output is applied to the Display/Control Card through the Interface Card to light the MEMR (memory read) LED on the 8800b front panel. This operation is performed when the P counter is incremented, indicating a new memory address.

3-14. INSTRUCTION FETCH CYCLE DETAILED OPERATION

The following paragraphs describe the Instruction Fetch Cycle operation in detail. Refer to Figure 3-3, Instruction Fetch Cycle Timing, during the explanation. The Instruction Fetch Cycle operation (M1) requires four Ø1 and Ø2 clock pulses. Each clock period performs a particular operation as described in the following paragraphs.

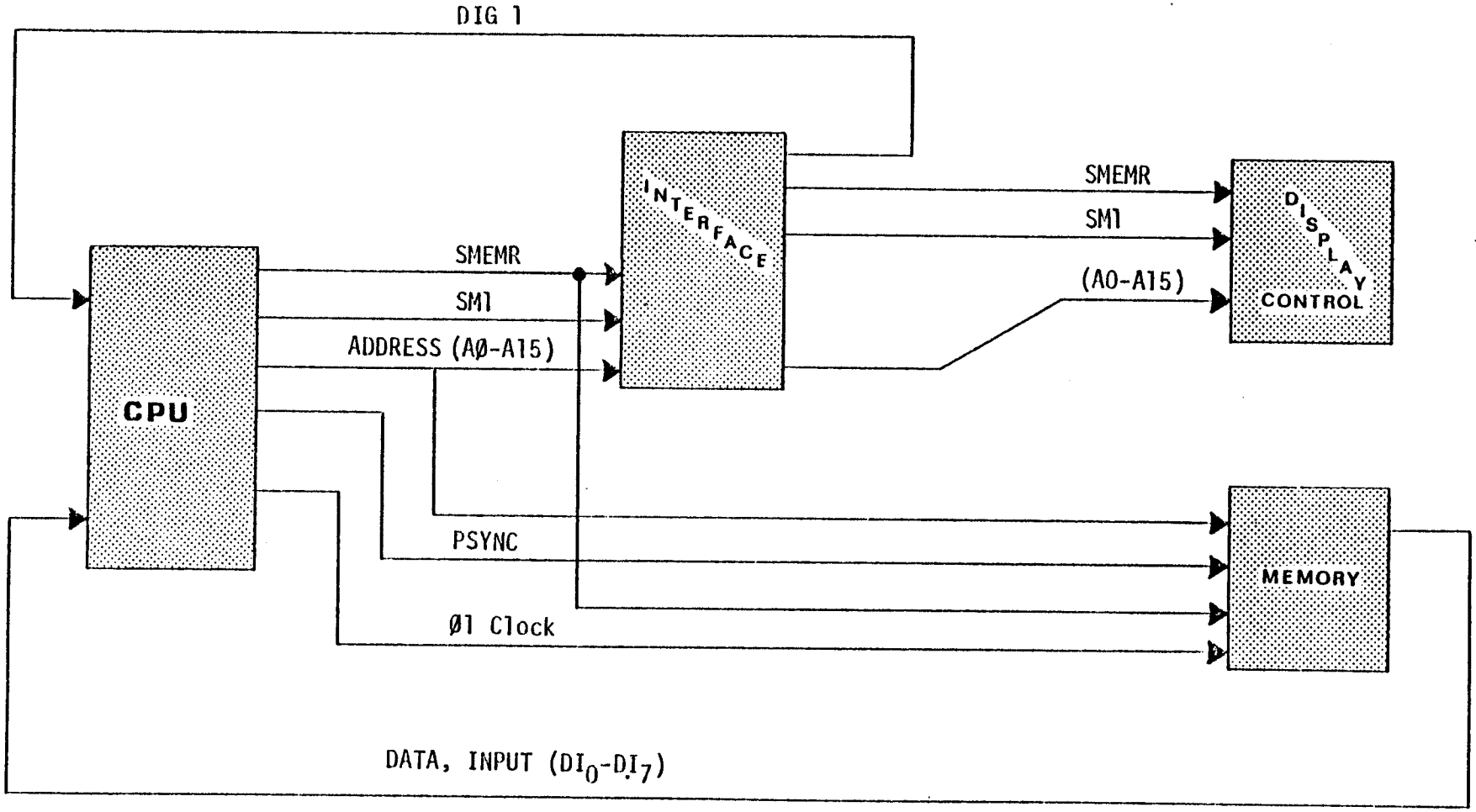


Figure 3-2. Instruction Fetch Cycle Block Digaram

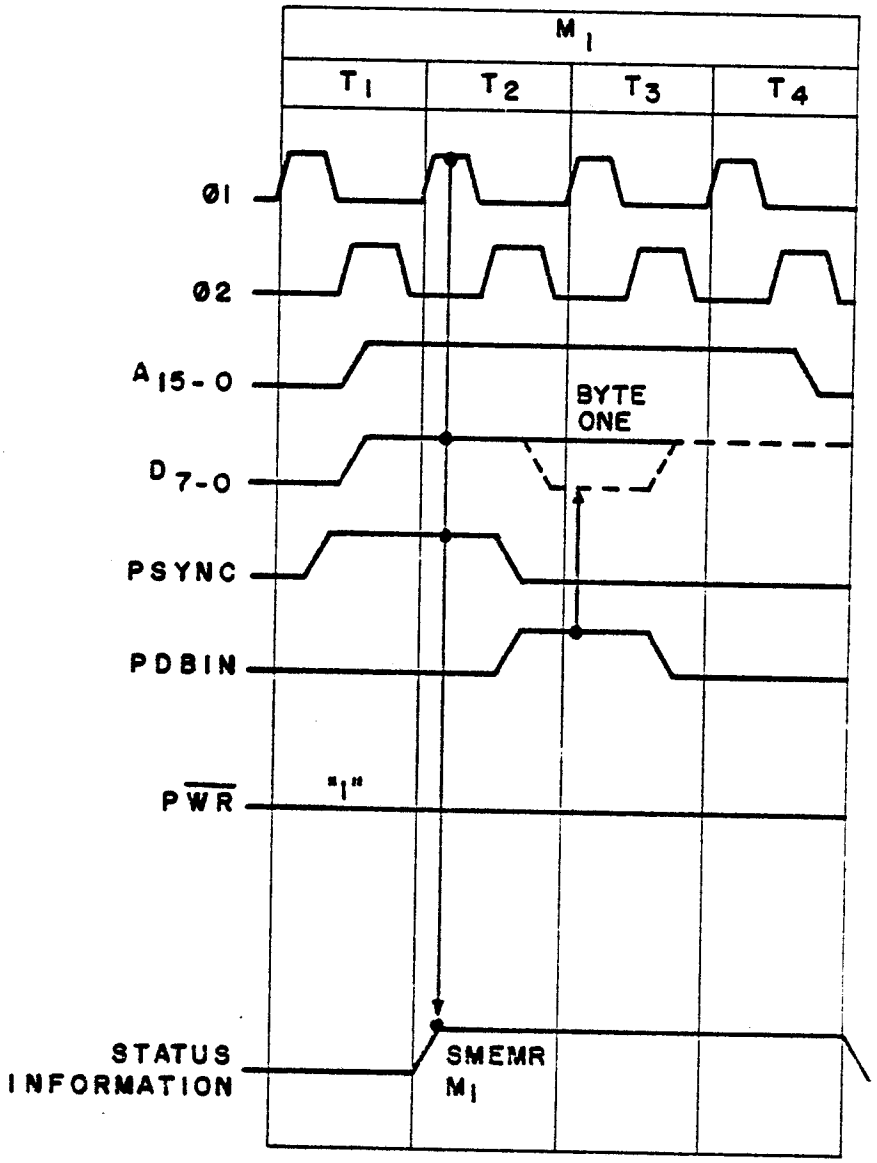


Figure 3-3. Instruction Fetch Cycle Timing.

During the latter portion of T1, several outputs are generated by the CPU (M) (Figure 3-14): address data AØ through A15 (zone B8), status data DØ through D7, and a SYNC signal (zone C8). The AØ through A15 data is applied to memory via the bus through non-inverting bus drivers, U, P, and N (zone B9) on the CPU. The address data (AØ-A15) is also applied through inverters P, N, and X on the Interface card (Figure 3-15, sheet 1, zone B5) and presented to the Display/Control card. The AØ through A15 signals present on the Display/Control card light the appropriate AØ through A15 LEDs, indicating the memory address.
The DØ through D7 data is applied to K (zone B5) on the CPU through the bi-directional circuits D and E. The status data is enabled through D and E at this time because $\overline{CS}$ and $\overline{DIEN}$ are LOW. The SYNC output is applied to the clock generator F (zone B7) and memory as PSYNC via pin 76 (zone D1) on the bus through the non-inverting bus driver V (zone D8). The PSYNC signal conditions memory to decode the address data. The SYNC input at F will enable a signal during T2.

During the beginning of T2, a low $\overline{STSTB}$ (zone B7) is generated from F as a result of the HIGH SYNC input and internal timing of F. The $\overline{STSTB}$ is applied to the data latch K (zone B5), allowing the status data DØ through D7 to be stored in K. The status data present at the output of K conditions the memory to fetch the instruction (072_8) from its addressed memory location (e.g. 000 100_8) by enabling the following signals.

A SM1 and SMEMR HIGH output from K is presented on pins 44 and 47 of the bus (zone A5) through non-inverting bus drivers X and R. The SM1 and SMEMR signals are applied through inverter V on the Interface card (Figure 3-15, sheet 2, zone B5) and presented to the Display/Control card as $\overline{SM1}$ and $\overline{SMEMR}$. The $\overline{SM1}$ and $\overline{SMEMR}$ signals present on the Display/Control card light the M1 and MEMR LEDs (Figure 3-16, sheet 3, zone C3) on the front panel of the 8800b, indicating machine cycle one is performing a memory read operation. The SMEMR output from the CPU (Figure 3-14, zone A5) is applied to memory, initiating a data transfer to the CPU during T3.

At the beginning of T3, the instruction (072_8) data is transferred from memory to M on the CPU. The memory data (DIØ through DI7) is supplied to the CPU card (Figure 3-14, zone B1) from the bus. The data is presented to M through bi-directional gates D and E (zone C7), inverter bus drivers L and J (zone B4), and inverters Y and S (zone B3) by the DBIN signal.

At the latter portion of T2 and the beginning of T3, a high DBIN output (zone C8) is generated by M. The DBIN output is applied to the $\overline{\text{DIEN}}$ inputs (zone C7) of D and E and pin 4 of NAND gate C (zone B4) as PDBIN. This signal enables pin 6 of NAND gate C LOW (DIG1 is high when the front panel is not used). This allows data input from memory (DIØ-DI7) to be enabled through inverting bus drivers L and J (zone B4) and applied through bi-directional gates D and E to M (zone C7).

Clock period T4 of machine cycle one allows for 8080 processing of the received instruction data from memory. If the instruction data present in the CPU requires a data transfer to or from an external device, a memory read cycle (M2) is initiated. However, if the instruction data present in the CPU requires a data transfer to or from memory, two memory read cycles (M2 and M3) are initiated.

3-15. MEMORY READ CYCLE

The Memory Read Cycle (M2) follows the Instruction Fetch Cycle (M1). During a Memory Read Cycle, an address is transferred to the CPU from memory. This address is either an external device number or a memory location (depending upon the instructions received during M1).

3-16. MEMORY READ CYCLE OPERATION (Figure 3-4)

The CPU performs one or two Memory Read Cycle operations. If the CPU is to communicate with an external device, one Memory Read Cycle is required because the external device number consists of 8 data

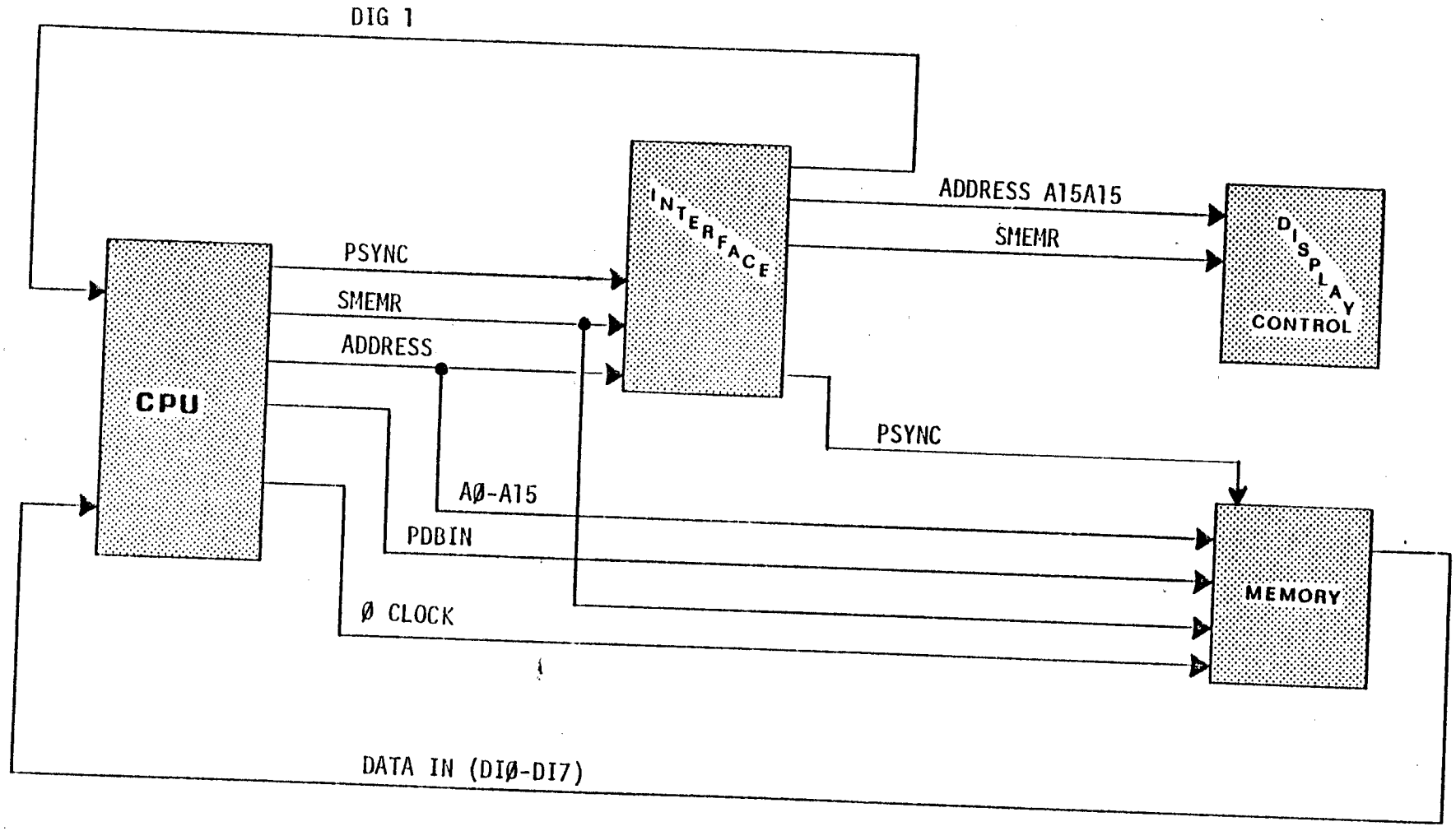


Figure 3-4. Memory Read Cycle Block Diagram

bits (1 byte). However, if the CPU is instructed to communicate with memory, two Memory Read Cycles are required because the memory address consists of 16 data bits (2 bytes).

The two Memory Read Cycles obtain the memory address (e.g. 000 200_8) that is required by the CPU to complete the instruction. Since one byte (8 bits) of the two byte address is transferred during one Memory Read Cycle, two cycles are required. The first Memory Read Cycle obtains the least significant bits (LSBs) of the address (200_8) from memory and stores them in the CPU. The second cycle obtains the most significant bits (MSBs) of the address (000_8) from memory and stores them in the CPU.

The Memory Read Cycles are very similar to the Instruction Fetch Cycle. They require a memory address location (e.g. 000 101_8 and 000 102_8) that indicates where the LSBs and MSBs of the address (000 200_8) are stored. After completion of the Instruction Fetch Cycle, the program counter in the CPU is incremented to 000 101_8 and the first Memory Read Cycle is initiated. Several signals are generated by the CPU in order to read the LSBs of the address (200_8) from memory.

A PSYNC output from the CPU is applied to memory through the Interface Card to condition the memory for address decoding. Next the ADDRESS (000 101_8), consisting of sixteen parallel outputs (AØ-A15) from the CPU, is presented to the Display/Control Card and memory. The AØ through A15 signals light the appropriate address light emitting diodes (LEDs) on the Display/Control Card. The ADDRESS and PSYNC signals present at the memory from the CPU initiate decoding of the address (000 101_8).

The CPU then generates three signals, SMEMR, PDBIN, and Ø1 to complete the Memory Read Cycle. The SMEMR, PDBIN, and Ø1 outputs are presented to memory to enable decoding of the address (000 101_8). With the address decoded, the 200_8 data present in that location is transferred to the CPU on the eight DATA IN (DIØ-DI7) lines. The DIG1 input to the CPU from the Interface Card is enabled when the 8800b is in the run mode, permitting memory data to be transferred to the CPU.

The SMEMR output is presented to the Display/Control Card through the Interface Card to light the MEMR (memory read) LED on the 8800b front panel. The second Memory Read Cycle operation is identical to the first. It transfers the MSBs of the address (000_8) to the CPU.

3-17. MEMORY READ CYCLE DETAILED OPERATION

The following paragraphs describe the Memory Read Cycle operation in detail. Refer to Figure 3-5, Memory Read Cycle Timing, during the explanation.

The two Memory Read Cycle operations (M2 and M3) obtain the memory address (e.g. 000 200_8) required by the CPU to complete an instruction. As stated previously, the LSBs of the address (200_8) are transferred to the CPU during M2, and the MSBs of the address (000_8) are transferred to the CPU during M3. There are three clock periods (T1-T3) required for each Memory Read Cycle operation.

During the latter portion of T1, several outputs are generated by the CPU (Figure 3-14); Address data AØ through A15 (zone B8), status data DØ through D7, and a SYNC signal (zone C8). The AØ through A15 data is presented to memory and the 8800b front panel via the bus through non-inverting bus drivers U, P, and N (zone B9) on the CPU. The DØ through D7 data is applied to K (zone B5) on the CPU through the bi-directional circuits D and E. The status data is enabled through D and E at this time because $\overline{CS}$ and $\overline{DIEN}$ are LOW. The SYNC output is applied to the clock generator F (zone B7) and memory as PSYNC via pin 76 (zone D1) on the bus through non-inverting bus driver V (zone D8). The PSYNC signal conditions memory to decode the address data.

During the beginning of T2, a $\overline{STSTB}$ (zone B7) is generated (LOW) from F as a result of the HIGH SYNC input and internal timing of F. The $\overline{STSTB}$ is applied to the data latch K (zone B5), allowing the status data DØ through D7 to be stored in K. The status data present at the output of K allows the CPU to read the LSBs of the memory address

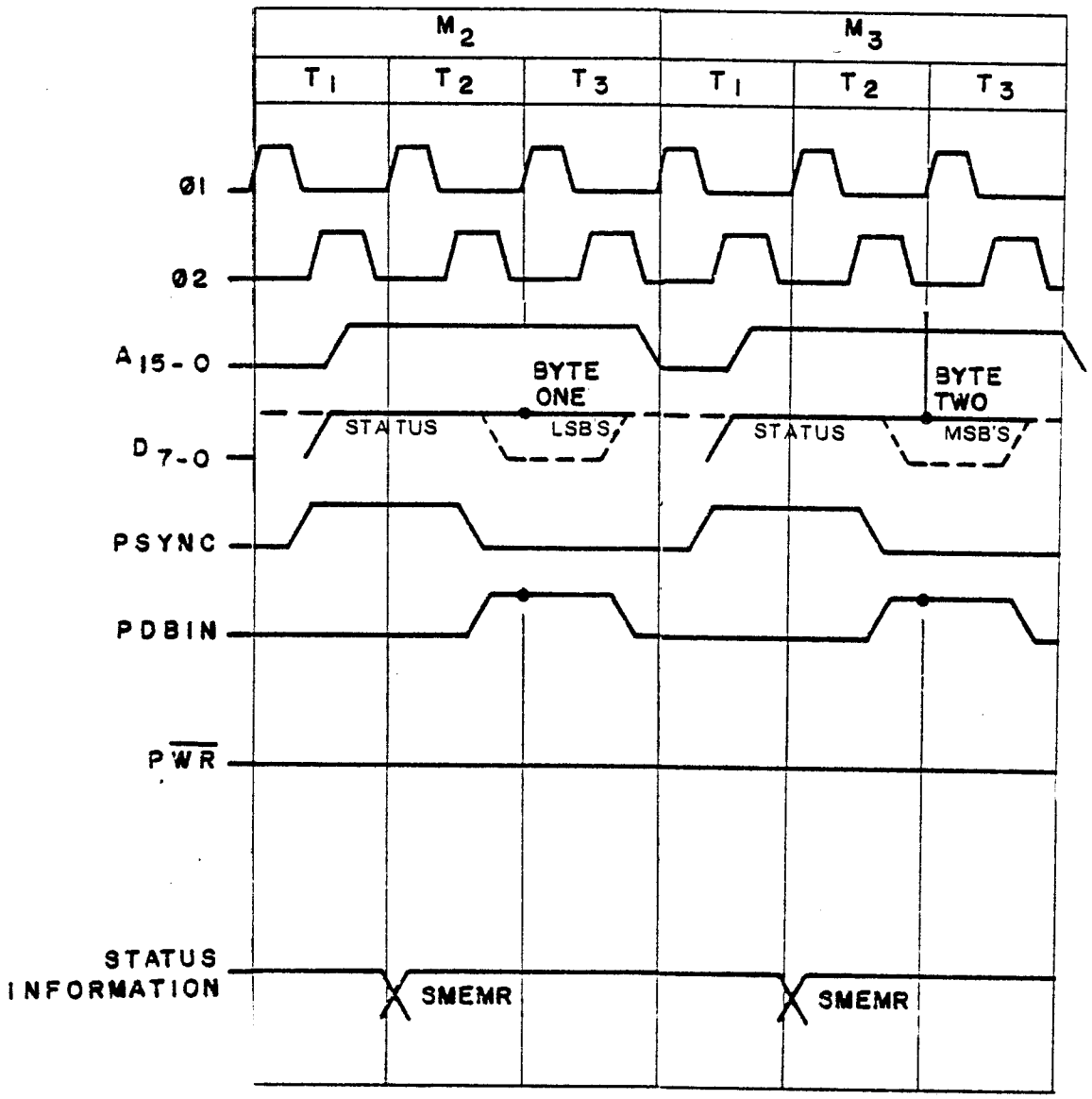


Figure 3-5. Memory Read Cycle Timing.

location (ex. 000 101_8) by enabling the SMEMR signal.

A SMEMR output (HIGH) from K is presented on pin 47 of the bus (zone A4) through non-inverting bus drivers X and R. The SMEMR signal is applied through inverter V on the Interface Card (Figure 3-15, sheet 2, zone B4) and presented to the Display/Control card as $\overline{\text{SMEMR}}$. The $\overline{\text{SMEMR}}$ signal present on the Display/Control card lights the MEMR LED (Figure 3-16, zone C3) on the front panel of the 8800b, indicating a memory read operation is occurring. The SMEMR output from the CPU (Figure 3-14, zone A5) is applied to memory in order to initiate a data transfer to the CPU during T3.

At the beginning of T3, the LSBs of the memory storage location (200_8) are transferred from memory to the 8080 (M) on the CPU. The memory data in (DIØ through DI7) is applied to the CPU card (Figure 3-14, zone B1) from the bus. The data is presented to M through bi-directional gates D and E (zone C7), inverter bus drivers L and J (zone B4), and inverters Y and S (zone B3) by the PDBIN signal.

At the latter portion of T2 and the beginning of T3, a DBIN output (zone C8) HIGH is generated by M. The DBIN output is applied to the $\overline{\text{DIEN}}$ inputs (zone C7) of D and E and pin 4 of NAND gate C (zone B4) as PDBIN. This signal enables pin 6 of NAND gate C LOW (DIG 1 is high when front panel is not used). This allows the data in from memory (DIØ - DI7) to be enabled through inverting bus drivers L and J (zone B4) and applied through bi-directional gates D and E to M (zone C7). The second Memory Read Cycle operation (M3) transfers the contents of memory address (000 102_8) which contain the MSBs of the memory address number to the CPU. It is important to note that only one Memory Read Cycle operation is required if the CPU is to communicate with an external device.

3-18. EXTERNAL DEVICE TO CPU DATA TRANSFER

An External Device to CPU data transfer is accomplished when an input instruction (333_8) is fetched from a memory location during M1, and the external device number (XXX_8) is read from a memory location during M2 by the CPU. The data from the external device is transferred to the CPU by an Input Read Cycle operation (M3).

3-19. INPUT READ CYCLE OPERATION (Figure 3-6)

The Input Read Cycle operation will allow the CPU to obtain data from an external device. After the completion of the Memory Read Cycle (M2), the program counter is not incremented until the completion of the Input Read Cycle. Several signals are generated by the CPU in order to obtain data from the external device.

The SINP output and external device ADDRESS (XXX_8) number, consisting of the first eight individual outputs (AØ-A7) from the CPU, is presented to the external device input/output channel, thereby enabling the I/O card. With the I/O enabled, a PDBIN signal from the CPU allows the I/O to transfer the external device data to the CPU on the eight DATA IN (DIØ-DI7) lines for storage. The DIG 1 input to the CPU from the Interface is enabled during the 8800b run mode and allows the external device data to be stored in the CPU. The SINP and AØ through A15 outputs are supplied to the Display/Control Card through the Interface Card to illuminate the INP (input) and ADDRESS LEDs on the 8800b front panel.

3-20. INPUT READ CYCLE DETAILED OPERATION

The following paragraphs describe the Input Read Cycle operation in detail. Refer to Figure 3-7, Input Read Cycle Timing, during the explanation. The Input Read Cycle operation (M3) requires three Ø1 and Ø2 clock pulses. During each clock period, a specific operation is performed as described in the following paragraphs.

During the latter portion of T1, several outputs are generated by the CPU (Figure 3-14); address data AØ through A15 (zone B8), status data DØ through D7, and a SYNC signal (zone C8). The AØ through A15 data contains the external device number (AØ-A7 and A8-A15 contain identical data) and is applied to the I/O card via the bus through non-inverting bus drivers U, P, and N (zone B9) on the CPU in order to enable the I/O card. The address data (AØ-A15) is also applied through inverters P, W, and X on the Interface Card (Figure 3-15, sheet 1, zone B5) and