# Table of Contents

1. **INTEL 8085 MICROPROCESSOR INTERNAL ARCHITECTURE** ............................................ 2  
   1.1 The 8085 CPU ........................................................................................................... 2  
   1.1.1 Address, Data and Control Buses ................................................................. 2  
   1.1.2 Timing and Control Units .............................................................................. 4  
   1.1.3 Interrupt control ............................................................................................... 5  
   1.1.4 Serial I/O Control ............................................................................................. 5  
   1.2 Register Section of 8085 CPU (8-Bit) Microprocessor ........................................... 5  
   1.2.1 The Instruction Register (IR) ........................................................................... 6  
   1.2.2 The Program Counter (PC) .............................................................................. 6  
   1.2.3 Accumulator (A) ............................................................................................... 7  
   1.2.4 The Status (Flag) Register ............................................................................... 7  
   1.2.5 Temporary Registers ....................................................................................... 8  
   1.2.6 General Purpose Registers (GPRs) ................................................................. 10  
   1.2.7 Stack Pointer (SP) ............................................................................................ 11  
   1.2.8 Index Register .................................................................................................. 11  
   1.2.9 The Memory Address Register (MAR) ............................................................ 11  
   1.2.10 The Incrementer /Decrementer Address Latch .............................................. 12  

2.0 **PIN DIAGRAM AND PIN DESCRIPTION OF 8085** .................................................. 13  
   2.1 Microprocessor Pin Description ............................................................................. 14  
   2.1.1 Tri state devices ............................................................................................... 14  
   2.1.2 Classification of Signals .................................................................................. 15  

3.0 **OPERATION OF MACHINE CYCLES** .................................................................... 21  
   3.1 Machine Cycle and Instruction Cycle ..................................................................... 21  
   3.2 Types of Machine cycles ...................................................................................... 22  
   3.3 Microprocessor-based system’s Memory addressing capability ................................ 23  

4.0 **MCS-85 FAMILY** ..................................................................................................... 24  
   Recommended Simulators .......................................................................................... 25  
   YouTube resources ................................................................................................... 26  
   References ................................................................................................................. 26
1. INTEL 8085 MICROPROCESSOR INTERNAL ARCHITECTURE

The collection of registers that constitutes a particular system and the data transfers that are possible among them make up the system architecture. The types of the registers in the microprocessor and the possible data transfers among them determine the microprocessor’s architecture.

A microprocessor system implements its functions by transferring and transforming data in the registers of the system. The microprocessor controls and synchronizes the data transfers and transformation according to the instructions read into it from the application program in the system’s ROM.

A block diagram of the internal architecture of the 8085 is shown in figure 1. The 8085 contains a register array with both dedicated and general purpose registers;

1. A 16-bit program counter (PC)
2. A 16-bit stack pointer (SP)
3. Six 8-bit general purpose registers arranged in pairs BC,DE,HL
4. A temporary register pair: WZ

1.1 The 8085 CPU

Intel’s 8085 is a 40-pin chip that is an enhancement of the 8080. It is a general purpose microprocessor that is made up of functional units capable of clock generation, system bus control and interrupts priority selection, in addition to execution of instructions.

1.1.1 Address, Data and Control Buses.

An 8-bit internal data bus carries instructions and data between the CPU registers. The external buses are the ones connected to other chips like memory, I/O and so on. The 8085 transfers data on an 8-bit bi-directional 3-state bus (AD0-7) which is time-multiplexed so as to also transmit the eight lower-order address bits. The main reason for multiplexing the AD0-7 buses was to retain the number of pin on the chip to 40.

An additional eight lines (A8-15) expand the 8085 family system memory addressing.
capability to 16 bits, thereby allowing 64K bytes of memory to be addressed directly by the CPU.

There are two buffer registers called the address buffer and the address-data buffer. The contents of the stack pointer or program counter registers can be loaded into the address buffer and address-data buffer. The output of these buffers then drives the external address bus and address-data bus. Memory and I/O chips are connected to these buses. The 8-bit internal data bus is also connected to the address-data buffer.

The 8085 can address up to 256 different I/O locations. The designer/programmer may also choose to address I/O ports as memory locations using memory-mapped I/O techniques.

Control bus is used to carry various control and status signals like RD, WR, ALE, READY,
1.1.2 Timing and Control Units

The 8085 timing and control unit generates control signals that can be used to select appropriate external devices and functions to perform READ and WRITE operations and also to select memory or I/O port. It includes an oscillator and a control-sequencer that produces the control signals needed for internal and external control.
The control-sequence is microprogrammed; it has a ROM that stores all the microroutines needed for executing the instructions. As each microinstruction is read out of the control ROM, control signals are sent to the internal and external data buses. The control ROM is sometimes called the control store.

Most of the control and status signals originate from this unit.

1.1.3 

**Interrupt control**

Interrupts are used to disrupt the normal sequence of main program execution to draw CPU’s attention to an external event e.g. a request from an I/O device. When the CPU receives an interrupt, it temporarily suspends the execution of the current program, attends to the interrupting device or event, and then returns to continue with the main program execution. The 8085 has five hardware interrupt inputs and one interrupt acknowledge output which is handled by the interrupt control section.

1.1.4 

**Serial I/O Control**

The SID and SOD pins help to minimize chip count in small systems by providing for easy interface to a serial port using software for timing and for coding and decoding of the state.

Each time a RIM instruction is executed, the status of the SID pin is read into bit 7 of the accumulator. Likewise, SIM is used to latch bit 7 of the accumulator out to the SOD output via an internal flip flop, providing that bit 6 of the accumulator is set to 1. SID can also be used as general purpose TEST input and SOD can serve as a one-bit control output.

1.2 **Register Section of 8085 CPU (8-Bit) Microprocessor**

The most common operation that takes place inside the microprocessor chip is the transfer of binary information from one register to another. The number and types of registers that a microprocessor contains is a key part of its architecture; and it has a major effect on the programming effort required in a given application.

Though the register structure/configuration differs for various microprocessors, the basic functions performed by the various registers are essentially the same in all microprocessors. They are used to store data, addresses, instruction codes and information on the status of
various microprocessor operations.
The 8085 is provided with internal 8-bit registers and 16-bit registers. The 8085 has eight addressable 8-bit registers. Six of them can be used either as 8-bit registers or as 16-bit register pairs. Register pairs are treated as though they were single 16-bit registers; the high-order byte of a pair is located in the first register and the low-order byte is located in the second. In addition to the register pairs, the 8085 contains two more 16-bit registers.
The registers in the 8085 CPU are:

1.2.1 The Instruction Register (IR).

This is used to store the opcode of the current instruction that is being fetched from the memory before execution. When the microprocessor fetches the opcode from memory, it stores it in the IR while the decoding circuitry determines the operation is to be performed. (From the IR, the opcode is taken to the Instruction Decoder which generates the control signals to activate the action specified by the opcode).
The IR is automatically used by the CPU during each Instruction cycle and the programmer cannot access it. The size of the IR is the same as the word size of the CPU e.g. 8-bits for 8085 CPU.
The output of the instruction decoder and the internal clock generator generates the state and machine cycle timing signals.

1.2.2 The Program Counter (PC)

The Program counter is used to contain the address in memory of the next instruction to be fetched (or executed) by the CPU. The CPU places the contents of the PC on the address bus and fetches the first byte of the instruction from that memory location.
The microprocessor automatically increments the PC after each use and in this way executes the stored program sequentially unless the program contains an instruction which alters the sequence (e.g. a JUMP Instruction).
The size of the PC depends on the number of address bits the microprocessor uses (e.g. 16-bits for the 8085 CPU).
1.2.3 Accumulator (A).

The Accumulator is a register that takes part in most of the operations performed by the ALU. It is also the register in which the results are placed after most ALU operations. In many ALU instructions; the Accumulator is the source of one of the operands and the destination of the result. (The contents of the accumulator serve as one input to the ALU, and the accumulator receives the output from the ALU).

In addition to its use in ALU instructions, the Accumulator can be used as a storage register for data that are being sent to an output or as a receiving register for data that are being read from an input device.

Some microprocessors have more than one Accumulator; more accumulators mean that more data can be held in the register section of the CPU thereby speeding up the program execution and data manipulation. The Accumulator generally has the same number of bits as the microprocessor word size (e.g. 8-bit for 8085 CPU).

1.2.4 The Status (Flag) Register.

Also called the process status register, the flag register consists of individual bit with different meanings assigned by the microprocessor manufacturer. These bits are called flags, and each flag is used to indicate the status of ALU/Accumulator operations. The various flags store status information, which is critical for microprocessor decision making. The value of the flags can be examined under program control to determine what sequence of instruction is to follow. The contents of the status register is also called the Program Status Word (PSW), of which must be known by the CPU to determine the operating conditions that may affect future operations.

The 8085 CPU flag register contains the following FIVE Flags:

(a). **Carry flag, CY.**

This flag is SET if the ALU operation has resulted in a carry and RESET if otherwise. It is used to indicate whether the sum exceeds the microprocessor word size.

(b). **Zero flag, Z.**

The microprocessor control signals SETS Z (to 1) when the result of an Arithmetic or
Logical operations has resulted into a zero and RESETS Z (to 0) when the result of the ALU operation is not zero.

A result that has a carry but has a zero answer byte in the accumulator will set both the carry flag and the zero flag.

(c). **Sign flag, S.**

This flag is SET if the ALU operation has resulted to a NEGATIVE answer and is RESET if the result is POSITIVE. The flag is set to the condition of the most significant bit of the accumulator following the execution of arithmetic or logic instruction. These instructions use bit 7 of the data to represent the sign of the number contained in the accumulator. This permits the manipulation of numbers in the range from -128 to +127.

(d). **Parity flag, P.**

Parity represents the number of 1s in the accumulator content. The Parity flag is SET if the result of an ALU operation has EVEN parity (even number of ones) and is RESET if the result has ODD parity.

(e). **Auxiliary Carry flag, Ac.**

This flag is SET if the ALU operation has resulted in a carry from bit 3 to bit 4 positions. It is mainly used in BCD arithmetic.

![8085 Flag Register](image)

Figure 2. 8085 Flag Register.

<table>
<thead>
<tr>
<th>S</th>
<th>Z</th>
<th>X</th>
<th>Ac</th>
<th>X</th>
<th>P</th>
<th>X</th>
<th>CY</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIGN</td>
<td>ZERO</td>
<td>AUXILIARY</td>
<td>CARRY</td>
<td>PARITY</td>
<td>CARRY</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

X—Don’t Care.

1.2.5. **Temporary Registers.**
The 8085 also has little-known hidden registers that are invisible/not accessible to the programmer but used internally: the WZ register pair (Each of these is an 8-bit register), and two 8-bit registers for the ALU: ACT and TMP. These temporary registers are used by the control unit to hold operands or addresses that are part of an instruction, until they are transferred to another register in the microprocessor or used as operands in a computation.

(a) ACT: The Accumulator Temporary register (ACT) holds the accumulator value while an ALU operation is performed. This allows the results from the ALU to be written back to the accumulator without disturbing the input, which would cause instability. It also holds constant values (e.g. for incrementing or decrementing, or decimal adjustment) without affecting the accumulator. Finally, the ACT allows ALU operations that don't use the accumulator.

(b) TMP: The second temporary register (TMP) holds the other argument for the ALU operation. The TMP register typically holds a value from memory or another register and feed the Accumulator. It may also be used as a temporary storage by the ALU.

Figure 3. Architecture of the 8085 ALU (http://www.righto.com)
(c) **WZ registers pair:**

This register pair is used as a temporary variable when the command that exchanges the contents of two register pairs is executed (i.e. when executing XCHG instruction). Other examples of the use of this register pair include when a 3-byte instruction containing a 2-byte address is to be fetched into the microprocessor. The first byte, the opcode is placed in the IR by the first memory reference. Two additional memory references obtain the two address bytes, which are placed in the temporary registers W and Z. These registers are used together as a register pair. During instruction execution, the address in W and Z is transferred to the address latch to address memory or I/O for data transfer.

The WZ register pair is not programmable.

1.2.6. **General Purpose Registers (GPRs).**

These registers are used for many of the temporary functions required inside the CPU. They can be used to store data that are currently used during a program execution, thereby speeding up the program execution; since the CPU does not have to perform a memory read operation each time that data is needed. They give the programmer a great deal of flexibility in dealing with a programming task.

These registers are often used as counters to keep track of the number of times a particular instruction sequence in a program has been executed. The number of the general purpose registers varies from microprocessor to microprocessor. The 8085 has SIX, which are labelled B, C, D, E, H, and L.

They can be used individually, such as when operating on an 8-bit data or in pairs such as when operating on 16-bit data or address. When used in pairs, only the following combinations are permitted for the 8085 CPU; B-C, D-E, and H-L.
1.2.7. Stack Pointer (SP).

The Stack is a portion of RAM reserved for the temporary storage and retrieval of information, typically the contents of the microprocessor’s internal registers. The data is stored and retrieved from the stack on a Last-In-First-Out (LIFO) or First-In-Last-Out (FILO) basis.

The Stack pointer, SP acts as a special memory address register used only for the stack portion of the RAM. Whenever a word is to be written/stored on or read from the stack, it is stored at or read from the address specified by the Stack pointer. Thus at all times, the SP holds the address of the TOP OF THE STACK.

The contents of the SP are automatically decremented after a word is stored on to the Stack and incremented before a word is read from the Stack. The incrementing and decrementing is done automatically by the CPU’s control unit.

There are some registers which are common in 8-bit processors but are not specifically incorporated for the 8085 CPU. These are

1.2.8 Index Register.

The Index register is an internal register within the microprocessor unit whose contents is modified (subtracted from or added to) during the execution of an instruction to determine the memory location which will be accessed by the CPU. The byte that is added to or subtracted from the Index register contents is called the Displacement.

Special instructions are provided which automatically adds this displacement to the register contents. Index Registers are used to access data from any sequential block of data (data tables).

1.2.9 The Memory Address Register (MAR).

This is sometimes called a storage address register or an address latching register. It is used to hold the address of the data the CPU is currently accessing in memory or writing into the memory. This leaves the PC to contain the address of the next opcode to be fetched.
Therefore there are TWO sources of address for the microprocessor address, the PC and the MAR. The PC is used for the opcode address while the MAR is used for data/operand address. A multiplexer is used to switch either the PC or the MAR onto the address bus, depending on whether the CPU is in opcode fetch cycle or memory read/write cycle.

1.2.10 The Incrementer /Decrementer Address Latch.

This 16-bit register is used to increment or decrement the content of the PC or SP as a part of execution of instruction related to them. It is actually a function performed by the ALU.

The incrementer is used to increment the PC value, after 8085 has fetched a byte of the instruction. This way, the PC will be pointing to the next instruction by the time the current instruction is fully fetched. It is also used for the incrementing the SP value after a byte is popped out of the stack top. It can also be used for incrementing an 8-bit or 16-bit register.

The decremener is used for decrementing the SP value before a byte is pushed above the stack top. It is also used for decrementing an 8-bit or 16-bit register.
2.0 PIN DIAGRAM AND PIN DESCRIPTION OF 8085

Figure 3. Intel 8085 microprocessor chip

Figure 4. Intel 8085 microprocessor Pin diagram
2.1 Microprocessor Pin Description

The 8085 microprocessor is a clock-driven semiconductor device consisting of electronic logic circuits manufactured by using either a large-scale integration (LSI) or very-large-scale integration (VLSI) technique. It is capable of performing various computing functions and making decisions to change the sequence of program execution. It has all the circuitry that makes up a CPU (hence it is a CPU on a single chip). 8085 microprocessor is a 40 pin IC, DIP package

2.1.1 Tri state devices

A trisate (3-state) device is a digital device which can have 3 output states, namely high, low and a high impedance state.

Example

When enable E is high the gate is enabled and the output Q can be 1 or 0 (if A is 0, Q is 1, otherwise Q is 0). However, when E is low the gate is disabled and the output Q enters into a high impedance state.

<table>
<thead>
<tr>
<th>Enable (E)</th>
<th>A</th>
<th>Q</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>High</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>Low</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td>High-Z Floating (high impedance)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td></td>
<td>High-Z Floating (high impedance)</td>
</tr>
</tbody>
</table>

When 2 or more devices are connected to a common bus, to prevent the devices from interfering with each other, the tristate gates are used to disconnect all devices except the one that is communicating at a given instant.
2.1.2 Classification of Signals

The various signals in a microprocessor can be grouped /classified as

\((a)\) Power supply and Clock frequency signals:

**Pins: 1, 2, 20, 37 and 40**

Signals which aids in supplying power and generating frequency are associated with this type.

Pins includes

- **Vcc** + 5 volt power supply
- **Vss** Ground
- **X1, X2:** These are the terminals which are connected to external oscillator.

The 8085 has an on-chip oscillator with all the required circuitry except for the crystal, LC, or RC network that controls the frequency. This external LC circuit, or RC network is connected to X1 and X2 (pin 1 and 2) to set the frequency of internal clock generator.

The frequency is internally divided by two. Since the basic operating timing frequency is 3 MHz, a 6 MHz crystal is connected externally.

- **CLK (OUT)** (output)-Clock Output is used as the system clock for peripheral and devices interfaced with the microprocessor. Its frequency is always same as the frequency at which the microprocessor operates.
(b) Multiplexed Address / Data Bus: AD0 - AD7 (input/output; 3-state)

Pins: 12 to 19

Pins 12 to 19 carry the lower 8 address bits or the 8 data bits. The lower half of the address bus is multiplexed with the data bus to keep the pin count at 40. These times multiplexed set of lines (AD0-AD7) are used to carry the lower order 8 bit address as well as data bus. During the opcode fetch operation, in the first clock cycle, the lines deliver the lower order address A0 - A7. It enables the address to get latched into the on-chip latch of peripherals. These bits are latched so that they contribute 8 bits out of the 16-bit address bits required during the memory operation. The falling edge of ALE is set to guarantee setup and hold times for the address information. It then becomes the data bus during the second and third clock cycles.

Pin 30: ALE stands for address latch enable. The falling edge of the ALE signal Strobes (loads) the address on the address bus. Normally each memory chip connected to the 8085 CPU has its own MAR, usually called address latch. This latch stores the incoming address from the address bus and address-data bus. The ALE signal comes out of pin 30 and goes to peripheral ships such as memory chips. It (ALE signal) occurs during the first clock state of a machine cycle and enables the address to get latched into the on-chip latch of peripheral. The falling edge of ALE is set to guarantee setup and hold times for the address information. The falling edge of ALE can also be used to strobe the status information. ALE is never 3-stated.
(c) **Address Bus:** A8 - A15 (output; 3-state)

**Pins 21 to 28**

They are used to carry the most significant 8 bits of the memory address or the 8 bits of the I/O address. They are 3-stated during the Hold and Halt modes and during RESET.

(d) **Control signals and Status:**

**Pin 30, 31, 32 and 34:**

- **ALE (output)** - Address Latch Enable. This signal helps to capture the lower order address presented on the multiplexed address / data bus.
- **RD (output 3-state, active low)** - Read memory or I/O device. This indicates that the selected memory location or I/O device is to be read and that the data bus is ready for accepting data from the memory or I/O device.
- **WR (output 3-state, active low)** - Write memory or IO device. This indicates that the data on the data bus is to be written into the selected memory location or I/O device.
- **IO/M (output)** - Select memory or an IO device. This status signal indicates that the read / write operation relates to whether the memory or I/O device. It goes high to indicate an I/O operation. It goes low for memory operations.

Pins RD, WR and IO/M function together. They are connected to memory and I/O chips via a decoder. A low IO/M indicates a memory operation (read or write) and a high IO/M means that an I/O instruction is being executed. In other words, a low IO/M signal enables the memory chips and a high IO/M enables the IO/M chips.

The RD and WR determine whether a read or a write is done. A low WR means a write operation and a low RD means a read operation. They are never both low at the same time.

For example, during a memory read, IO/M goes low, WR goes high, and RD goes low. When an output instruction (OUT) is executed, IO/M goes high; WR goes low and RD, goes high.
**SO, S1 (Output)**

**Pins : 29 and 33**

They carry output signals known as status signals showing the data bus status.

<table>
<thead>
<tr>
<th>S1</th>
<th>S0</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>HALT</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>WRITE</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>READ</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Opcode FETCH</td>
</tr>
</tbody>
</table>

These signals (and the IO/M signal) indicate whether an instruction fetch, memory read, memory write, IO read, IO write or other operation is taking place.

<table>
<thead>
<tr>
<th>IO/M</th>
<th>S1</th>
<th>S2</th>
<th>Data Bus Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Halt</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Memory WRITE</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Memory READ</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Opcode FETCH</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>Halt</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>IO WRITE</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>IO READ</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Interrupt Acknowledge</td>
</tr>
</tbody>
</table>

*Opcode Fetch* is a memory READ operation

**Interrupt Acknowledge** is an IO (Output) operation

- 3-state (high impedance)
- X = unspecified (don’t care)

(e)  **Interrupts Signals and other externally initiated signals:**

**Pins 6 to 11**

They are the signals initiated by an external device to request the microprocessor to do a particular task or work. They are part of interrupt control unit. The 8085 has five inputs for
interrupt requests.
A priority exists among the interrupt pins: some are more important than others. In order of their importance, the five interrupts signals are designated TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR. If two or more interrupts go high at the same time, the 8085 will service them in order of their importance (TRAP first, RST 7.5 second and so on).

**Pin 6 to 10** are inputs for the interrupt signals. **Pin 11** is an output pin with a signal called the interrupt acknowledge ($INTA$). This particular signal is used to response to an INTR interrupt.

- On receipt of an interrupt, the microprocessor acknowledges the interrupt by the active low ($INTA$) (Interrupt Acknowledge) signal.

**READY (input)**

**Pin 35.**
The main function of this pin is to synchronize slower peripheral to faster microprocessor. If ready pin is high during a read/write cycle, it indicates that memory or peripheral is ready to send/receive data the microprocessor will complete the operation and proceeds for the next operation. If ready pin is low the 8085 will wait for READY to go high before completing the read/write cycle. The READY signal is generated by the external memory or peripheral. The microprocessor enters into WAIT state while the READY pin is disabled.

**RESET IN** (Input, active low) (Input)

**Pin 36**
This signal is used to reset the microprocessor. This signal may come from an operator reset button or other source. When RESET is low, the CPU will reset the program counter, instruction register, Interrupt Enable and HLDA flip flops and other circuits. The buses are tri-stated. The CPU is held in the reset condition as long as Reset is applied. It also sends a high RESET OUT to pin 3
RESET OUT (Output)

Pin 3
RESET OUT indicates that CPU is being reset. It can be used as a system RESET (used to reset all the connected devices when the microprocessor is reset).
The signal is synchronized to the processor clock and lasts an integral number of clock periods. The data and address buses and the control lines are 3-stated during RESET and because of asynchronous nature of RESET, the processor internal registers and flags may be altered by RESET with unpredictable results.

HOLD (Input) - HOLD

Pin 39
This signal indicates that another Master (external device) is requesting the use of the Address and Data Buses.
The CPU, upon receiving the HOLD request will relinquish the use of the bus as soon as the completion of the current machine cycle. Internal processing which does not require the use of external buses can continue The processor can regain the buses only after the Hold is removed. When the Hold is acknowledged, the Address, Data, RD, WR, and IO/M lines are 3-stated.

HLDA (HOLD ACKNOWLEDGE)

Pin 38
HOLD ACKNOWLEDGE indicates that the CPU has received the Hold request and that it will relinquish the buses in the next clock cycle. HLDA goes low after the HOLD request is removed. The CPU takes the control of the buses one-half clock cycle after HLDA goes low.

(f) Serial communication Signal, SID and SOD.

Pin 5 and 4
Serial input data (SID), pin 5
Serial input data line is used to transfer serial data into the 8085 CPU. The data on this line is
loaded into accumulator bit 7 whenever a RIM instruction is executed.

*Serial output data (SOD), pin 4*

Serial Output data line is used to transfer serial data from the 8085 CPU. The output SOD is set or reset as specified by the SIM instruction. The SIM instruction loads the value of bit 7 of the accumulator into SOD latch if bit 6 of the accumulator is 1. SID and SOD are used for serial communication.

### 3.0 OPERATION OF MACHINE CYCLES

#### 3.1 Machine Cycle and Instruction Cycle

During normal operation, the microprocessor sequentially fetches and executes one instruction after another. Timing is provided by the clock. One clock period is called a timing state, or T-state. The number of timing T-states needed to fetch and execute an instruction is called *Instruction cycle*. The instruction cycle can also be referred to as the fetching and execution of a single instruction.

A *machine cycle* is a complete fetch-execute operation. The execution of each instruction by the 8085 consists of a sequence of from one to five machine cycles, and each machine cycle consists of a minimum of from three to six clock cycles.
The opcode fetch cycle is a complete machine cycle.

3.2 **Types of Machine cycles**

There are seven different types of machine cycles in the 8085 CPU

i. Opcode FETCH

ii. Memory READ

iii. Memory WRITE

iv. I/O READ

v. I/O WRITE

vi. Interrupt Acknowledge

vii. Bus Idle

Three status signals, IO/\bar{M}, S1, and S0, generated at the beginning of each machine cycle identify each type and remain valid for the duration of the cycle.

The first machine in an instruction cycle is always an opcode fetch, and the 8-bit obtained during an opcode fetch are always interpreted as the opcode of an instruction. The number of machine cycles required to execute an instruction depends on that particular instruction. More on this topic shall be discussed at *the timing diagrams* topic.
3.3 Microprocessor-based system’s Memory addressing capability

Assuming that a digital system uses one bit memory address, the number of addresses that can be generated is 2 (i.e. 0 and 1).
If the address bits are two, the number of addresses generated is 4 (i.e. 00, 01, 10, and 11).
If the bits are three, the number of addresses generated is 8 (i.e. 000,001,010,011,100,101, 110,111),
Therefore if the number of address bits is \( n \), the number of addresses generated by the digital system is \( 2^n \).

The 8085 CPU uses 16-bit address, hence the number of addresses generated by the 8085 is :

\[
2^{16} = 2^6 \times 2^{10} = 64K
\]

If each memory location stores 8-bits in the 8085 Microprocessor based system, the memory capacity that can be addressed by the 8085 CPU is

\[
64K \times 8 \text{ bits} = 64K\text{bytes (KB)}
\]

The 8085 processor uses 8-bits addressing for I/O devices. The total number of I/O devices that can be addressed by the CPU is

\[
2^8 = 256 \text{ I/O devices}
\]
4.0  **MCS-85 FAMILY**

These are the Chip used in 8085 microprocessor-based system

The 8085 CPU was one part of a family of chips developed by Intel, for building a complete system. Many of these support chips were also used with other processors. The original IBM PC based on the Intel 8088 processor used several of these chips.

1. 8085-CPU
2. 8155-RAM+ 3 I/O Ports+Timer
3. 8156-RAM+ 3 I/O Ports+Timer
4. 8185-SRAM
5. 8355-16,384-bit (2048 ×8) ROM with I/O
6. 8604-4096-bit (512 ×8) PROM
7. 8755-EPROM+2 I/O Ports
8. 8202-Dynamic RAM Controller
9. 8203-Dynamic RAM Controller
10. 8205-1 Of 8 Binary Decoder
11. 8206-Error Detection & Correction Unit
12. 8207-DRAM Controller
13. 8210-TTL To MOS Shifter & High Voltage Clock Driver
14. 8212-8-bit I/O Port
15. 8216-4-bit Parallel Bidirectional Bus Driver
16. 8218/8219-Bus Controller
17. 8226-4-bit Parallel Bidirectional Bus Driver
18. 8231-Arithmetic Processing Unit
19. 8232-Floating Point Processor
20. 8237-DMA Controller
21. 8251-Communication Controller
22. 8253-Programmable Interval Timer
23. 8254-Programmable Interval Timer
24. 8255-Programmable Peripheral Interface
24. 8256-Multifunction Support Controller
25. 8257-DMA Controller
26. 8259-Programmable Interrupt Controller
27. 8271-Programmable Floppy Disk Controller
28. 8272-Single/Double Density Floppy Disk Controller
29. 8273-Programmable HDLC/SDLC Protocol Controller
30. 8274-Multi-Protocol Serial Controller
31. 8275-CRT Controller
32. 8276-Small System CRT Controller
33. 8275-Programmable Key Board Interface
34. 8279-Key Board/Display Controller
35. 8282-8-bit Non-Inverting Latch with Output Buffer
36. 8283-8-bit Inverting Latch with Output Buffer
37. 8291-GPIB Talker/Listener
38. 8293-GPIB Transceiver
39. 8294-Data Encryption/Decryption Unit+1 O/P Port
40. 8295-Dot Matrix Printer Controller

8156-RAM+ 3 I/O Ports+Timer, 8251-Communication Controller and 8259-Programmable Interrupt Controller IC chips shall be discussed in this course

Recommended Simulators

1. GNUSim8085
   This consists of a simulator, assembler and a debugger. It is available for both Windows and Linux operating systems.

   Downloadable from http://gnusim8085.org/

   It is easy to use and will be used to simulate most assembly programs in this course.
2. **ENVI85A** is an integrated editor-assembler-emulator for the 8085 which makes programming the 8085 easy. It runs in DOS. It was written by professors Stefan Fedyschyn and Edwin Kay. Downloadable from the Lehigh university website. URL [http://www.lehigh.edu/~ejk0/envi85.html](http://www.lehigh.edu/~ejk0/envi85.html)

There are other suitable 8085 simulators available on the internet.

**YouTube resources**

[https://www.youtube.com/watch?v=I78iyzXQrP4](https://www.youtube.com/watch?v=I78iyzXQrP4)

**References**

1. Digital Computers Electronics; An introduction to Microcomputers. 2\textsuperscript{nd} edition by Albert Paul Malvino Ph.D.
2. 8085A/8085A-2 datasheet by Intel Corporation
4. The MCS-80/85 Family User manual by Intel Corporation
5. M. Krishna Kumar Lecture Notes
6. Assorted websites
7. Introduction to Microprocessor 8085, By Dr. D. K. Kaushik