The Engineering Staff of
TEXAS INSTRUMENTS INCORPORATED
Semiconductor Group

TMS 8080 Microprocessor

Second Edition

FEBRUARY 1976

TEXAS INSTRUMENTS
INCORPORATED

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Information contained in this publication is believed to be accurate and reliable. However, responsibility is assumed neither for its use nor for any infringement of patents or rights of others that may result from its use. No license is granted by implication or otherwise under any patent or patent right of Texas Instruments or others.

TMS 8080 MICROPROCESSOR

1. ARCHITECTURE

1.1 INTRODUCTION

The TMS 8080 is an 8-bit parallel central processing unit (CPU) fabricated on a single chip using a high-speed N-channel silicon-gate process. (See Figure 1). A complete microcomputer system with a 2-μs instruction cycle can be formed by interfacing this circuit with any appropriate memory. Separate 8-bit data and 16-bit address buses simplify the interface and allow direct addressing of 65,536 bytes of memory. Up to 256 input and 256 output ports are also provided with direct addressing. Control signals are brought directly out of the processor and all signals, excluding clocks, are TTL compatible.

1.2 THE STACK

The TMS 8080 incorporates a stack architecture in which a portion of external memory is used as a pushdown stack for storing data from working registers and internal machine status. A 16-bit stack pointer (SP) is provided to facilitate stack location in the memory and to allow almost unlimited interrupt handling capability. The CALL and RST (restart) instructions use the SP to store the program counter (PC) into the stack. The RET (return) instruction uses the SP to acquire the previous PC value. Additional instructions allow data from registers and flags to be saved in the stack.

1.3 REGISTERS

The TMS 8080 has three categories of registers: general registers, program control registers, and internal registers. The general registers and program control registers are listed in Table 1. The internal registers are not accessible by the programmer. They include the instruction register, which holds the present instruction, and several temporary storage registers to hold internal data or latch input and output addresses and data.

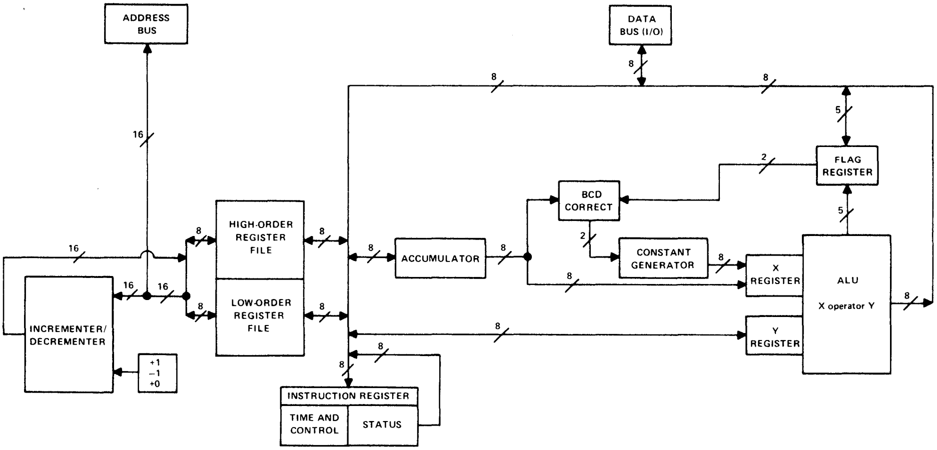


FIGURE 1—TMS 8080 FUNCTIONAL BLOCK DIAGRAM

1.4 THE ARITHMETIC UNIT

Arithmetic operations are performed in an 8-bit parallel arithmetic unit that has both binary and decimal capabilities. Four testable internal flag bits are provided to facilitate program control, and a fifth flag is used for decimal corrections. Table 2 defines these flags and their operation. Decimal corrections are performed with the DAA instruction. The DAA corrects the result of binary arithmetic operation on BCD data as shown in Table 3.

1.5 STATUS AND CONTROL

Two types of status are provided by the TMS8080. Certain status is indicated by dedicated control lines. Additional status is transmitted on the data bus during the beginning of each instruction cycle (machine cycle). Table 4 indicates the pin functions of the TMS8080. Table 5 defines the status information that is presented during the beginning of each machine cycle (SYNC time) on the data bus.

1.6 I/O OPERATIONS

Input/output operations (I/O) are performed using the IN and OUT instructions. The second byte of these instructions indicates the device address (256 device addresses). When an IN instruction is executed, the input device address appears in duplicate on A7 through A0 and A15 through A8, along with $\overline{\text{WO}}$ and INP status on the data bus. The addressed input device then puts its input data on the data bus for entry into the accumulator. When an OUT instruction is executed, the same operation occurs except that the data bus has OUT status and then has output data.

Direct memory access channels (DMA) can be OR-tied directly with the data and address buses through the use of the HOLD and HLDA (hold acknowledge) controls. When a HOLD request is accepted by the CPU, HLDA goes high, the address and data lines are forced to a high-impedance or "floating" condition, and the CPU stops until the HOLD request is removed.

Interfacing with different speed memories is easily accomplished by use of the WAIT and READY pins. During each machine cycle, the CPU polls the READY input and enters a wait condition until the READY line becomes true. When the WAIT output pin is high, it indicates that the CPU has entered the wait state.

Designing interrupt driven systems is simplified through the use of vectored interrupts. At the end of each instruction, the CPU polls the INT input to determine if an interrupt request is being made. This action does not occur if the CPU is in the HOLD state or if interrupts are disabled. The INTE output indicates if the interrupt logic is enabled (INTE is high). When a request is honored, the INTA status bit becomes high, and an RST instruction may be inserted to force the CPU to jump to one of eight possible locations. Enabling or disabling interrupts is controlled by special instructions (EI or DI). The interrupt input is automatically disabled when an interrupt request is accepted or when a RESET signal is received.

1.7 INSTRUCTION TIMING

The execution time of the instructions varies depending on the operation required and the number of memory references needed. A machine cycle is defined to be a memory referencing operation and is either 3, 4, or 5 state times long. A state time (designated S) is a full cycle of clocks $\phi 1$ and $\phi 2$. (NOTE: The exception to this rule is the DAD instruction, which consists of 1 memory reference in 10 state times). The first machine cycle (designated M1) is either 4 or 5 state times long and is the "instruction fetch" cycle with the program counter appearing on the address bus. The CPU then continues with as many M cycles as necessary to complete the execution of the instruction (up to a maximum of 5). Thus the instruction execution time varies from 4 state times (several including ADDr) to 18 (XTHL). The WAIT or HOLD conditions may affect the execution time since they can be used to control the machine (for example to "single step") and the HALT instruction forces the CPU to stop until an interrupt is received. As the instruction execution is completed (or in the HALT state) the INT pin is polled for an interrupt. In the event of an interrupt, the PC will not be incremented during the next M1 and an RST instruction can be inserted.

TABLE 1
TMS 8080 REGISTERS

NAME	DESIGNATOR	LENGTH	PURPOSE
Accumulator	A	8	Used for arithmetic, logical, and I/O operations
B Register	B	8	General or most significant 8 bits of double register BC
C Register	C	8	General or least significant 8 bits of double register BC
D Register	D	8	General or most significant 8 bits of double register DE
E Register	E	8	General or least significant 8 bits of double register DE
H Register	H	8	General or most significant 8 bits of double register HL
L Register	L	8	General or least significant 8 bits of double register HL
Program Counter	PC	16	Contains address of next byte to be fetched
Stack Pointer	SP	16	Contains address of the last byte of data saved in the memory stack
Flag Register	F	5	Five flags (C, Z, S, P, C1)

NOTE: Registers B and C may be used together as a single 16-bit register, likewise, D and E, and H and L.

TABLE 2
FLAG DESCRIPTIONS

SYMBOL	TESTABLE	DESCRIPTION
C	YES	C is the carry/borrow out of the MSB (most significant bit) of the ALU (Arithment Logic Unit). A TRUE condition (C = 1) indicates overflow for addition or underflow for subtraction.
Z	YES	A TRUE condition (Z = 1) indicates that the output of the ALU is equal to zero.
S	YES	A TRUE condition (S = 1) indicates that the MSB of the ALU output is equal to a one (1).
P	YES	A TRUE condition (P = 1) indicates that the output of the ALU has even parity (the number of bits equal to one is even).
C1	NO	C1 is the carry out of the fourth bit of the ALU (TRUE condition). C1 is used only for BCD correction with the DAA instruction.

TABLE 3
FUNCTION OF THE DAA INSTRUCTION

Assume the accumulator (A) contains two BCD digits, X and Y

	7 4	3 0
ACC	X	Y

ACCUMULATOR BEFORE DAA				ACCUMULATOR AFTER DAA			
C	$A_7 \ldots A_4$	C1	$A_3 \ldots A_0$	C	$A_7 \ldots A_4$	C1	$A_3 \ldots A_0$
0	$X < 10$	0	$Y < 10$	0	X	0	Y
0	$X < 10$	1	$Y < 10$	0	X	0	$Y + 6$
0	$X < 9$	0	$Y \geqslant 10$	0	$X + 1$	1	$Y + 6$
1	$X < 10$	0	$Y < 10$	1	$X + 6$	0	Y
1	$X < 10$	1	$Y < 10$	1	$X + 6$	0	$Y + 6$
1	$X < 10$	0	$Y \geqslant 10$	1	$X + 7$	1	$Y + 6$
0	$X \geqslant 10$	0	$Y < 10$	1	$X + 6$	0	Y
0	$X \geqslant 10$	1	$Y < 10$	1	$X + 6$	0	$Y + 6$
0	$X \geqslant 9$	0	$Y \geqslant 10$	1	$X + 7$	1	$Y + 6$

NOTE: The corrections shown in Table 3 are sufficient for addition. For subtraction, the programmer must account for the borrow condition that can occur and give erroneous results. The most straight forward method is to set A = 99_{16} and carry = 1. Then add the minuend to A after subtracting the subtrahend from A.

TABLE 4
TMS 8080 PIN DEFINITIONS

SIGNATURE	PIN	I/O	DESCRIPTION
A15 (MSB)	36	OUT	A15 through A0 comprise the address bus. True memory or I/O device addresses appear on this 3-state bus during the first state time of each instruction cycle.
A14	39	OUT	
A13	38	OUT	
A12	37	OUT	
A11	40	OUT	
A10	1	OUT	
A9	35	OUT	
A8	34	OUT	
A7	33	OUT	
A6	32	OUT	
A5	31	OUT	
A4	30	OUT	
A3	29	OUT	
A2	27	OUT	
A1	26	OUT	
A0 (LSB)	25	OUT	
D7 (MSB)	6	IN/OUT	D7 through D0 comprise the bidirectional 3-state data bus. Memory, status, or I/O data is transferred on this bus.
D6	5	IN/OUT	
D5	4	IN/OUT	
D4	3	IN/OUT	
D3	7	IN/OUT	
D2	8	IN/OUT	
D1	9	IN/OUT	
D0 (LSB)	10	IN/OUT	
V_{SS}	2		Ground reference
V_{BB}	11		Supply voltage (−5 V nominal)
V_{CC}	20		Supply voltage (5 V nominal)
V_{DD}	28		Supply voltage (12 V nominal)
$\phi1$	22	IN	Phase 1 clock.
$\phi2$	15	IN	Phase 2 clock. See page 19 for $\phi1$ and $\phi2$ timing.
RESET	12	IN	Reset. When active (high) for a minimum of 3 clock cycles, the RESET input causes the TMS 8080 to be reset. PC is cleared, interrupts are disabled, and after RESET, instruction execution starts at memory location 0. To prevent a lockup condition, a HALT instruction must not be used in location 0.
HOLD	13	IN	Hold signal. When active (high) HOLD causes the TMS 8080 to enter a hold state and float (put the 3-state address and data bus in a high-impedance state). The chip acknowledges entering the hold state with the HLDA signal and will not accept interrupts until it leaves the hold state.
INT	14	IN	Interrupt request. When active (high) INT indicates to the TMS8080 that an interrupt is being requested. The TMS8080 polls INT during a HALT or at the end of an instruction. The request will be accepted except when INTE is low or the CPU is in the HOLD condition.
INTE	16	OUT	Interrupts enabled. INTE indicates that an interrupt will be accepted by the TMS 8080 unless it is in the hold state. INTE is set to a high logic level by the EI (Enable Interrupt) instruction and reset to a low logic level by the DI (Disable Interrupt) instruction. INTE is also reset when an interrupt is accepted and by a high on RESET.
DBIN	17	OUT	Data bus in. DBIN indicates whether the data bus is in an input or an output mode. (high = input, low = output).

TABLE 4 (CONTINUED)

SIGNATURE	PIN	I/O	DESCRIPTION
$\overline{WR}$	18	OUT	Write. When active (low) $\overline{WR}$ indicates a write operation on the data bus to memory or to an I/O port.
SYNC	19	OUT	Synchronizing control line. When active (high) SYNC indicates the beginning of each machine cycle of the TMS8080. Status information is also present on the data bus during SYNC for external latches.
HLDA	21	OUT	Hold acknowledge. When active (high) HLDA indicates that the TMS8080 is in a hold state.
READY	23	IN	Ready control line. An active (high) level indicates to the TMS 8080 that an external device has completed the transfer of data to or from the data bus. READY is used in conjunction with WAIT for different memory speeds.
WAIT	24	OUT	Wait status. When active (high) WAIT indicates that the TMS8080 has entered a wait state pending a READY signal from memory.

TABLE 5

TMS 8080 STATUS

SIGNATURE	DATA BUS BIT	DESCRIPTION
INTA	D0	Interrupt acknowledge.
$\overline{WO}$	D1	Indicates that current machine cycle will be a read (input) (high = read) or a write (output) (low = write) operation.
STACK	D2	Indicates that address is stack address from the SP.
HLTA	D3	HALT instruction acknowledge.
OUT	D4	Indicates that the address bus has an output device address and the data bus has output data.
M1	D5	Indicates instruction acquisition for first byte.
INP	D6	Indicates address bus has address of input device.
MEMR	D7	Indicates that data bus will be used for memory read data.

2. TMS 8080 INSTRUCTION SET

2.1 INSTRUCTION FORMATS

TMS 8080 instructions are either one, two, or three bytes long and are stored as binary integers in successive memory locations in the format shown below.

One-Byte Instructions

D7 D6 D5 D4 D3 D2 D1 D0	OP CODE

Two-Byte Instructions

D7 D6 D5 D4 D3 D2 D1 D0	OP CODE
D7 D8 D5 D4 D3 D2 D1 D0	OPERAND

Three-Byte Instructions

D7 D6 D5 D4 D3 D2 D1 D0	OP CODE
D7 D6 D5 D4 D3 D2 D1 D0	LOW ADDRESS OR OPERAND 1
D7 D6 D5 D4 D3 D2 D1 D0	HIGH ADDRESS OR OPERAND 2

2.2 INSTRUCTION SET DESCRIPTION

Operations resulting from the execution of TMS 8080 instructions are described in this section. The flags that are affected by each instruction are given after the description.

2.2.1 INSTRUCTION SYMBOLS

SYMBOL	DESCRIPTION	
$<b_2>$	Second byte of instruction	
$<b_3>$	Third byte of instruction	
r_a	Register #	Register Name
	000	B
	001	C
	010	D
	011	E
	100	H
	101	L
	111	A
r_b	Register #	Register Name
	00	BC
	01	DE
	10	HL
	11	SP
r_c	Register #	Register Name
	0	BC
	1	DE
r_d	Register #	Register Name
	00	BC
	01	DE
	10	HL
r_dL	Least significant 8 bits of r_d	
r_dH	Most significant 8 bits of r_d	
f	Flags	True condition
	Zero (Z)	Result is zero
	Carry (C)	Carry/borrow out of MSB is one
	Parity (P)	Parity of result is even
	Sign (S)	MSB of result is one
	Carry 1(C1)	Carry out of fourth bit is one
M	Memory address defined by registers H and L	
( )	Contents of specified address or register	
[ ]	Contents at address contained in specified register	
←	Is transferred to	
↔	Exchange	
Am	Bit m of A register (accumulator)	
{ }	Flags affected	
b_2	Single byte immediate operand	
b_3b_2	Double byte immediate operand	
$(nnn)_8$	(nnn) is an octal (base 8) number	

2.2.2 ACCUMULATOR GROUP INSTRUCTIONS

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
ACI	b_2	2	2/7	(A) ← (A) + <b_2>+(carry), add the second byte of the instruction and the contents of the carry flag to register A and place in A. {C,Z,S,P,C1}
ADC	M	1	2/7	(A) ← (A) + (M) + (carry). {C,Z,S,P,C1}
ADC	r_a	1	1/4	(A) ← (A) + (r_a) + (carry). {C,Z,S,P,C1}
ADD	M	1	2/7	(A) ← (A) + (M), add the contents of M to register A and place in A. {C,Z,S,P,C1}
ADD	r_a	1	1/4	(A) ← (A) + (r_a). {C,Z,S,P,C1}
ADI	b_2	2	2/7	(A) ← (A) + <b_2>. {C,Z,S,P,C1}
ANA	M	1	2/7	(A) ← (A) AND (M), take the logical AND of M and register A and place in A. The carry flag will be reset low. {C,Z,S,P,C1}
ANA	r_a	1	1/4	(A) ← (A) AND (r_a). {C,Z,S,P,C1}
ANI	b_2	2	2/7	(A) ← (A) AND <b_2>. {C,Z,S,P,C1}
CMA		1	1/4	(A) ← ($\overline{A}$), complement A.
CMC		1	1/4	(carry) ← $\overline{\text{(carry)}}$, complement the carry flag. {C}
CMP	M	1	2/7	(A) – (M), compare the contents of M to register A and set the flags accordingly. {C,Z,S,P,C1} (A) = (M) Z = 1 (A) ≠ (M) Z = 0 (A) < (M) C = 1 (A) ≥ (M) C = 0
CMP	r_a	1	1/4	(A) – (r_a). {C,Z,S,P,C1}
CPI	b_2	2	2/7	(A)–<b_2>. {C,Z,S,P,C1}
DAA		1	1/4	(A)←BCD correction of (A). The 8 bit A contents is corrected to form two 4 bit BCD digits after a binary arithmetic operation. A fifth flag C1 indicates the overflow from A_3. The carry flag C indicates the overflow from A_7 (See Table 3). {C,Z,S,P,C1}
DAD	r_b	1	1/10	(HL) ← (HL) + (r_b), add the contents of double register r_b to double register HL and place in HL. {C}
LDA	b_3b_2	3	4/13	(A)←[<b_3> <b_2>]
LDAX	r_c	1	2/7	(A)←[(r_c)]
ORA	M	1	2/7	(A) ← (A) OR (M), take the logical OR of the contents of M and register A and place in A. The carry flag will be reset. {C,Z,S,P,C1}
ORA	r_a	1	1/4	(A) ← (A) OR (r_a).{C,Z,S,P,C1}
ORI	b_2	2	2/7	(A) ← (A) OR <b_2>. {C,Z,S,P,C1}
RAL		1	1/4	A_{m+1}←A_m, A_0←(carry), (carry)←(A_7). Shift the contents of register A to the left one bit through the carry flag. {C}
RAR		1	1/4	A_m←A_m+1, A_7←(carry), (carry)←A_0. {C}
RLC		1	1/4	A_{m+1}←A_m, A_0←A_7 (carry)←(A_7). Shift the contents of register A to the left one bit. Shift A_7 into A_0 and into the carry flag. {C}
RRC		1	1/4	A_m←A_{m+1}, A_7←A_0, (carry)←(A_0). {C}

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
SBB	M	1	2/7	(A)←(A)−(M)−(carry), subtract the contents of M and the contents of the carry flag from register A and place in A. Two's complement subtraction is used and a true borrow causes the carry flag to be set (underflow condition). {C,Z,S,P,C1}
SBB	r_a	1	1/4	(A)←(A)−(r_a)−(carry). {C,Z,S,P,C1}
SBI	b_2	2	2/7	(A)←(A)−<b_2>−(carry). {C,Z,S,P,C1}
STA	b_3b_2	3	4/13	[<b_3> <b_2>]←(A), store contents of A in memory address given in bytes 2 and 3.
STAX	r_c	1	2/7	[(r_c)]←(A), store contents of A in memory address given in BC or DE.
STC		1	1/4	(carry)←1, set carry flag to a 1 (true condition).
SUB	M	1	2/7	(A)←(A)−(M), subtract the contents of M from register A and place in A. Two's complement subtraction is used and a true borrow causes the carry flag to be set (underflow condition). {C,Z,S,P,C1}
SUB	r_a	1	1/4	(A)←(A)−(r_a).{C,Z,S,P,C1}
SUI	b_2	2	2/7	(A)←(A)−<b_2>. {C,Z,S,P,C1}
XRA	M	1	2/7	(A)←(A) XOR (M), take the exclusive OR of the contents of M and register A and place in A. The carry flag will be reset. {C,Z,S,P,C1}
XRA	r_a	1	1/4	(A)←(A) XOR (r_a). {C,Z,S,P,C1}
XRI	b_2	2	2/7	(A)←(A) XOR <b_2>. {C,Z,S,P,C1}

2.2.3 INPUT/OUTPUT INSTRUCTIONS

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
IN	b_2	2	3/10	(A)←(input data from data bus), byte 2 is sent on bits A7-A0 and A15-A8 as the input device address. INP status is given on the data bus.
OUT	b_2	2	3/10	(Output data)←(A), byte 2 is sent on bits A7-A0 and A15-A8 as the output device address. OUT status is given on the data bus.

2.2.4 MACHINE INSTRUCTIONS

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
HLT		1	2/7	Halt, all machine operations stop. All registers are maintained. Only an interrupt can return the TMS 8080 to the run mode. Note that a HLT should not be placed in location zero, otherwise after the reset pin is active, the TMS 8080 will enter a nonrecoverable state (until power is removed), i.e., in halt with interrupts disabled. This condition also occurs if a HLT is executed while interrupts are disabled. HLTA status is given on the data bus.
NOP		1	1/4	(PC)←(PC)+1, no operation.

2.2.5 PROGRAM COUNTER AND STACK CONTROL INSTRUCTIONS

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
CALL	b_3b_2	3	5/17	[(SP)−1] [(SP)−2]←(PC), (SP)←(SP)−2, (PC)←<b_3> <b_2>, transfer PC to the stack address given by SP, decrement SP twice, and jump unconditionally to address given in bytes 2 and 3.
Conditional call instructions for true flags:				
(f)			5/17 (Pass)	If (f) = 1, [(SP)−1] [(SP)−2]←(PC), (SP)←(SP)−2, (PS)←<b_3> <b_2>, otherwise (PC)←(PC)+3. If the flag specified, f, is 1, then execute a call. Otherwise, execute the next instruction.
CC (carry)	b_3b_2	3	3/11 (Fail)	
CPE (parity)	b_3b_2	3		
CM (sign)	b_3b_2	3		
CZ (zero)	b_3b_2	3		
Conditional call instructions for false flags:				
(f)			5/17 (Pass)	If (f) = 0, [(SP)−1] [(SP)−2]←(PC), (SP)←(SP)−2, (PC)←<b_3> <b_2>, otherwise (PC)←(PC)+3.
CNC (carry)	b_3b_2	3	3/11 (Fail)	
CPO (parity)	b_3b_2	3		
CP (sign)	b_3b_2	3		
CNZ (zero)	b_3b_2	3		
DI		1	1/4	Disable interrupts. INTE is driven false to indicate that no interrupts will be accepted.
EI		1	1/4	Enable interrupts. INTE is driven true to indicate that an interrupt will be accepted. Execution of this instruction is delayed to allow the next instruction to be executed before the INT input is polled.
JMP	b_3b_2	3	3/10	(PC)←<b_3> <b_2>, jump unconditionally to address given in bytes 2 and 3.
Conditional jump instructions for true flags:				
(f)			3/10	If (f) = 1, (PC)←<b_3><b_2>, otherwise (PC)←(PC)+3. If the flag specified, f, is 1, execute a JMP. Otherwise, execute the next instruction.
JC (carry)	b_3b_2	3		
JPE (parity)	b_3b_2	3		
JM (sign)	b_3b_2	3		
JZ (zero)	b_3b_2	3		
Conditional jump instructions for false flaqs:				
(f)			3/10	If (f) = 0, (PC)←<b_3> <b_2>, othewise (PC)←(PC)+3.
JNC (carry)	b_3b_2	3		
JPO (parity)	b_3b_2	3		
JP (sign)	b_3b_2	3		
JNZ (zero)	b_3b_2	3		
PCHL		1	1/5	(PC)←(HL)
POP	PSW	1	3/10	(F)←[(SP)], (A)←[(SP)+1], (SP)←(SP)+2, restore the last stack values addressed by SP into A and F. Increment SP twice.
POP	r_d	1	3/10	(r_{dL})←[(SP)], (r_{dH})←[(SP)+1], (SP)←(SP)+2.
PUSH	PSW	1	3/11	[(SP)−1]←(A), [(SP)−2]←(F), (SP)←(SP)−2, save the contents of A and F into the stack addressed by SP. Decrement SP twice.
PUSH	r_d	1	3/11	[(SP)−1]←(r_{dL}), [(SP)−2]←(r_{dH}), (SP)←(SP)−2.
RET		1	3/10	(PC)←[(SP)] [(SP)+1], (SP)←(SP)+2, return to program at memory address given by last values in the stack. The SP is incremented by two.

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
Conditional return instructions for true flags:				
(f)			3/11 (Pass)	If (f) = 1, (PC)←[(SP)] [(SP+1)], (SP)←(SP)+2. If the flag specified, f, is 1, execute a RET. Otherwise, execute the next instruction.
RC (carry)	C	1	1/5 (Fail)	
RPE (parity)	P	1		
RM (sign)	S	1		
RZ (zero)	Z	1		
Conditional return instructions for false flags:				
(f)			3/11 (Pass)	If (f) = 0, (PC)←[(SP)] [(SP)+1], (SP)←(SP)+2.
RNC (carry)	C	1	1/5 (Fail)	
RPO (parity)	P	1		
RP (sign)	S	1		
RNZ (zero)	Z	1		
RST		1	3/11	[(SP)−1] [(SP)−2] ←(PC) (SP)←(SP)−2, (PC)←$0000R0_8$ where R is a 3 bit field in RST (RST=$3R7_8$). Transfer PC to the stack address given by SP, decrement SP twice, and jump to the address specified by R.
SPHL		1	1/5	(SP)←(HL).

2.2.6 REGISTER GROUP INSTRUCTIONS

MNEMONIC	OPERANDS	BYTES	M CYCLES/ STATES	DESCRIPTION
DCR	M	1	3/10	(M)←(M)−1, decrement the contents of memory location specified by H and L. {Z,S,P,C1}
DCR	r_a	1	1/5	$(r_a)\leftarrow(r_a)-1$, decrement the contents of register r_a. {Z,S,P,C1}
DCX	r_b	1	1/5	$(r_b)\leftarrow(r_b)-1$, decrement double registers BC, DE, HL, or SP.
INR	M	1	3/10	(M)←(M)+1, increment the contents of memory location specified by H and L. { Z,S,P,C1 }
INR	r_a	1	1/5	$(r_a)\leftarrow(r_a)+1$, increment the contents of register r_a. {Z,S,P,C1}
INX	r_b	1	1/5	$(r_b)\leftarrow(r_b)+1$, increment double registers BC, DE, HL, or SP.
LHLD	b_3b_2	3	5/16	$(L)\leftarrow[<b_3> <b_2>]$; $(H)\leftarrow [<b_3> <b_2>+1]$, load registers H and L with contents of the two memory locations specified by bytes 3 and 2.
LXI	$r_bb_3b_2$	3	3/10	$(r_{bH})\leftarrow<b_3>$; $(r_{bL})\leftarrow<b_2>$, load double registers BC, DE, HL, or SP immediate with bytes 3, 2, respectively.
MVI	M,b_2	2	3/10	$(M)\leftarrow<b_2>$, store immediate byte 2 in the address specified by HL
MVI	r_ab_2	2	2/7	$(r_a)\leftarrow<b_2>$, load register r_a immediate with byte 2 of the instruction.
MOV	Mr_a	1	2/7	$(M)\leftarrow(r_a)$, store register r_a in the memory location addressed by H and L.
MOV	r_aM	1	2/7	$(r_a)\leftarrow(M)$, load register r_a with contents of memory addressed by HL.
MOV	$r_{a1}r_{a2}$	1	1/5	$(r_{a1})\leftarrow(r_{a2})$, load register r_{a1} with contents of r_{a2}, r_{a2} contents remain unchanged.
SHLD	b_3b_2	3	5/16	$[<b_3> <b_2>]\leftarrow(L)$; $[<b_3> <b_2>+1)]\leftarrow(H)$, store the contents of H and L into two successive memory locations specified by bytes 3 and 2.
XCHG		1	1/4	(H)↔(D); (L)↔(E), exchange double registers HL and DE
XTHL		1	5/18	(L)↔[(SP)], (H)↔[(SP)+1], (SP)=(SP), exchange the top of the stack with register HL.

2.3 INSTRUCTION SET OPCODES ALPHABETICALLY LISTED

MNEMONIC	BYTES	DESCRIPTION	REGISTER AFFECTED	POSITIVE-LOGIC HEX OPCODE D7–D4	D3–D0	OCT	CLOCK CYCLES*	DEC
ACI	2	Add immediate to A with carry†		C	E	316	7	206
ADC M	1	Add memory to A with carry†		8	E	216	7	142
ADC r	1	Add register to A with carry†	B	8	8	210	4	136
			C	8	9	211		137
			D	8	A	[illegible]		138
			E	8	B	213		139
			H	8	C	214		140
			L	8	D	215		141
			A	8	F	217		143
ADD M	1	Add memory to A†		8	6	206	7	134
ADD r	1	Add register to A†	B	8	0	200	4	128
			C	8	1	201		129
			D	8	2	202		130
			E	8	3	203		131
			H	8	4	204		132
			L	8	5	205		133
			A	8	7	207		135
ADI	2	Add immediate to A†		C	6	306	7	198
ANA M	1	AND memory with A†		A	6	246	7	166
ANAr	1	AND register with A†	B	A	0	240	4	160
			C	A	1	241		161
			D	A	2	242		162
			E	A	3	243		163
			H	A	4	244		164
			L	A	5	245		165
			A	A	7	247		167
ANI	2	AND immediate with A†		E	6	346	7	230
CALL	3	Call unconditional		C	D	315	17	205
CC	3	Call on carry		D	C	334	11/17	220
CM	3	Call on minus		F	C	374	11/17	252
CMA	1	Complement A		2	F	057	4	047
CMC	1	Complement carry‡		3	F	077	4	063
CMP M	1	Compare memory with A†		B	E	276	7	190
CMP r	1	Compare register with A						
			B	B	8	270	4	184
			C	B	9	271		185
			D	B	A	272		186
			E	B	B	273		187
			H	B	C	274		188
			L	B	D	275		189
			A	B	F	277		191
CNC	3	Call on no carry		D	4	324	11/17	212
CNZ	3	Call on no zero		C	4	304	11/17	196
CP	3	Call on positive		F	4	364	11/17	244
CPE	3	Call on parity even		E	C	354	11/17	236
CPI	2	Compare immediate with A†		F	E	376	7	254
CPO	3	Call on parity odd		E	4	344	11/17	228
CZ	3	Call on zero		C	C	314	11/17	204
DAA	1	Decimal adjust A†		2	7	047	4	039

*Two possible cycle times (11/17) indicate instruction cycles dependent on condition flags.

†All flags (C, Z, S, P, C1) affected.

‡Only carry flag affected.

				POSITIVE-LOGIC			
			REGISTER	HEX OPCODE		CLOCK	DEC
MNEMONIC	BYTES	DESCRIPTION	AFFECTED	D7–D4	D3–D0	CYCLES	
DAD B	1	Add B&C to H&L‡		0	9	10	009
DAD D	1	Add D&E to H&L‡		1	9	10	025
DAD H	1	Add H&L to H&L‡		2	9	10	041
DAD SP	1	Add stack pointer to H&L‡		3	9	10	057
DCR M	1	Decrement Memory§		3	5	10	053
DCR r	1	Decrement Register§	B	0	5	5	005
			C	0	D		013
			D	1	5		021
			E	1	D		029
			H	2	5		037
			L	2	D		045
			A	3	D		061
DCX B	1	Decrement B&C		0	B	5	011
DCX D	1	Decrement D&E		1	B	5	027
DCX H	1	Decrement H&L		2	B	5	043
DCX SP	1	Decrement stack pointer		3	B	5	059
DI	1	Disable interrupts		F	3	4	243
EI	1	Enable interrupts		F	B	4	251
HLT	1	Halt		7	6	7	118
IN	2	Input		D	B	10	219
INR M	1	Increment memory§		3	4	10	052
INR r	1	Increment register§	B	0	4	5	004
			C	0	C		012
			D	1	4		020
			E	1	C		028
			H	2	4		036
			L	2	C		044
			A	3	C		060
INX B	1	Increment B&C register		0	3	5	003
INX D	1	Increment D&E register		1	3	5	019
INX H	1	Increment H&L register		2	3	5	035
INX SP	1	Increment stack pointer		3	3	5	051
JC	3	Jump on carry		D	A	10	218
JM	3	Jump on minus		F	A	10	250
JMP	3	Jump unconditional		C	3	10	195
JNC	3	Jump on no carry		D	2	10	210
JNZ	3	Jump on no zero		C	2	10	194
JP	3	Jump on positive		F	2	10	242
JPE	3	Jump on parity even		E	A	10	234
JPO	3	Jump on parity odd		E	2	10	226
JZ	3	Jump on zero		C	A	10	202
LDA	1	Load A direct		3	A	13	058
LDAX B	1	Load A indirect		0	A	7	010
LDAX D	1	Load A indirect		1	A	7	026
LHLD	3	Load H&L direct		2	A	16	042
LXI B	3	Load immediate register pair B&C		0	1	10	001
LXI D	3	Load immediate register pair D&E		1	1	10	017
LXI H	3	Load immediate register		2	1	10	033
LXI SP	3	Load immediate stack pointer		3	1	10	049

‡Only carry flag affected.

§All flags except carry affected.

MNEMONIC	BYTES	DESCRIPTION	REGISTER AFFECTED	POSITIVE-LOGIC HEX OPCODE D7–D4	POSITIVE-LOGIC HEX OPCODE D3–D0	OC	CLOCK CYCLES	DE
MOV M,r	1	Move register to memory	B	7	0	[illegible]	7	112
			C	7	1			113
			D	7	2			114
			E	7	3			115
			H	7	4			116
			L	7	5			117
			A	7	7			119
MOV r,M	1	Move memory to register	B	4	6	10	7	70
			C	4	E			078
			D	5	6			086
			E	5	E			094
			H	6	6			102
			L	6	E			110
			A	7	E			126
MOV r_1, r_2	1	Move register to register	B,B	4	0		5	064
			B,C	4	1			065
			B,D	4	2			066
			B,E	4	3			067
			B,H	4	4			068
			B,L	4	5			069
			B,A	4	7			071
			C,B	4	8			072
			C,C	4	9			073
			C,D	4	A			074
			C,E	4	B			075
			C,H	4	C			076
			C,L	4	D			077
			C,A	4	F			079
			D,B	5	0			080
			D,C	5	1			081
			D,D	5	2			082
			D,E	5	3			083
			D,H	5	4			084
			H,L	5	5			085
			D,A	5	7			087
			E,B	5	8			088
			E,C	5	9			089
			E,D	5	A			090
			E,E	5	B			091
			E,H	5	C			092
			E,L	5	D			093
			E,A	5	F			095
			H,B	6	0			096
			H,C	6	1			097
			H,D	6	2			098
			H,E	6	3			099
			H,H	6	4			100
			H,L	6	5			101
			H,A	6	7			103
			L,B	6	8			104

MNEMONIC	BYTES	DESCRIPTION	REGISTER AFFECTED	POSITIVE-LOGIC HEX OPCODE D7–D4	D3–D0	CLOCK CYCLES*
MOV r_1, r_2	1	Move register to register (continued)	L,C	6	9	105
			L,D	6	A	106
			L,E	6	B	107
			L,H	6	C	108
			L,L	6	D	109
			L,A	6	F	111
			A,B	7	8	120
			A,C	7	9	121
			A,D	7	A	122
			A,E	7	B	123
			A,H	7	C	124
			A,L	7	D	125
			A,A	7	F	127
MVI M	2	Move immediate memory		3	6	10 054
MVI r	2	Move immediate register	B	0	6	7 006
			C	0	E	014
			D	1	6	022
			E	1	E	030
			H	2	6	038
			L	2	E	046
			A	3	E	062
NOP	1	No operation	4	0	0	4 000
ORA M	1	OR memory with A†		B	6	7 182
ORA r	1	OR register with A†	B	B	0	4 176
			C	B	1	177
			D	B	2	[illegible]
			E	B	3	179
			H	B	4	180
			L	B	5	181
			A	B	7	183
ORI	2	OR immediate with A†		F	6	7 246
OUT	2	Output		D	3	10 211
PCHL	1	H&L to program counter		E	9	5 233
POP B	1	Pop register pair B&C off stack		C	1	10 193
POP D	1	Pop register pair D&E off stack		D	1	10 209
POP H	1	Pop register pair H&L off stack		E	1	10 225
POP PSW	1	Pop A and flags off stack†		F	1	10 241
PUSH B	1	Push register pair B&C		C	5	11 197
PUSH D	1	Push register pair D&C		D	5	11 213
PUSH H	2	Push register pair H&L on stack		E	5	11 229
PUSH PSW	1	Push A and Flags on stack		F	5	11 245
RAL	1	Rotate A left through carry ‡		1	7	4 023
RAR	1	Rotate A right through carry‡		1	F	4 031
RC	1	Return on carry		D	8	5/11 216
RET	1	Return		C	9	10 201
RLC	1	Rotate A left‡		0	7	4 007
RM	1	Return on minus		F	8	5/11 248
RNC	1	Return on no carry		D	0	5/11 208
RNZ	1	Return on no zero		C	0	5/11 192
RP	1	Return on positive		F	0	5/11 240

*Two possible cycles times (11/17) indicate instruction cycles dependent on condition flags.
†All flags (C, Z, S, P, C1) affected.
‡Only carry flag affected.

MNEMONIC	BYTES	DESCRIPTION	REGISTER AFFECTED	POSITIVE-LOGIC HEX OPCODE D7–D4	D3–D0	CLOCK CYCLES*	DEC
RPE	1	Return on parity even		E	8	5/11	232
RPO	1	Return on parity odd		E	0	5/11	224
RRC	1	Rotate A right‡		0	F	4	015
RST	1	Restart				11	
			PC←0000_{16}	C	7		199
			PC←0008_{16}	C	F		207
			PC←0010_{16}	D	7		215
			PC←0018_{16}	D	F		223
			PC←0020_{16}	E	7		231
			PC←0028_{16}	E	F		239
			PC←0030_{16}	F	7		247
			PC←0038_{16}	F	F		255
RZ	1	Return on Zero		C	8	5/11	200
SBB M	1	Subtract memory from A with borrow†		9	E	7	158
SBB r	1	Subtract register from A with borrow†	B	9	8	4	152
			C	9	9		153
			D	9	A		154
			E	9	B		155
			H	9	C		156
			L	9	D		157
			A	9	F		159
SBI	2	Subtract immediate from A with borrow†		D	E	7	222
SHLD	3	Store H&L direct		2	2	16	034
SPHL	1	H&L to stack pointer		F	9	5	249
STA	3	Store A direct		3	2	13	050
STAX B	1	Store A indirect		0	2	7	002
STAX D	1	Store A indirect		1	2	7	018
STC	1	Set carry‡		3	7	4	055
SUB M	1	Subtract memory from A†		9	6	7	150
SUB r	1	Subtract register from A†	B	9	0	4	144
			C	9	1		145
			D	9	2		146
			E	9	3		147
			H	9	4		148
			L	9	5		149
			A	9	7		151
SUI	2	Subtract immediate from A†		D	6	7	214
XCHG	1	Exchange D&E, H&L registers		E	B	4	235
XRA M	1	Exclusive OR memory with A†		A	E	7	174
XRA r	1	Exclusive OR register with A†	B	A	8 50	4	168
			C	A	9		169
			D	A	A		170
			E	A	B		171
			H	A	C		172
			L	A	D		173
			A	A	F		175
XRI	2	Exclusive OR immediate with A†		E	E	7	238
XTHL	1	Exchange top of stack H&L		E	3	18	227

*Two possible cycles times (11/17) indicate instruction cycles dependent on condition flags.

† All flags (C, Z, S, P, C1) affected.

‡ Only carry flag affected.

3. TMS 8080 ELECTRICAL AND MECHANICAL SPECIFICATIONS

3.1 ABSOLUTE MAXIMUM RATINGS OVER OPERATING FREE-AIR TEMPERATURE RANGE (UNLESS OTHERWISE NOTED)*

Supply voltage, V_{CC} (see Note 1)	−0.3 V to 20 V
Supply voltage, V_{DD} (see Note 1	−0.3 V to 20 V
Supply voltage, V_{SS} (see Note 1)	−0.3 V to 20 V
All input and output voltages (see Note 1)	−0.3 V to 20 V
Continuous power dissipation	1.5 W
Operating free-air temperature range	0°C to 70°C
Storage temperature range	−65°C to 150°C

*Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: Under absolute maximum ratings voltage values are with respect to the normally most negative supply voltage, V_{BB} (substrate). Throughout the remainder cf this data sheet, voltage values are with respect to V_{SS} unless otherwise noted.

3.2 RECOMMENDED OPERATING CONDITIONS

	MIN	NOM	MAX	UNIT
Supply voltage, V_{BB}	−4.75	−5	−5.25	V
Supply voltage, V_{CC}	4.75	5	5.25	V
Supply voltage, V_{DD}	11.4	12	12.6	V
Supply voltage, V_{SS}		0		V
High-level input voltage, V_{IH} (all inputs except clocks) (see Note 2)	3.3		V_{CC}+1	V
High-level clock input voltage, $V_{IH(\phi)}$	V_{DD}−1		V_{DD}+1	V
Low-level input voltage, V_{IL} (all inputs except clocks) (see Note 3)	−1		0.8	V
Low-level clock input voltage, $V_{IL(\phi)}$ (see Note 3)	−1		0.6	V
Operating free-air temperature, T_A	0		70	°C

3.3 ELECTRICAL CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (UNLESS OTHERWISE NOTED)

PARAMETER		TEST CONDITIONS	MIN	TYP†	MAX	UNIT
I_I	Input current (any input except clocks and data bus)	V_I = 0 V to V_{CC}			±10	μA
$I_{I(\phi)}$	Clock input current	$V_{I(\phi)}$ = 0 V to V_{DD}			±10	μA
$I_{I(DB)}$	Input current, data bus	$V_{I(DB)}$ = 0 V to V_{CC}			−100	μA
$I_{I(hold)}$	Address or data bus input current during hold	$V_{I(ad)}$ or $V_{I(DB)}$ = V_{CC}			10	μA
		$V_{I(ad)}$ or $V_{I(DB)}$ = 0 V			−100	
V_{OH}	High-level output voltage	I_{OH} = 100 μA	3.7			V
V_{OL}	Low-level output voltage	$I_{OL(DB)}$ = 1.7 mA, I_{OL} = 0.75 mA (any output except DB)			0.45	V
$I_{BB(av)}$	Average supply current from V_{BB}	Operating at $t_{c(\phi)}$ = 480 ns, T_A = 25°C		−0.01	−1	mA
$I_{CC(av)}$	Average supply current from V_{CC}			60	75	
$I_{DD(av)}$	Average supply current from V_{DD}			40	67	
C_i	Capacitance, any input except clock	V_{CC} = V_{DD} = V_{SS} = 0 V, V_{BB} = −4.75 to −5.25 V, f = 1 MHz, All other pins at 0 V		10	20	pF
$C_{i(\phi)}$	Clock input capacitance			5	10	
C_o	Output capacitance			10	20	

†All typical values are at T_A = 25°C and nominal voltages.

NOTES: 2. Active pull-up resistors of nominally 2 kΩ will be switched onto the data bus when DBIN is high and the data input voltage is more positive than V_{IH} min.

3. The algebraic convention where the most negative limit is designated as minimum is used in this specification for logic voltage levels only.

3.4 TIMING REQUIREMENTS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (SEE FIGURE 2)

		MIN	MAX	UNIT
$t_{c(\phi)}$	Clock cycle time (see Note 5)	480	2000	ns
$t_{r(\phi)}$	Clock rise time	5	50	ns
$t_{f(\phi)}$	Clock fall time	5	50	ns
$t_{w(\phi 1)}$	Pulse width, clock 1 high	60		ns
$t_{w(\phi 2)}$	Pulse width, clock 2 high	220		ns
$t_{d(\phi 1L\text{-}\phi 2)}$	Delay time, clock 1 low to clock 2	0		ns
$t_{d(\phi 2\text{-}\phi 1)}$	Delay time, clock 2 to clock 1	70		ns
$t_{d(\phi 1H\text{-}\phi 2)}$	Delay time, clock 1 high to clock 2 (time between leading edges)	130		ns
$t_{su(da\text{-}\phi 1)}$	Data setup time with respect to clock 1	50		ns
$t_{su(da\text{-}\phi 2)}$	Data setup time with respect to clock 2	150		ns
$t_{su(hold)}$	Hold input setup time	140		ns
$t_{su(int)}$	Interrupt input setup time	180		ns
$t_{su(rdy)}$	Ready input setup time	120		ns
$t_{h(da)}$	Data hold time (see Note 6)	$t_{PD(DBI)}$		ns
$t_{h(hold)}$	Hold input hold time	0		ns
$t_{h(int)}$	Interrupt input hold time	0		ns
$t_{h(rdy)}$	Ready input hold time	0		ns

NOTES: 5. $t_{c(\phi)} = t_{d(\phi 1L\text{-}\phi 2)} + t_{r(\phi 2)} + t_{w(\phi 2)} + t_{f(\phi 2)} + t_{d(\phi 2\text{-}\phi 1)} + t_{r(\phi 1)}$. 480 ns $\leqslant t_{c(\phi)} \leqslant$ 2000 ns.

6. The data input should be enabled using the DBIN status signal. No bus conflict can then occur and the data hold time requirement is thus assured.

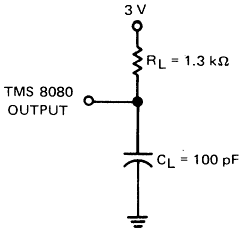
3.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (SEE FIGURE 2)

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{PD(ad)}$	Propagation delay time, clock 2 to address outputs	C_L = 100 pF, R_L = 1.3 kΩ		200	ns
$t_{PD(da)}$	Propagation delay time, clock 2 to data bus			220	ns
$t_{PD(cont)}$	Propagation delay time, clocks to control outputs			120	ns
$t_{PD(DBI)}$	Propagation delay time, clock 2 to DBIN output		25	140	ns
$t_{PD(int)}$	Propagation delay time, clock 2 to INTE output			200	ns
t_{DI}	Time for data bus to enter input mode			$t_{PD(DBI)}$	ns
t_{PXZ}	Disable time to high-impedance state during hold (address outputs and data bus)			120	ns

The time that the address outputs and output data will remain stable after $\overline{WR}$ goes high, t_{WA} and $t_{WD} \geqslant t_{d(\phi 1H\text{-}\phi 2)}$.
The time between address outputs becoming stable and $\overline{WR}$ going low, $t_{AW} \leqslant 2\ t_{c(\phi)} - t_{d(\phi 1H\text{-}\phi 2)} - t_{r(\phi)} - 120$ ns.
The time between output data becoming stable and $\overline{WR}$ going low, $t_{DW} \geqslant t_{c(\phi)} - t_{d(\phi 1H\text{-}\phi 2)} - t_{r(\phi)} - 150$ ns.
The following are relevant when interfacing to devices requiring V_{IH} min of 3.3 V:

a) Maximum output rise time (t_{TLH}) from 0.8 V to 3.3 V is 140 ns with C_L as specified for the propagation delay times above.

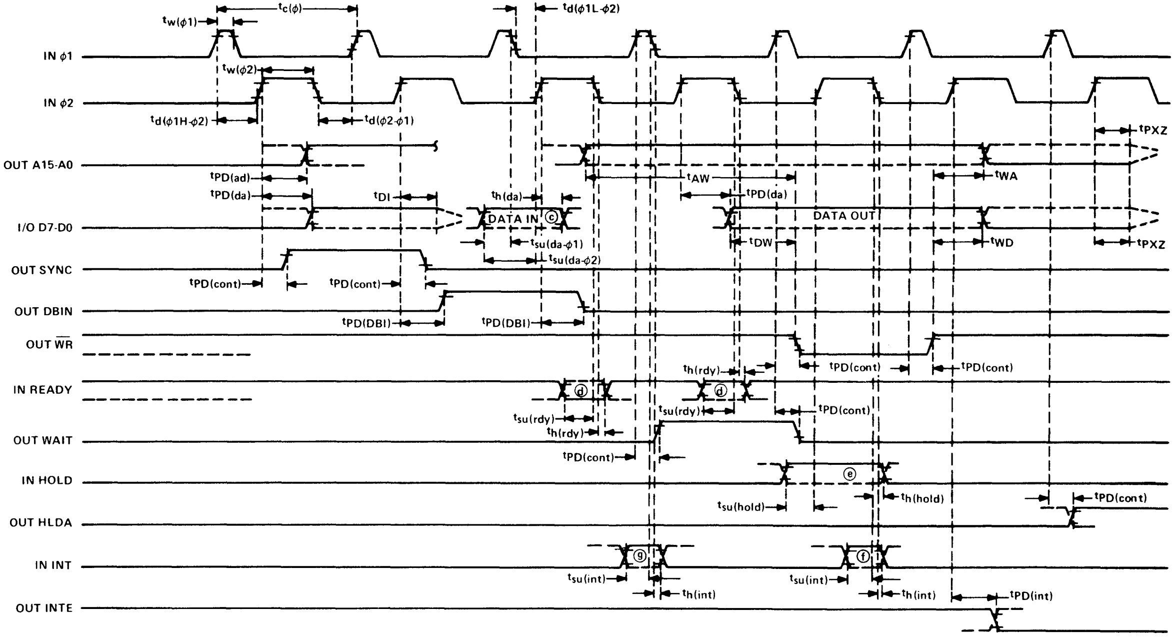
b) Maximum propagation delay times when measured to $V_{ref(H)}$ = 3 V (instead of 2 V) will be 60 ns more than as specified above with C_L as specified.



C_L includes probe and jig capacitance.

LOAD CIRCUIT

voltage waveforms (see notes a and b)



NOTES: a. This timing diagram shows timing relationships only, it does not represent any specific machine cycle.

b. Time measurements are made at the following reference voltages: Clock, $V_{ref(H)}$ = 9.5 V, $V_{ref(L)}$ = 1 V. Other inputs, $V_{ref(H)}$ = 2 V, $V_{ref(L)}$ = 0.8 V.

c. Data in must be stable for this period when DBIN is high during S3. Requirements for both $t_{su(da\text{-}\phi1)}$ and $t_{su(da\text{-}\phi2)}$ must be satisfied.

d. The ready signal must be stable for this period during S2 or SW. This requires external synchronization.

e. The hold signal must be stable for this period during S2 or SW when entering the hold mode and during S3, S4, S5 and SWH when in the hold mode. This requires external synchronization.

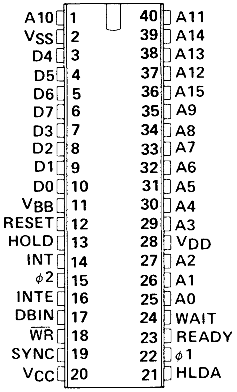
f. The interrupt signal must be stable during this period on the last clock cycle of any instruction to be recognized on the following instruction. External synchronization is not required.

g. During halt mode only, timing is with respect to the clock 1 falling edge.

FIGURE 2

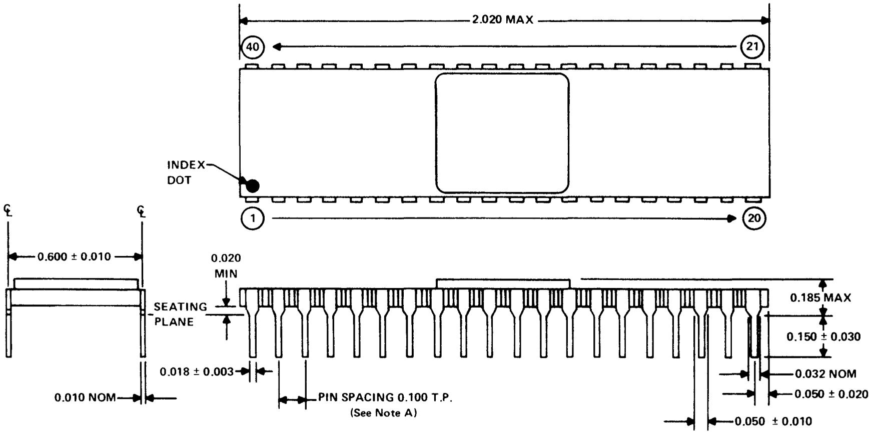
3.6 TERMINAL ASSIGNMENTS

TMS 8080



3.7 MECHANICAL DATA

40-PIN CERAMIC PACKAGE



NOTES: A. The true-position pin spacing is 0.100 between centerlines. Each pin centerline is located within 0.010 of its true longitudinal position relative to pins 1 and 40.

B. All dimensions are in inches unless otherwise noted.

1 Inch = 25,4 mm

TI cannot assume any responsibility for any circuits shown or represent that they are free from patent infringement. Texas Instruments reserves the right to make changes at any time in order to improve design and to supply the best product possible.

Printed in Germany

GERMANY
Texas Instruments
Deutschland G.m.b.H.
Haggertystr.1 , 8050 FREISING
Tel. 08161/80-1. Telex 0526529

FRANCE
Texas Instruments France
06 VILLENEUVE-LOUBET (A.M.)
Boite Postale 5
Tel. 93 31 03.64. Telex 46045

ITALY
Texas Instruments Italia
02100 RIETI (Italy)
Via Salaria per L'Aquilla, Cittaducale
Tel. 41314. Telex 62003

UNITED KINGDOM
Texas Instruments Limited
BEDFORD, England
Manton Lane
Tel. Bedford 67466. Telex 82178