# 8080A/Am9080A

8-Bit Microprocessor



## DISTINCTIVE CHARACTERISTICS

- High-speed version with 1.3μsec instruction cycle
- Military temperature range operation to 1.5μsec
- Ion-implanted, n-channel, silicon-gate MOS technology
- 3.2mA of output drive at 0.4V (two full TTL loads)
- 700mV of high, 400mV of low level noise immunity
- 820mW maximum power dissipation at ±5% power

## GENERAL DESCRIPTION

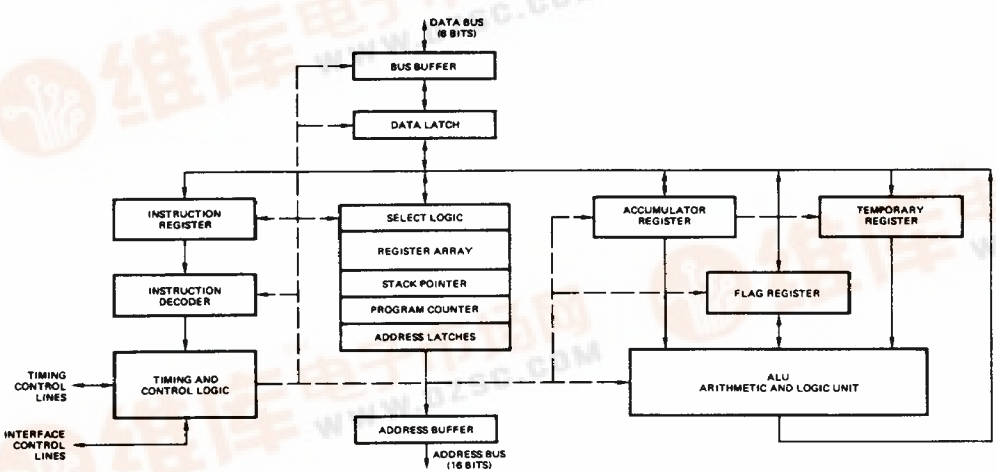
The 8080A products are complete, general-purpose, single-chip digital processors. They are fixed instruction set, parallel, 8-bit units fabricated with Advanced N-Channel Silicon Gate MOS technology. When combined with external memory and peripheral devices, powerful microcomputer systems are formed. The 8080A may be used to perform a wide variety of operations, ranging from complex arithmetic calculations to character handling to bit control. Several versions are available offering a range of performance options.

The processor has a 16-bit address bus that may be used to directly address up to 64K bytes of memory. The memory may be any combination of read/write and read-only. Data are transferred into or out of the processor on a bi-directional 8-bit data bus that is separate from the address lines. The data bus transfers instructions, data and status information between system devices. All transfers are handled using asynchronous handshaking controls so that any speed memory or I/O device is easily accommodated.

An accumulator plus six general registers are available to the programmer. The six registers are each 8 bits long and may be used singly or in pairs for both 8- and 16-bit operations. The accumulator forms the primary working register and is the destination for many of the arithmetic and logic operations.

A general purpose push-down stack is an important part of the processor architecture. The contents of the stack reside in R/W memory and the control logic, including a 16-bit stack pointer, is located on the processor chip. Subroutine call and return instructions automatically use the stack to store and retrieve the contents of the program counter. Push and Pop instructions allow direct use of the stack for storing operands, passing parameters and saving the machine state.

## BLOCK DIAGRAM



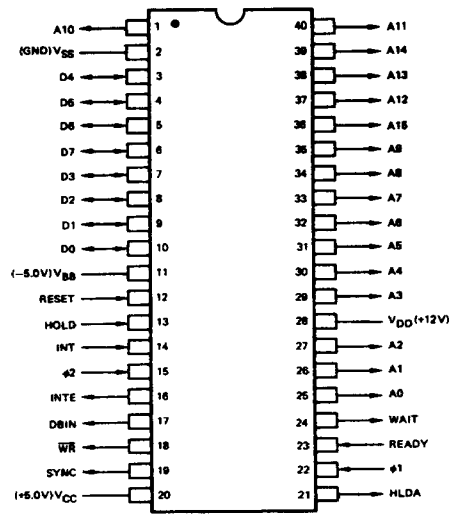

BD003800





Publication # 04123 | Rev. B | Amendment /0
Issue Date: May 1987

## CONNECTION DIAGRAM
### Top View
### DIPs

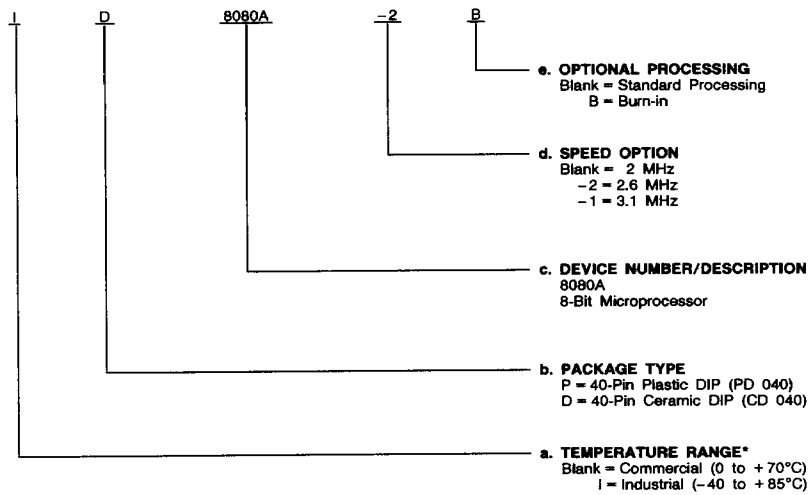



CD005573

Note: Pin 1 is marked for orientation.

## ORDERING INFORMATION – 8080A

AMD commodity products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of: **a. Temperature Range**
**b. Package Type**
**c. Device Number**
**d. Speed Option**
**e. Optional Processing**




| Valid Combinations | |
|---|---|
| P, D | 8080A |
| | 8080A-2 |
| | 8080A-1 |
| | 8080AB |
| | 8080A-2B |
| | 8080A-1B |
| tD | 8080AB |
| | 8080A-2B |

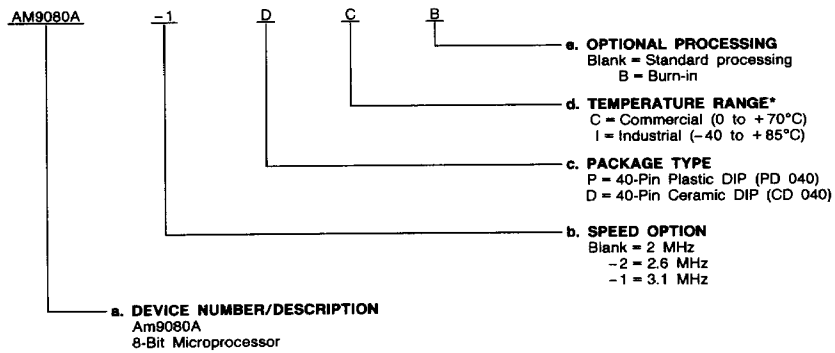
**Valid Combinations**

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released valid combinations, and to obtain additional data on AMD's standard military grade products.

*This device is also available in Military temperature range. See MOS Microprocessors and Peripherals Handbook (Order #09275A/0) for electrical performance characteristics.

## ORDERING INFORMATION – Am9080A

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of: **a. Device Number**
**b. Speed Option** (if applicable)
**c. Package Type**
**d. Temperature Range**
**e. Optional Processing**




| Valid Combinations | |
|---|---|
| AM9080A | PC, DC, |
| AM9080A-2 | DCB, DIB |
| AM9080A-1 | |

**Valid Combinations**

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released combinations, and to obtain additional data on AMD's standard military grade products.

*This device is also available in Military temperature range. See MOS Microprocessors and Peripherals Handbook (Order #09275A/0) for electrical performance characteristics.

## PIN DESCRIPTION

| TYPE | PINS | ABBREVIATION | SIGNAL |
|---|---|---|---|
| INPUT | 1 | $V_{SS}$ | Ground |
| INPUT | 3 | $V_{DD}$, $V_{CC}$, $V_{BB}$ | +12V, +5V, −5V Supplies |
| INPUT | 2 | $\phi_1$, $\phi_2$ | Clocks |
| INPUT | 1 | RESET | Reset |
| INPUT | 1 | HOLD | Hold |
| INPUT | 1 | INT | Interrupt |
| INPUT | 1 | READY | Ready |
| IN/OUT | 8 | $D_0$-$D_7$ | Data Bus |
| OUTPUT | 16 | $A_0$-$A_{15}$ | Address |
| OUTPUT | 1 | INTE | Interrupt Enable |
| OUTPUT | 1 | DBIN | Data Bus In Control |
| OUTPUT | 1 | $\overline{WR}$ | Write Not |
| OUTPUT | 1 | SYNC | Cycle Synchronization |
| OUTPUT | 1 | HLDA | Hold Acknowledge |
| OUTPUT | 1 | WAIT | Wait |

| Pin No. | Names | I/O | Description |
|---|---|---|---|
| 22, 15 | $\phi_1$, $\phi_2$ | I | The Clock inputs provide basic timing generation for all internal operations. They are non-overlapping two phase, high level signals. All other inputs to the processor are TTL compatible. |
| 12 | RESET | I | The Reset input initializes the processor by clearing the program counter, the instruction register, the interrupt enable flip-flop and the hold acknowledge flip-flop. The Reset signal should be active for at least three clock periods. The general registers are not cleared. |
| 13 | HOLD | I | The Hold input allows an external signal to cause the processer to relinquish control over the address lines and the data bus. When Hold goes active, the processor completes its current operation, activates the HLDA output, and puts the 3-state address and data lines into their high-impedance state. The Holding device can then utilize the address and data busses without interference. |
| 23 | READY | I | The Ready input synchronizes the processor with external units. When Ready is absent, indicating the external operation is not complete, the processor will enter the Wait state. It will remain in the Wait state until the clock cycle, following the appearance of Ready. |
| 14 | INT | I | The Interrupt input signal provides a mechanism for external devices to modify the instruction flow of the program in progress. Interrupt requests are handled efficiently with the vectored interrupt procedure and the general purpose stack. Interrupt processing is described in more detail on the next page. |
| 10-7, 3-6 | $D_0$-$D_7$ | I/O | The Data Bus is comprised of 8 bidirectional signal lines for transferring data, instructions and status information between the processor and all external units. |
| 25-27, 29-35, 1, 40, 37-39, 36 | $A_0$-$A_{15}$ | O | The Address Bus is comprised of 16 output signal lines used to address memory and peripheral devices. |
| 19 | SYNC | O | The Sync output indicates the start of each processor cycle and the presence of processor status information on the data bus. |
| 17 | DBIN | O | The Data Bus In output signal indicates that the bidirectional data bus is in the input mode and incoming data may be gated onto the Data Bus. |
| 24 | WAIT | O | The Wait output indicates that the processor has entered the Wait state and is prepared to accept a Ready from the current external operation. |
| 18 | $\overline{WR}$ | O | The Write output indicates the validity of output on the data bus during a write operation. |
| 21 | HLDA | O | The Hold Acknowledge output signal is a response to a Hold input. It indicates that processor activity has been suspended and the Address and Data Bus signals will enter their high-impedance state. |
| 16 | INTE | O | The Interrupt Enable output signal shows the status of the interrupt enable flip-flop, indicating whether or not the processor will accept interrupts. |

## 8080A/Am9080A INSTRUCTION SET

The instructions executed by the 8080A are variable length and may be one, two or three bytes long. The length is determined by the nature of the operation being performed and the addressing mode being used.

The instruction summary shows the number of successive memory bytes occupied by each instruction, the number of clock cycles required for the execution of the instruction, the binary coding of the first byte of each instruction, the mnemonic coding used by assemblers and a brief description of each operation. Some branch-type instructions have two execution times depending on whether the conditional branch is taken or not. Some fields in the binary code are labeled with alphabetic abbreviations. That shown as vvv is the address pointer used in the one-byte Call instruction (RST). Those shown as ddd or sss designate destination and source register fields that may be filled as follows:

111 A register
000 B register
001 C register
010 D register
011 E register
100 H register
101 L register
110 Memory

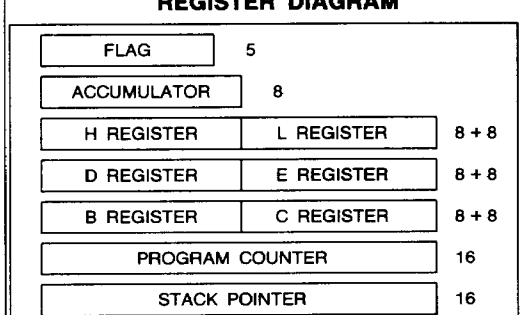
The register diagram shows the internal registers that are directly available to the programmer. The accumulator is the primary working register for the processor and is a specified or implied operand in many instructions. All I/O operations take place via the accumulator. Registers H, L, D, E, B and C may be used singly or in the indicated pairs. The H and L pair is the implied address pointer for many instructions.

The Flag register stores the program status bits used by the conditional branch instructions: carry, zero, sign and parity. The fifth flag bit is the intermediate carry bit. The flags and the accumulator can be stored on or retrieved from the stack with a single instruction. Bit positions in the flag register when pushed onto the stack (PUSH PSW) are:

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| S | Z | 0 | CY1 | 0 | P | 1 | CY2 |

Where S = sign, Z = zero, CY1 = intermediate carry, P = parity, CY2 = carry.

### REGISTER DIAGRAM




During Sync time at the beginning of each instruction cycle, the data bus contains operation status information that describes the machine cycle being executed. Positions for the status bits are:

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| MEMR | INP | M1 | OUT | HLTA | STK | $\overline{WO}$ | INTA |

### STATUS DEFINITION:

**INTA** Interrupt Acknowledge. Occurs in response to an Interrupt input and indicates that the processor will be ready for an interrupt instruction on the data bus when DBIN goes true.

**$\overline{WO}$** Write or Output indicated when signal is LOW. When HIGH, a Read or Input will occur.

**STK** Stack indicates that the content of the stack pointer is on the address bus.

**HLTA** Halt Acknowledge.

**OUT** Output instruction is being executed.

**M1** First instruction byte is being fetched.

**INP** Input instruction is being executed.

**MEMR** Memory Read operation.

### INTERRUPT PROCESSING

When the processor interrupt mechanism is enabled (INTE = 1), interrupt signals from external devices will be recognized unless the processor is in the Hold State. In handling an interrupt, the processor will complete the execution of the current instruction, disable further interrupts and respond with INTA status instead of executing the next sequential instruction in the interrupted program.

The interrupting device should supply an instruction opcode to the processor during the next DBIN time after INTA status appears.

Any opcode may be used except XTHL. If the instruction supplied is a single byte instruction, it will be executed. (The usual single byte instruction utilized is RST.) If the interrupt instruction is two or three bytes long, the next one or two processor cycles, as indicated by the DBIN signal, should be used by the external device to supply the succeeding byte(s) of the interrupt instruction. Note that INTA status from the processor is not present during these operations.

If the interrupt instruction is not some form of CALL, it is executed normally by the processor except that the Program Counter is not incremented. The next instruction in the interrupted program is then fetched and executed. Notice that the interrupt mechanism must be re-enabled by the processor before another interrupt can occur.

If the interrupt instruction is some form of CALL, it is executed normally. The Program Counter is stored and control transferred to the interrupt service subroutine. The routine has responsibility for saving and restoring the machine state and for re-enabling interrupts if desired. When the interrupt service is complete, a RETURN instruction will transfer control back to the interrupted program.

# INSTRUCTION SET SUMMARY

## DATA TRANSFER

| Op Code 7\|6\|5\|4\|3\|2\|1\|0 | No. of Bytes | Clock Cycles | Assembly Mnemonic | Instruction Description |
|---|---|---|---|---|
| 0 1 d d d s s s | 1 | 5 | MOVr, r | Move register to register |
| 0 1 1 1 0 s s s | 1 | 7 | MOVm, r | Move register to memory |
| 0 1 d d d 1 1 0 | 1 | 7 | MOVr, m | Move memory to register |
| 0 0 d d d 1 1 0 | 2 | 7 | MVI, r | Move to register, immediate |
| 0 0 1 1 0 1 1 0 | 2 | 10 | MVI, m | Move to memory, immediate |
| 0 0 1 1 1 0 1 0 | 3 | 13 | LDA | Load Acc, direct |
| 0 0 0 0 1 0 1 0 | 1 | 7 | LDAX B | Load Acc, indirect via B & C |
| 0 0 0 1 1 0 1 0 | 1 | 7 | LDAX D | Load Acc, indirect via D & E |
| 0 0 1 0 1 0 1 0 | 3 | 16 | LHLD | Load H & L, direct |
| 0 0 1 0 0 0 0 1 | 3 | 10 | LXI H | Load H & L, immediate |
| 0 0 0 1 0 0 0 1 | 3 | 10 | LXI D | Load D & E, immediate |
| 0 0 0 0 0 0 0 1 | 3 | 10 | LXI B | Load B & C, immediate |
| 0 0 1 1 0 0 0 1 | 3 | 10 | LXI SP | Load stack pointer, immediate |
| 0 0 1 0 0 0 1 0 | 3 | 16 | SHLD | Store H&L, direct |
| 0 0 1 1 0 0 1 0 | 3 | 13 | STA | Store Acc, direct |
| 0 0 0 0 0 0 1 0 | 1 | 7 | STAX B | Store Acc, indirect via B & C |
| 0 0 0 1 0 0 1 0 | 1 | 7 | STAX D | Store Acc, indirect via D & E |
| 1 1 1 1 1 0 0 1 | 1 | 5 | SPHL | Transfer H & L to stack pointer |
| 1 1 1 0 1 0 1 1 | 1 | 4 | XCHG | Exchange D & E with H & L |
| 1 1 1 0 0 0 1 1 | 1 | 18 | XTHL | Exchange top of stack with H & L |
| 1 1 0 1 1 0 1 1 | 2 | 10 | IN | Input to Acc |
| 1 1 0 1 0 0 1 1 | 2 | 10 | OUT | Output from Acc |

## CONTROL

| | | | | |
|---|---|---|---|---|
| 0 1 1 1 0 1 1 0 | 1 | 7 | HLT | Halt and enter wait state |
| 0 0 1 1 0 1 1 1 | 1 | 4 | STC | Set carry flag |
| 0 0 1 1 1 1 1 1 | 1 | 4 | CMC | Complement carry flag |
| 1 1 1 1 1 0 1 1 | 1 | 4 | EI | Enable interrupts |
| 1 1 1 1 0 0 1 1 | 1 | 4 | DI | Disable interrupts |
| 0 0 0 0 0 0 0 0 | 1 | 4 | NOP | No operation |

## BRANCH

| | | | | |
|---|---|---|---|---|
| 1 1 0 0 0 0 1 1 | 3 | 10 | JMP | Jump unconditionally |
| 1 1 0 1 1 0 1 0 | 3 | 10 | JC | Jump on carry |
| 1 1 0 1 0 0 1 0 | 3 | 10 | JNC | Jump on no carry |
| 1 1 0 0 1 0 1 0 | 3 | 10 | JZ | Jump on zero |
| 1 1 0 0 0 0 1 0 | 3 | 10 | JNZ | Jump on not zero |
| 1 1 1 1 0 0 1 0 | 3 | 10 | JP | Jump on positive |
| 1 1 1 1 1 0 1 0 | 3 | 10 | JM | Jump on minus |
| 1 1 1 0 1 0 1 0 | 3 | 10 | JPE | Jump on parity even |
| 1 1 1 0 0 0 1 0 | 3 | 10 | JPO | Jump on parity odd |
| 1 1 0 0 1 1 0 1 | 3 | 17 | CALL | Call unconditionally |
| 1 1 0 1 1 1 0 0 | 3 | 17-11 | CC | Call on carry |
| 1 1 0 1 0 1 0 0 | 3 | 17-11 | CNC | Call on no carry |
| 1 1 0 0 1 1 0 0 | 3 | 17-11 | CZ | Call on zero |
| 1 1 0 0 0 1 0 0 | 3 | 17-11 | CNZ | Call on not zero |
| 1 1 1 1 0 1 0 0 | 3 | 17-11 | CP | Call on positive |
| 1 1 1 1 1 1 0 0 | 3 | 17-11 | CM | Call on minus |
| 1 1 1 0 1 1 0 0 | 3 | 17-11 | CPE | Call on parity even |
| 1 1 1 0 0 1 0 0 | 3 | 17-11 | CPO | Call on parity odd |
| 1 1 0 0 1 0 0 1 | 1 | 10 | RET | Return unconditionally |
| 1 1 0 1 1 0 0 0 | 1 | 11-5 | RC | Return on carry |
| 1 1 0 1 0 0 0 0 | 1 | 11-5 | RNC | Return on no carry |
| 1 1 0 0 1 0 0 0 | 1 | 11-5 | RZ | Return on zero |
| 1 1 0 0 0 0 0 0 | 1 | 11-5 | RNZ | Return on not zero |
| 1 1 1 1 0 0 0 0 | 1 | 11-5 | RP | Return on positive |
| 1 1 1 1 1 0 0 0 | 1 | 11-5 | RM | Return on minus |
| 1 1 1 0 1 0 0 0 | 1 | 11-5 | RPE | Return on parity even |
| 1 1 1 0 0 0 0 0 | 1 | 11-5 | RPO | Return on parity odd |
| 1 1 1 0 1 0 0 1 | 1 | 5 | PCHL | Jump unconditionally, indirect via H & L |
| 1 1 V V V 1 1 1 | 1 | 11 | RST | Restart |

## ARITHMETIC

| | | | | |
|---|---|---|---|---|
| 1 0 0 0 0 s s s | 1 | 4 | ADDr | Add register to Acc |
| 1 0 0 0 1 s s s | 1 | 4 | ADCr | Add with carry register to Acc |
| 1 0 0 0 0 1 1 0 | 1 | 7 | ADDm | Add memory to Acc |
| 1 0 0 0 1 1 1 0 | 1 | 7 | ADCm | Add with carry memory to Acc |
| 1 1 0 0 0 1 1 0 | 2 | 7 | ADI | Add to Acc, immediate |
| 1 1 0 0 1 1 1 0 | 2 | 7 | ACI | Add with carry to Acc, immediate |
| 0 0 0 0 1 0 0 1 | 1 | 10 | DAD B | Double add B & C to H & L |
| 0 0 0 1 1 0 0 1 | 1 | 10 | DAD D | Double add D & E to H & L |
| 0 0 1 0 1 0 0 1 | 1 | 10 | DAD H | Double add H & L to H & L |
| 0 0 1 1 1 0 0 1 | 1 | 10 | DAD SP | Double add stack pointer to H & L |
| 1 0 0 1 0 s s s | 1 | 4 | SUBr | Subtract register from Acc |
| 1 0 0 1 1 s s s | 1 | 4 | SBBr | Subtract with borrow register from Acc |
| 1 0 0 1 0 1 1 0 | 1 | 7 | SUBm | Subtract memory from Acc |
| 1 0 0 1 1 1 1 0 | 1 | 7 | SBBm | Subtract with borrow memory from Acc |
| 1 1 0 1 0 1 1 0 | 2 | 7 | SUI | Subtract from Acc, immediate |
| 1 1 0 1 1 1 1 0 | 2 | 7 | SBI | Subtract with borrow from Acc, immediate |
| 0 0 1 0 0 1 1 1 | 1 | 4 | DAA | Decimal adjust Acc |

## STACK OPERATIONS

| | | | | |
|---|---|---|---|---|
| 1 1 0 0 0 1 0 1 | 1 | 11 | PUSH B | Push registers B & C on stack |
| 1 1 0 1 0 1 0 1 | 1 | 11 | PUSH D | Push registers D & E on stack |
| 1 1 1 0 0 1 0 1 | 1 | 11 | PUSH H | Push registers H & L on stack |
| 1 1 1 1 0 1 0 1 | 1 | 11 | PUSH PSW | Push Acc and flags on stack |
| 1 1 0 0 0 0 0 1 | 1 | 10 | POP B | Pop registers B & C off stack |
| 1 1 0 1 0 0 0 1 | 1 | 10 | POP D | Pop registers D & E off stack |
| 1 1 1 0 0 0 0 1 | 1 | 10 | POP H | Pop registers H & L off stack |
| 1 1 1 1 0 0 0 1 | 1 | 10 | POP PSW | Pop Acc and flags off stack |

## LOGICAL

| | | | | |
|---|---|---|---|---|
| 1 0 1 0 0 s s s | 1 | 4 | ANA r | And register with Acc |
| 1 0 1 0 0 1 1 0 | 1 | 7 | ANA m | And memory with Acc |
| 1 1 1 0 0 1 1 0 | 2 | 7 | ANI | And with Acc, immediate |
| 1 0 1 0 1 s s s | 1 | 4 | XRA r | Exclusive or register with Acc |
| 1 0 1 0 1 1 1 0 | 1 | 7 | XRA m | Exclusive Or memory with Acc |
| 1 1 1 0 1 1 1 0 | 2 | 7 | XRI | Exclusive Or with Acc, immediate |
| 1 0 1 1 0 s s s | 1 | 4 | ORA r | Inclusive Or register with Acc |
| 1 0 1 1 0 1 1 0 | 1 | 7 | ORA m | Inclusive Or memory with Acc |
| 1 1 1 1 0 1 1 0 | 2 | 7 | ORI | Inclusive Or with Acc, immediate |
| 1 0 1 1 1 s s s | 1 | 4 | CMP r | Compare register with Acc |
| 1 0 1 1 1 1 1 0 | 1 | 7 | CMP m | Compare memory with Acc |
| 1 1 1 1 1 1 1 0 | 2 | 7 | CPI | Compare with Acc, immediate |
| 0 0 1 0 1 1 1 1 | 1 | 4 | CMA | Compliment Acc |
| 0 0 0 0 0 1 1 1 | 1 | 4 | RLC | Rotate Acc left |
| 0 0 0 0 1 1 1 1 | 1 | 4 | RRC | Rotate Acc right |
| 0 0 0 1 0 1 1 1 | 1 | 4 | RAL | Rotate Acc left through carry |
| 0 0 0 1 1 1 1 1 | 1 | 4 | RAR | Rotate Acc right through carry |

## INCREMENT/DECREMENT

| | | | | |
|---|---|---|---|---|
| 0 0 d d d 1 0 0 | 1 | 5 | INR r | Increment register |
| 0 0 1 1 0 1 0 0 | 1 | 10 | INR m | Increment memory |
| 0 0 0 0 0 0 1 1 | 1 | 5 | INX B | Increment extended B & C |
| 0 0 0 1 0 0 1 1 | 1 | 5 | INX D | Increment extended D & E |
| 0 0 1 0 0 0 1 1 | 1 | 5 | INX H | Increment extended H & L |
| 0 0 1 1 0 0 1 1 | 1 | 5 | INX SP | Increment stack pointer |
| 0 0 d d d 1 0 1 | 1 | 5 | DCR r | Decrement register |
| 0 0 1 1 0 1 0 1 | 1 | 10 | DCR m | Decrement memory |
| 0 0 0 0 1 0 1 1 | 1 | 5 | DCX B | Decrement extended B & C |
| 0 0 0 1 1 0 1 1 | 1 | 5 | DCX D | Decrement extended D & E |
| 0 0 1 0 1 0 1 1 | 1 | 5 | DCX H | Decrement extended H & L |
| 0 0 1 1 1 0 1 1 | 1 | 5 | DCX SP | Decrement stack pointer |

## ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... −65°C to +150°C
All Input or Output Voltages
With Respect to $V_{BB}$ ..... −0.3V to +20V
$V_{CC}$, $V_{DD}$ and $V_{SS}$ With
Respect to $V_{BB}$ ..... −0.3V to +20V
Power Dissipation ..... 1.5W

*Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.*

## OPERATING RANGES

Commercial (C) Devices
Temperature ($T_A$) ..... 0 to +70°C
Supply Voltage ($V_{CC}$) ..... 5V ±5%
($V_{BB}$) ..... −5V ±5%
($V_{DD}$) ..... 12V ±5%
Industrial (I) Devices
Temperature ($T_A$) ..... −40 to +85°C
Supply Voltage ($V_{CC}$) ..... 5V ±5%
($V_{BB}$) ..... −5V ±5%
($V_{DD}$) ..... 12V ±5%

*Operating ranges define those limits between which the functionality and parameters of the device are guaranteed.*

## DC CHARACTERISTICS over operating ranges unless otherwise specified

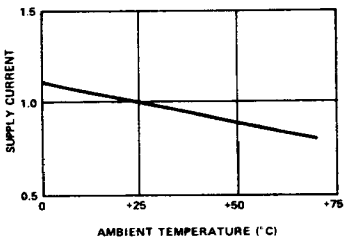
| Parameter | Description | Test Conditions | COM'L/IND Min | COM'L/IND Typ | COM'L/IND Max | Units |
|---|---|---|---|---|---|---|
| $V_{ILC}$ | Clock Input Low Voltage | | $V_{SS}-1$ | | $V_{SS}+0.8A$ | V |
| $V_{IHC}$ | Clock Input High Voltage | | 9.0 | | $V_{DD}+1$ | V |
| $V_{IL}$ | Input Low Voltage | | $V_{SS}-1$ | | $V_{SS}+0.8$ | V |
| $V_{IH}$ | Input High Voltage | | 3.3 | | $V_{CC}+1$ | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 1.9mA on all outputs, | | | 0.45 | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = −150μA. | 3.7 | | | V |
| $I_{DD(AV)}$ | Avg. Power Supply Current ($V_{DD}$) | operation $T_{CY}$ = .48μsec | | 40 | 70 | mA |
| $I_{CC(AV)}$ | Avg. Power Supply Current ($V_{CC}$) | | | 60 | 80 | mA |
| $I_{BB(AV)}$ | Avg. Power Supply Current ($V_{BB}$) | | | 0.01 | 1.0 | mA |
| $I_{IL}$ | Input Leakage | $V_{SS} \leq V_{IN} \leq V_{CC}$ | | | ±10 | μA |
| $I_{CL}$ | Clock Leakage | $V_{SS} \leq V_{CLOCK} \leq V_{DD}$ | | | ±10 | μA |
| $I_{DL}$ | Data Bus Leakage in Input Mode | $V_{SS} \leq V_{IN} \leq V_{SS}+0.8V$<br>$V_{SS}+0.8V \leq V_{IN} \leq V_{CC}$ | | | −100<br>−2.0 | μA<br>mA |
| $I_{FL}$ | Address and Data Bus Leakage During HOLD | $V_{ADDR/DATA} = V_{CC}$<br>$V_{ADDR/DATA} = V_{SS}+0.45V$ | | | +10<br>−100 | μA |

See Section 6 for Thermal Characteristics Information.

## CAPACITANCE ($T_A = 25°C$, $V_{CC} = V_{DD} = V_{SS} = 0V$, $V_{BB} = -5V$)

| Parameters | Description | Test Conditions | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $C_\phi$ | Clock Capacitance | $f_c = 1$ MHz | 17 | 25 | pf |
| $C_{IN}$ | Input Capacitance | Unmeasured Pins | 6 | 10 | pf |
| $C_{OUT}$ | Output Capacitance | Returned to $V_{SS}$ | 10 | 20 | pf |

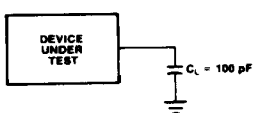
Notes: 1. The RESET signal must be active for a minimum of 3 clock cycles .
2. $\Delta I$ supply / $\Delta T_A = -0.45\%/°C$.




OP001690

Typical Supply Current vs. Temperature, Normalized [2]

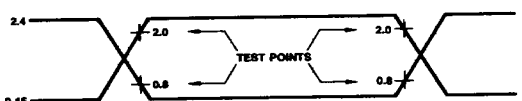
### SWITCHING TEST LOAD CIRCUIT




TC001840

$C_L = 100pF$
$C_L$ INCLUDES JIG CAPACITANCE

### SWITCHING TEST INPUT/OUTPUT WAVEFORM




WF007450

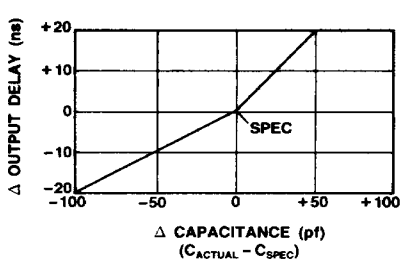
## SWITCHING CHARACTERISTICS over operating ranges unless otherwise specified

| Parameters | Description | Test Conditions | Min | Max | -1 Min | -1 Max | -2 Min | -2 Max | Unit |
|---|---|---|---|---|---|---|---|---|---|
| $t_{CY}$ [3] | Clock Period | | 0.48 | 2.0 | 0.32 | 2.0 | 0.38 | 2.0 | μsec |
| $t_r$, $t_f$ | Clock Rise and Fall Time | | 0 | 50 | 0 | 25 | 0 | 50 | nsec |
| $t_{\phi 1}$ | $\phi_1$ Pulse Width | | 60 | | 50 | | 60 | | nsec |
| $t_{\phi 2}$ | $\phi_2$ Pulse Width | | 220 | | 145 | | 175 | | nsec |
| $t_{D1}$ | Delay $\phi_1$ to $\phi_2$ | | 0 | | 0 | | 0 | | nsec |
| $t_{D2}$ | Delay $\phi_2$ to $\phi_1$ | | 70 | | 60 | | 70 | | nsec |
| $t_{D3}$ | Delay $\phi_1$ to $\phi_2$ Leading Edges | | 80 | | 60 | | 70 | | nsec |
| $t_{DA}$ | Address Output Delay From $\phi_2$ | $C_L$ = 100pF | | 200 | | 150 | | 175 | nsec |
| $t_{DD}$ | Data Output Delay From $\phi_2$ | $C_L$ = 100pF | | 200 | | 180 | | 200 | nsec |
| $t_{DC}$ | Signal Output Delay From $\phi_1$ or $\phi_2$ (SYNC, WR, WAIT, HLDA) | $C_L$ = 50pF | | 120 | | 110 | | 120 | nsec |
| $t_{DF}$ | DBIN Delay From $\phi_2$ | $C_L$ = 50pF | 25 | 140 | 25 | 130 | 25 | 140 | nsec |
| $t_{DI}$ [1] | Delay for Input Bus to Enter Input Mode | | | $t_{DF}$ | | $t_{DF}$ | | $t_{DF}$ | nsec |
| $t_{DS1}$ | Data Set-up Time During $\phi_1$ and DBIN | | 30 | | 10 | | 20 | | nsec |
| $t_{DS2}$ | Data Set-up Time to $\phi_2$ During DBIN | | 150 | | 120 | | 130 | | nsec |
| $t_{DH}$ [1] | Data Hold time From $\phi 2$ During DBIN | | [1] | | [1] | | [1] | | nsec |
| $t_{IE}$ | INTE Output Delay From $\phi_2$ | $C_L$ = 50 pF | | 200 | | 200 | | 200 | nsec |
| $t_{RS}$ | READY Set-up Time During $\phi_2$ | | 120 | | 90 | | 90 | | nsec |
| $t_{HS}$ | HOLD Set-up Time to $\phi_2$ | | 140 | | 120 | | 120 | | nsec |
| $t_{IS}$ | INT Set-up Time During $\phi_2$ | | 120 | | 100 | | 100 | | nsec |
| $t_H$ | Hold Time From $\phi_2$ (READY, INT, HOLD) | | 0 | | 0 | | 0 | | nsec |
| $t_{FD}$ | Delay to Float During Hold (Address and Data Bus) | | | 120 | | 120 | | 120 | nsec |
| $t_{AW}$ | Address Stable Prior to WR | | [5] | | [5] | | [5] | | nsec |
| $t_{DW}$ | Output Data Stable Prior to WR | | [6] | | [6] | | [6] | | nsec |
| $t_{WD}$ | Output Data Stable From WR | | [7] | | [7] | | [7] | | nsec |
| $t_{WA}$ | Address Stable From WR | $C_L$ = 100pF: Address, Data<br>$C_L$ = 50pF: $\overline{WR}$, HLDA, DBIN | [7] | | [7] | | [7] | | nsec |
| $t_{HF}$ | HLDA to Float Delay | | [8] | | [8] | | [8] | | nsec |
| $t_{WF}$ | WR to Float Delay | | [9] | | [9] | | [9] | | nsec |
| $t_{AH}$ | Address Hold Time After DBIN during HLDA | | -20 | | -20 | | -20 | | nsec |

**Notes:** (Parenthesis gives -1, -2 specifications, respectively)

1. Data input should be enabled with DBIN status. No bus conflict can then occur and data hold time is assured. $t_{DH}$ = 50 ns or $t_{DF}$, whichever is less.
2. $t_{CY} = t_{D3} + t_{r\phi 2} + t_{\phi 2} + t_{f\phi 2} + t_{D2} + t_{r\phi 1} \geq 480$ ns (-1:320 ns, -2:380 ns).

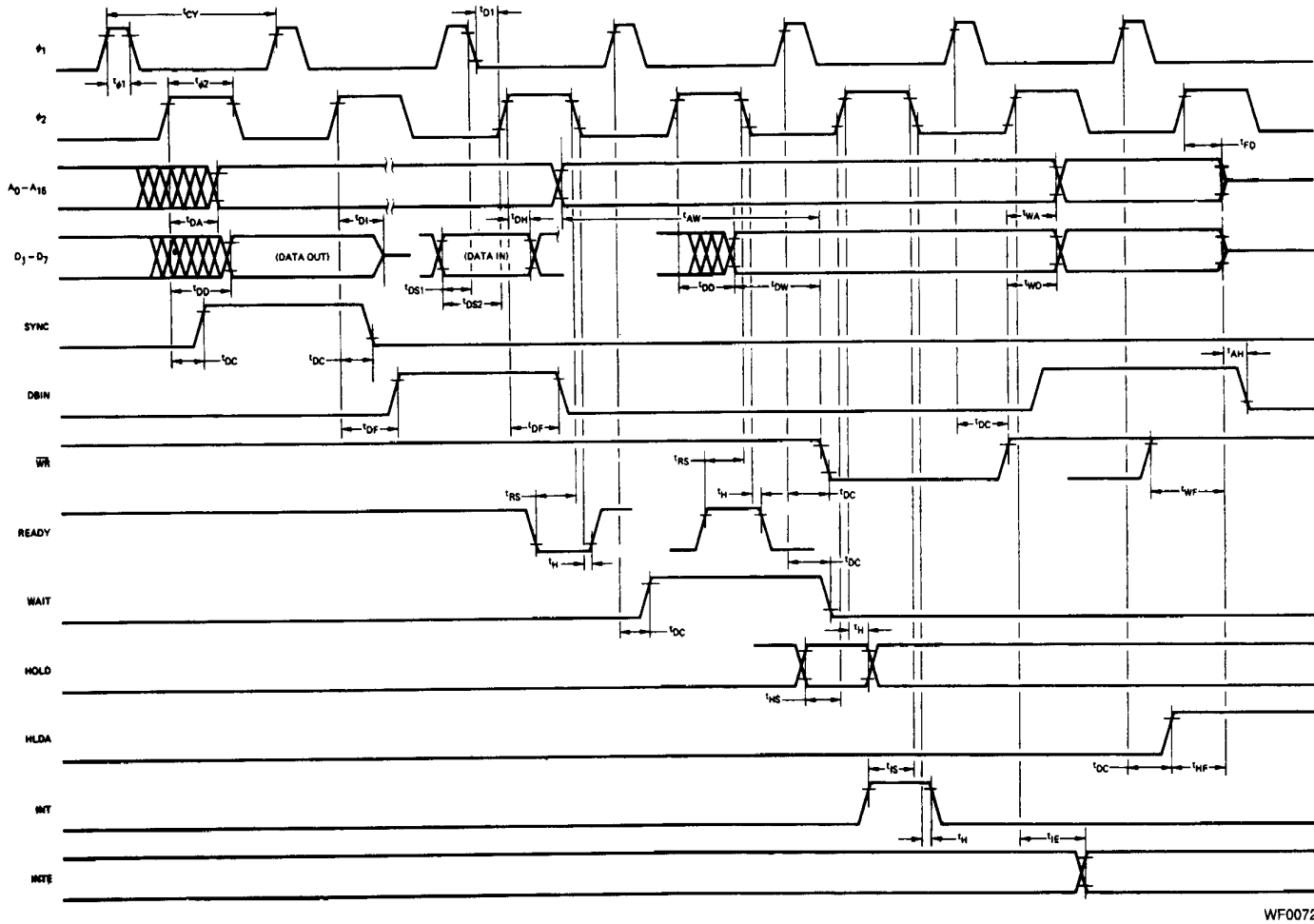
**TYPICAL Δ OUTPUT DELAY VS. Δ CAPACITANCE**




OP001810

3. The following are relevant when interfacing the 8080A to devices having $V_{IH}$ = 3.3V:
   a) Maximum output rise time from .8V to 3.3V = 100ns @ $C_L$ = SPEC.
   b) Output delay when measured to 3.0V = SPEC + 60ns @ $C_L$ = SPEC.
   c) If $C_L$ = SPEC, add .6ns/pF if $C_L > C_{SPEC}$, subtract .3ns/pF (from modified delay) if $C_L < C_{SPEC}$.
4. $t_{AW} = 2t_{CY} - t_{D3} - t_{r\phi 2} - 140$ ns (-1:110 ns, -2:130 ns).
5. $t_{DW} = t_{CY} - t_{D3} - t_{r\phi 2} - 170$ ns (-1:150 ns, -2:170 ns).
6. If not HLDA, $t_{WD} = t_{WA} = t_{D3} + t_{r\phi 2} + 10$ ns. If HLDA, $t_{WD} = t_{WA} = t_{WF}$.
7. $t_{HF} = t_{D3} + t_{r\phi 2} - 50$ ns).
8. $t_{WF} = t_{D3} + t_{r\phi 2} - 10$ ns.
9. Data in must be stable for this period during DBIN $T_3$. Both $t_{DS1}$ and $t_{DS2}$ must be satisfied.
10. Ready signal must be stable for this period during $T_2$ or $T_W$. (Must be externally synchronized.)
11. Hold signal must be stable for this period during $T_2$ or $T_W$ when entering hold mode, and during $T_3$, $T_4$, $T_5$ and $T_{WH}$ when in hold mode. (External synchronization is not required.)
12. Interrupt signal must be stable during this period of the last clock cycle of any instruction in order to be recognized on the following instruction. (External synchronization is not required.)
13. This timing diagram shows timing relationships only; it does not represent any specific machine cycle.

## SWITCHING WAVEFORMS




WF007240

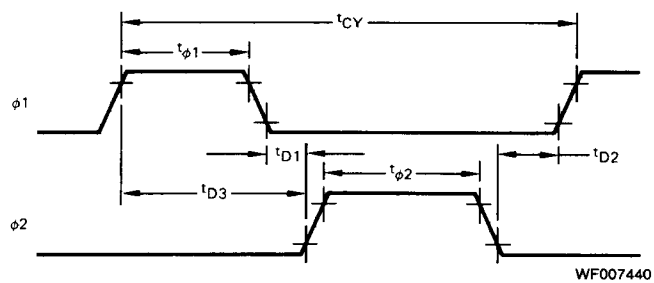
This chart presents relative timing waveform relationships and does not show actual processor operating cycles.

Note: Clock "1" = 8.0V, "0" = 1.0V; Inputs "1" = 3.3V, "0" = 0.8V; Outputs "1" = 2.0V, "0" = 0.8V.

## CLOCK SWITCHING CHARACTERISTICS over operating range unless otherwise specified

| Parameters | Description | Am9080A-1, 8080A-1 | | Am9080A-2, 8080A-2 | | Am9080A, 8080A | | Units |
|---|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | Min | Max | |
| $t_{CY}$ | Clock Period | 320 | 2000 | 380 | 2000 | 480 | 2000 | ns |
| $t_r$, $t_f$ | Clock Transition Times | 0 | 25 | 0 | 50 | 0 | 50 | ns |
| $t_{\phi 1}$ | Clock $\phi 1$ Pulse Width | 50 | | 60 | | 60 | | ns |
| $t_{\phi 2}$ | Clock $\phi 2$ Pulse Width | 145 | | 175 | | 220 | | ns |
| $t_{D1}$ | $\phi 1$ to $\phi 2$ Offset | 0 | | 0 | | 0 | | ns |
| $t_{D2}$ | $\phi 2$ to $\phi 1$ Offset | 60 | | 70 | | 70 | | ns |
| $t_{D3}$ | $\phi 1$ to $\phi 2$ Delay | 60 | | 70 | | 80 | | ns |

## CLOCK WAVEFORM DETAIL




$$t_{CY} = t_{D3} + t_{r\phi 2} + t_{\phi 2} + t_{f\phi 2} + t_{D2} + t_{r\phi 1}$$