microprocessor and microcontroller

Unit1; microprocessor & Applications

Introduction

Evolution of Microprocessor

- **1.1.2 History and Evolution of Computers**

Like human civilization computers also have evolved through centuries for many generations to reach to the level of today. People thought of computation and performing some elementary operations such as addition, subtraction, multiplication and division long back in 16th century or even earlier [1]. In those days these were some clever mechanical devices designed with gears, levers, wheels and the like.

- **1.1.2.1 The Mechanical Era**

● Blaze Pascal [1623-62], a French philosopher first invented an early, influential mechanical calculator that could add or subtract decimal numbers.

● Gottfried Leibniz [1946-1716], in Germany extended Pascal's design to perform multiplication and division.

Evolution of Microprocessor

- Charles Babbage’s difference engine in the 19th century to perform multistep

operations automatically without human intervention at every step.

- Some later developments in the design of general-purpose program-controlled

computer includes Z1 in 1938 (still a mechanical computer from Germany), Z3 in

1941,

- an automatic calculator known as Harvard Mark I in 1944 from Harvard University.

Evolution of Microprocessor

- **1.1.2.2. Electronic Age**
- The mechanical computers suffered from two serious limitations: it is inherently slow in
- computation because of its movable parts and the transmission of information by
- mechanical means is unreliable. Moreover, the size is also very large. With the development
- of electronic valves and vacuum tubes in early 1900, permitted the processing and storage
- of digital data at a much higher speed than that of any mechanical device.

Evolution of Microprocessor

- **The first generation (1940-1950)**

● First by John V. Atanasoff (1903-95), at Iowa State University, in late 1930s

● The Electronic Numeric Integrator and Calculator (ENIAC) by Mauchly and Eckert in the University of Pennsylvania, 1943-46

● The first commercial product by Eckert-Mauchly Corp was UNIAC (Universal Automatic Computer) in 1951.

● IAS Computer, by von Neumann, at the Institute for Advanced Studies in Princeton

began to work on the design of a new stored program electronic computer, 1947

Evolution of Microprocessor

- **The second generation (1954-64)**
- **Key Features**

● The vacuum tubes and electronics valves were soon replaced by bipolar junction transistors, in 1947

● the second-generation computers based on transistors soon replaced the first generation of vacuum tube-based machines

● drastic reduction in size and cost and power.

● Computational speed also enhanced to a great extent.

● ferrite cores became the dominant technology for main memories until it is superseded by the all-transistor memories in 1970s.

Evolution of Microprocessor

- Magnetic disks became the principal technologies for secondary memories since then.
- With the introduction of index registers, it is possible to have indexed instructions, which increments or decrements the designated index I.
- Another innovation in second generation was the introduction of program-control instructions: call and return which allow the linking of programs.
- It allowed to perform operations on floating point numbers,
- With the introduction of compilers in has become possible to write instructions in high level language

Evolution of Microprocessor

● For system management, batch processing came into existence. Batch processing makes use of supervisory program known as batch monitor which is a rudimentary version of operating system, which is a system program designed to manage computer’s resources efficiently.

The third generation (1965-75)

Key Features

● introduction of silicon based integrated circuits (ICs) in the design of computer hardware, in 1961

● This replaced all the second-generation computers designed with discrete transistors,

● reduced size, cost and enhanced the speed

Evolution of Microprocessor

- The most significant event during this period was the recognition of the need to standardize the computers,
- more and more software were developed and used more efficiently.
- IBM developed the most influential third-generation computer, the System/360

which was announced in 1964 and came in the year 1965.

- Since then System/360 became the de-facto standard and all the models in the series

are software compatible with each other (share common instruction set).

- Introduction of status register (SR) to save the program status word, for any

exceptional conditions, errors, divide by zero or any urgent service requests such as interrupts.

Evolution of Microprocessor

- **The VLSI Era**

With the advent of integrated circuits (IC) in 1959 at Texas Instruments [2] and its commercialization in1961, the dominant technology for manufacturing the computer logic circuits and memory has been the ICs. Initially, the ICs started with few transistors (less than 100) and gradually progressed through technology advancements allowing more and more devices to be accommodated in a single chip with the advent of metal-oxide semiconductor (MOS) ICs. This has resulted in medium scale (MSI), large scale (LSI) and very large scale integrated (VLSI) circuits containing 1000, 10000 and millions of transistors, respectively.

Evolution of Microprocessor

- This allowed to fabricate CPU, main memory, multichip module

or even all the electronic circuits of a computer on a single chip at a very low cost [3]. This is an enormous advancement allowing to develop a wide range of machines starting form

portable personal computers, microcontrollers to supercomputers containing thousands of CPUs.

8085 CPU- Architecture

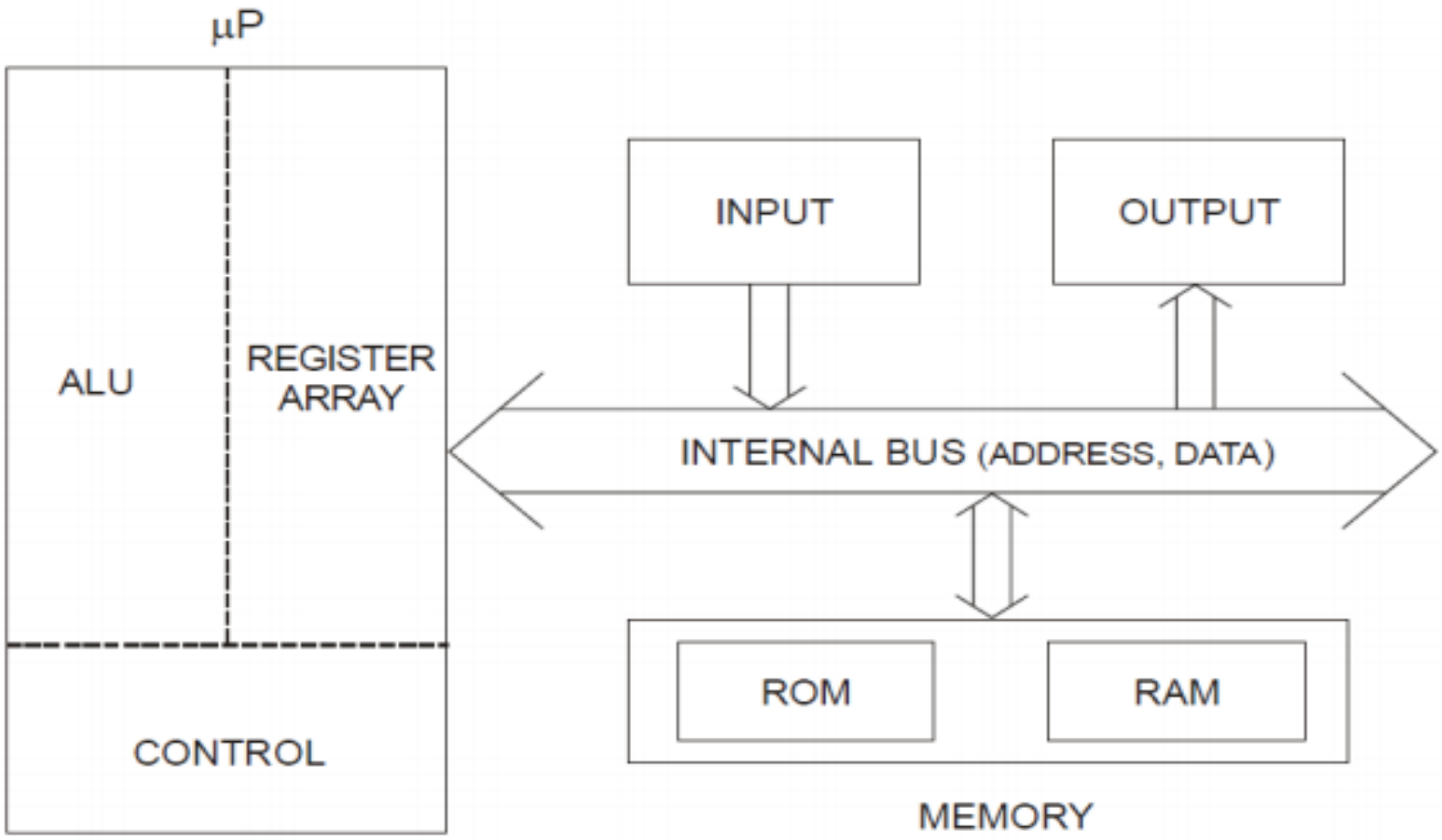


Figure 1.2 Architecture of Microprocessor

8085 CPU- Architecture

- **Buses**
- Figureabove shows interconnection of busses in various blocks. These busses allow exchange of words between the blocks. A bus has a wire or line for each bbit and thus allows exchange of all bits of a word in parallel. the processing of bits in microprocessor is also in parallel.The busses can thus be viewed as data highways.
- **The following are the types of busses**
- **Address Bus**
- It consists of set of connected wires known as bus which carries the address, to identify a memory location or an I/O port. It is usually a binary pattern of 0s and 1s. For example, an 8-bit address bus has eight lines thus it can identify 28 = 256 different locations. So, the locations in hexadecimal format can be written as 00H – FFH. It is unidirectional

8085 CPU- Architecture

- **Data Bus**
- The data bus is used to transfer data either between memory and processor or between I/O device and processor and vice-versa. For example, an 8-bit processor will generally have an 8-bit data bus and a 16-bit processor will have 16-bit data bus. It is bidirectional.
- **Control Bus**
- The control unit in the processor generates the control signals and the control bus carry those signals, which consists of signals for selection of memory or I/O device from the given address, selecting and controlling the appropriate functional units also for direction of data transfer and synchronization of data transfer in case of slow devices.
- Control Bus The control unit in the processor generates the control signals and the control bus carry those signals, which consists of signals for selection of memory or I/O device from the given address, selecting and controlling the appropriate functional units also for direction of data transfer and synchronization of data transfer in case of slow devices.

8085 CPU- Architecture

- **1.3 Overview of 8085 Microprocessor**
- 8085 microprocessor is an 8-bit processor developed by Intel corporation in the year 1976. Although it is the most basic processor but it has all the important features of today's higher bit microprocessors.
- So, to understand the internal architecture of a microprocessor it is better to start with a simple processor.
- **Key Features of 8085**

- It is an 8-bit processor
- It is a single chip MOS device with 40 pins

8085 CPU- Architecture

- It has 8-bit data bus and 16-bit address bus. However, the lower order address and data are multiplexed (AD0-AD7)
- It works on +5V DC power supply
- It has a maximum clock frequency of 3MHz where the minimum frequency is 500 KHz
- It has five addressing modes
- It can address up to 64 k memory locations
- Its instruction set consisting of 72 instructions
- To address an I/O or peripheral device it uses both memory-mapping as well as IO mapped IO. It uses 16 bit addresses for memory-mapped IO and 8 bit addresses for IO mapped IO
- It is a CISC processor and uses five stage execution unit

8085 CPU- Architecture

1.3.1 PIN Diagram and Architecture of 8085

The 8085 microprocessor is an IC having 40 pins. The layout of which is shown in Fig. 1.6

and Fig.1.7. The signals which are connected to these pins can be classified into six groups. These are

- Address bus
- Data bus
- Control & status signals
- Power supply and frequency signals
- Externally initiated signals
- Serial I/O signals

PIN Diagram Of 8085 MP

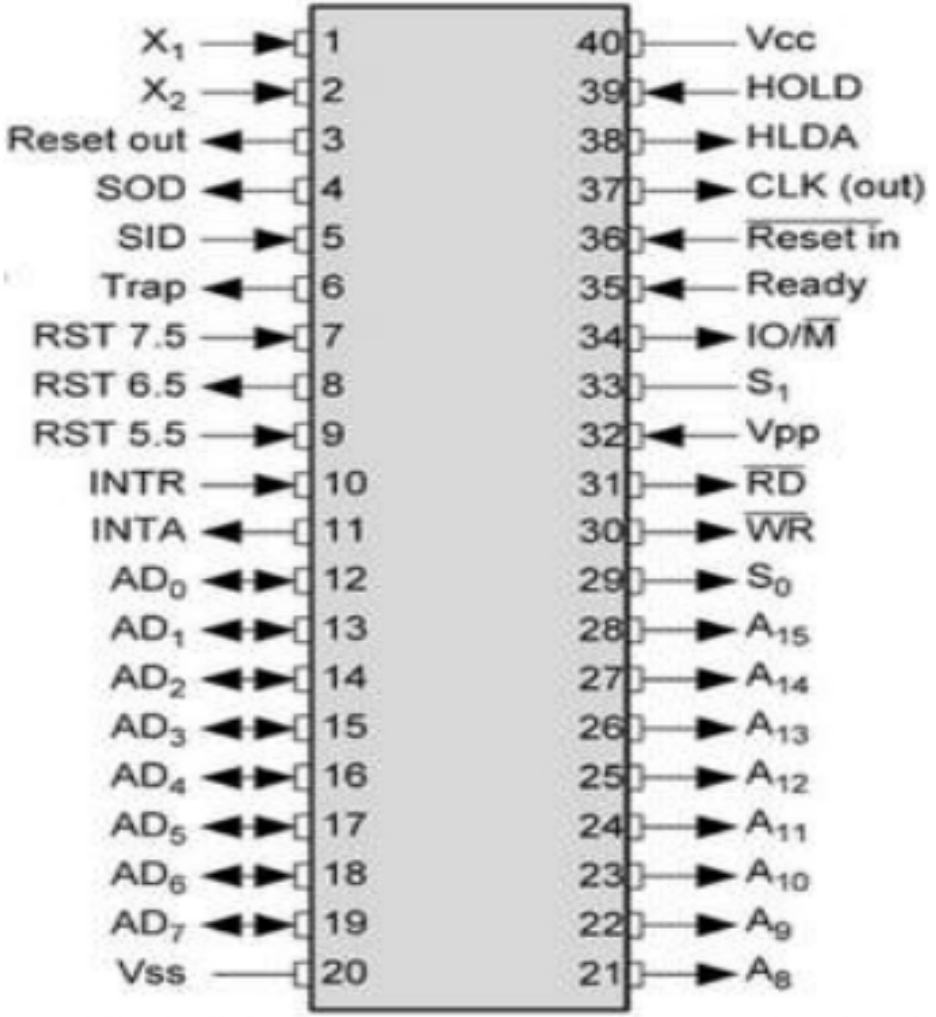


Figure 1.6: 8085 Microprocessor Pin layout and associated signals

PIN Diagram Of 8085 MP

- **1.3.1.1 The address bus and Data bus**
- Address bus is 16-bit wide and consists of 16 signal lines/wires for communication.
- Higher order address bus consists of bit lines (A8-A15) which is unidirectional.
- Usually goes to tri-state/high impedance state during HOLD and HALT mode. Lower order address consists of bit lines (A0-A7) is multiplexed with data bus. Thus, we have multiplexed address/data bus (AD0-AD7) and is bidirectional data bus. Thus, we have multiplexed address/data bus (AD0-AD7) and is bidirectional.
- It behaves as address bus during first clock cycle. In the subsequent clock cycles (3rd and 4th) it acts as data bus.
- These 8 signal lines goes to tri-state/high impedance state in HOLD and HALT mode.

PIN Diagram Of 8085 MP

1.3.1.2 Control and Status signals

This group consists of the following signals,

- ALE : Address Latch Enable
- *RD* : Read control signal
- *WR* : Write control signal
- IO/*M*, S1 and S0 : Status signal

ALE occurs during the first clock cycle of a machine state and enables the address to get

latched. The falling edge of ALE guarantee the setup and hold times for the address

information. ALE can also be used to strobe the status information. It is never tri-stated.

PIN Diagram Of 8085 MP

- IO/*M*is a status signal used to differentiate between I/O and memory operations. When this signal goes high it indicates an I/O operation and when it is low indicates a memory operation.
- This signal is usually combined with RD and WR signals to generate control signals for I/O and memory.
- S1and S0 are the status signals similar to IO/*M*, however, they are rarely used in small systems.

1.3.1.3 Power Supply & Clock Frequency Signals

Again, this group consists of four signals,

- Vcc : +5 V DC power supply
- Vss :Ground
- X1, X2 : Crystal Oscillator with a frequency of 6 MHz is connected to these two pins
- CLK : Clock output

PIN Diagram Of 8085 MP

1.3.1.4 Externally Initiated Signals and Interrupts

- *RESET IN* When the signal on this pin goes low, the PC is set to 0 and the buses are tri-stated and the processor is reset.
- RESET OUT: This signal indicates that the processor is in reset state. The signal can be used to reset other devices.
- READY: When this signal goes low, the processor waits for an integral number of clock cycles until it goes high.
- HOLD: This signal indicates that a peripheral such as DMA (direct memory access) controller is requesting for the use of address and data bus
- HLDA: This signal acknowledges the HOLD request
- INTR: Interrupt request is a general-purpose interrupt

PIN Diagram Of 8085 MP

- *INTA* This active low signal is used to acknowledge an interrupt
- RST 7.5, RST 6.5, RST 5,5 – restart interrupt: These are vectored interrupts and

have the higher priority than other interrupts i.e. INTR

- TRAP: This is a non-maskable interrupt and has the highest priority 1.3.1.5 Serial I/O Signals This group consists of only two signals,
- SID: Serial input signal. Bit on this line is loaded to D7 bit of register A using RIM instruction.
- SOD: Serial output signal. Output SOD is set or reset by using SIM instruction

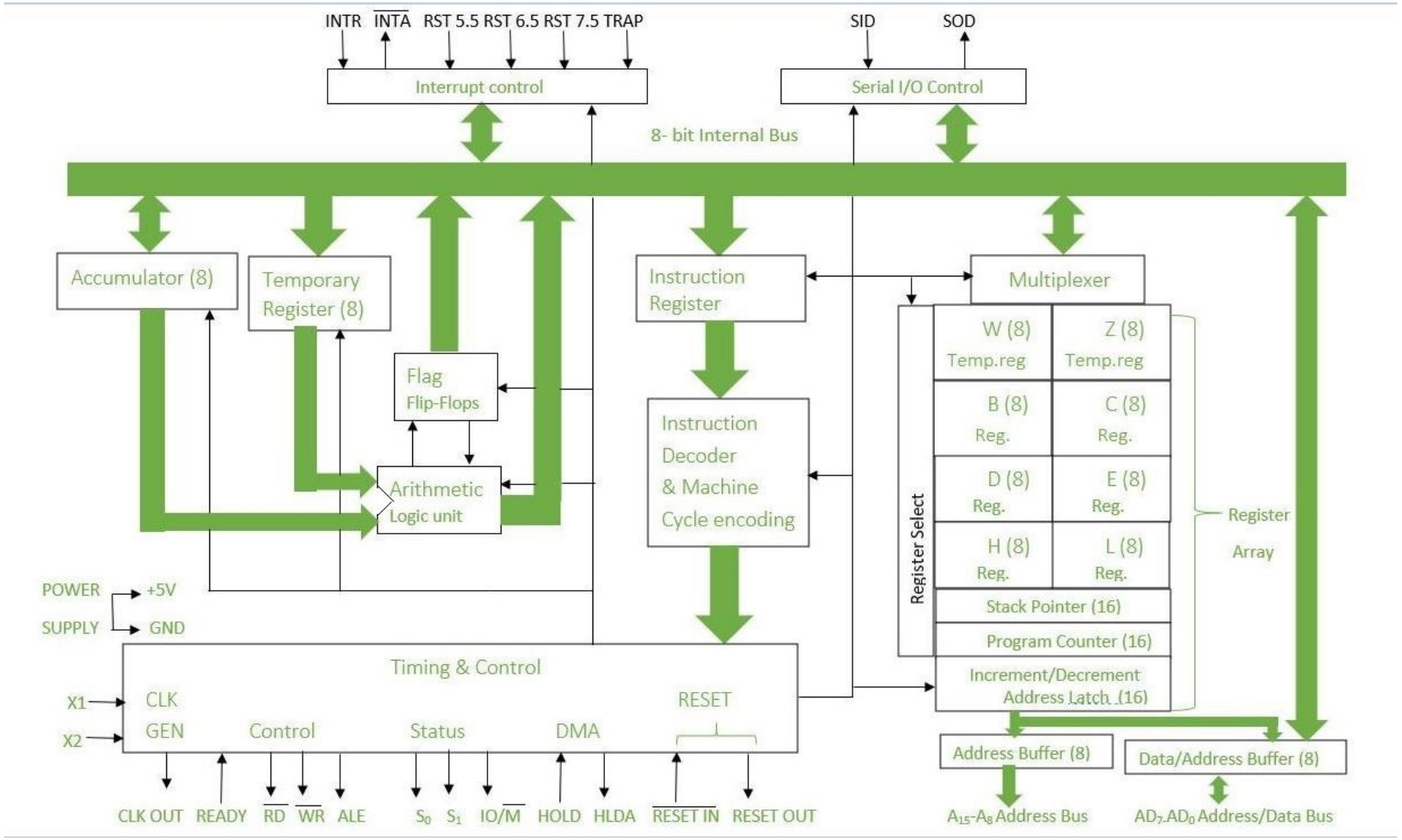
Architecture of 8085 microprocessor

- A microprocessor’s architecture is often specified with its instruction set, addressing modes, data types and their formats, design of its CPU, main memory and IO subsystem.
- The logic design of the microprocessor is usually called as the architecture.
- Microprocessor is a programmable logic device designed with registers, flipflops and timing circuitries.
- It has a set of instruction to manipulate data and communicate with peripherals
- The microprocessor can be programmed to perform various tasks on a given data by selecting necessary instruction from its set

Architecture of 8085 microprocessor

- It can also respond to external signals. Various functions of microprocessor can be classified into:
- Microprocessor-initiated operations
- Internal operations
- Peripheral or externally initiated operations

Architecture of 8085 microprocessor



Architecture of 8085 microprocessor

- **The ALU**
- The arithmetic/logic unit or the ALU performs various computing functions.
- These are basically the arithmetic operation such as addition, subtraction and logic operations such as AND, OR, XOR. For data manipulation, it takes help of Accumulator and a temporary register which are considered as a part of ALU.
- **The Register Array**
- It is consisting of various registers identified as B, C, D, E, H, L. These are primarily used to store data temporarily during program execution and are accessible to the user through instructions.

Architecture of 8085 microprocessor

- They can be used singly either to store 8-bit data or can be used in pair (such as BC, DE or HL) for 16 bit-operations.
- The register pair HL is also used as data pointer (holds memory address).
- The 8085 microprocessor also has an accumulator and a temporary register for the data processing by the ALU, which not accessible by the user.
- The accumulator is used to store 8-bit data and also to store the result of an ALU operation.
- **Flag Registers**
- The ALU also contains a 8-bit flag register to accommodate 05 flags

Architecture of 8085 microprocessor

- These flags set or reset according to the program operation, i.e. data condition in A or other registers. These are
- Zero (Z) : it is set if the result of an arithmetic operation is zero
- Carry (CY) : is set if there is a carry/borrow after arithmetic operation
- Sign (S) : is used to indicate the sign of data in the accumulator
  - is set to 1 if negative and set to 0 (reset) if positive
- Parity (P) : is set if the number is even and reset if the number is odd
- Auxiliary Carry (AC): is set if there is a carry out from bit 3

positionMost commonly used flags are-Zero, Carry and Sign. Microprocessor uses these flags to test data conditions before jumping

Architecture of 8085 microprocessor

- Most commonly used flags are-Zero, Carry and Sign. Microprocessor uses these flags to test data conditions before jumping.
- **The Control Unit**
- It provides necessary timing and control signals to all the operations. Also control flow of data between microprocessor and main memory/peripherals.
- A bit pattern usually called micro_x0002_program initiates execution of an instruction.
- By setting a sequence of control signal, it selects appropriate logic circuits in ALU and performs the task. Control signals are communicated through the control bus

Architecture of 8085 microprocessor

- **Program Counter and SP**
- The program counter (PC) is a16 bit registers to hold memory addresses. Memory addresses are of 16-bit. The PC is used to sequence the execution of instruction. It points to memory address where from next byte to be fetched.
- The stack pointer or SP is a memory pointer to point memory location in R/W memory, called stack.
- It is also a 16-bit register. Beginning of stack is defined by loading a 16-bit address in SP. Stack is an area of memory used to hold data that will be retrieved soon. The stack is usually accessed in a Last in First out (LIFO) fashion.

Architecture of 8085 microprocessor

- **Instruction Register (IR) and Decoder**
- These are some non-programmable registers. Instructions are stored in IR after being fetched by the processor. Decoder decodes the instruction in IR.

Addressing Modes

- **Addressing Modes**
- It specifies the way to represent the data to be operated on by the instruction. In other words, the addressing modes indicates the formats of specifying the operands.
- The 8085 microprocessor has the following five different types of addressing modes.

- Immediate addressing
- Memory direct/Direct addressing
- Register direct addressing
- Indirect addressing
- Implicit (or implied) addressing

Addressing Modes

- **Immediate Addressing**
- In immediate addressing the operand is present in the instruction itself. The operand (either a byte or a word) will be either transferred to a register or a memory location. For example, MVI A, 32H : means the immediate operand 32H is to be transferred to the register A.
- MOV A, #6AH: means copy the immediate operand 6AH to accumulator.
- The format of immediate addressing is shown in Fig. 1.9a, while program memory and immediate data, how is being transferred to accumulator is shown in Fig. 1.9b.

Addressing Modes

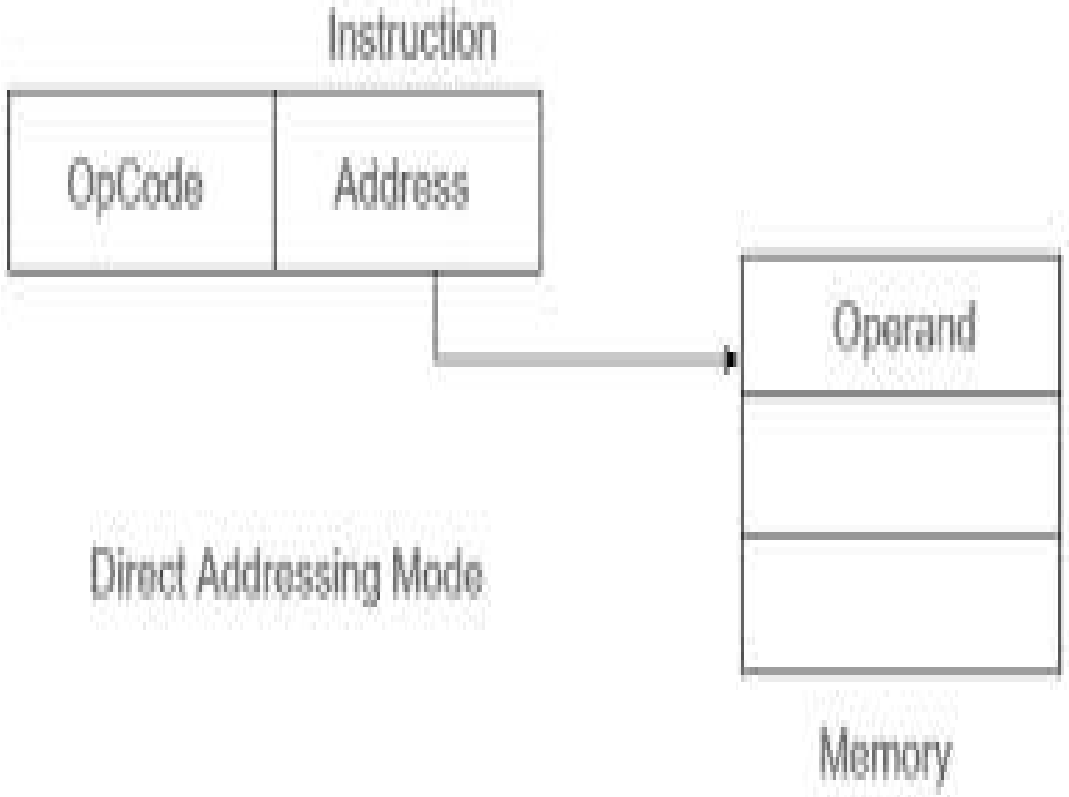


Addressing Modes

- **Memory direct/ Direct addressing**
- This type of addressing indicates that data transfer operation is direct. The address of a memory
- location where the operand is stored is directly specified in the instruction. For example,
- LDA 2051H : it means load the content of the memory location specified by 2051H into
- accumulator register (A). A pictorial representation of this mode of operation is shown below

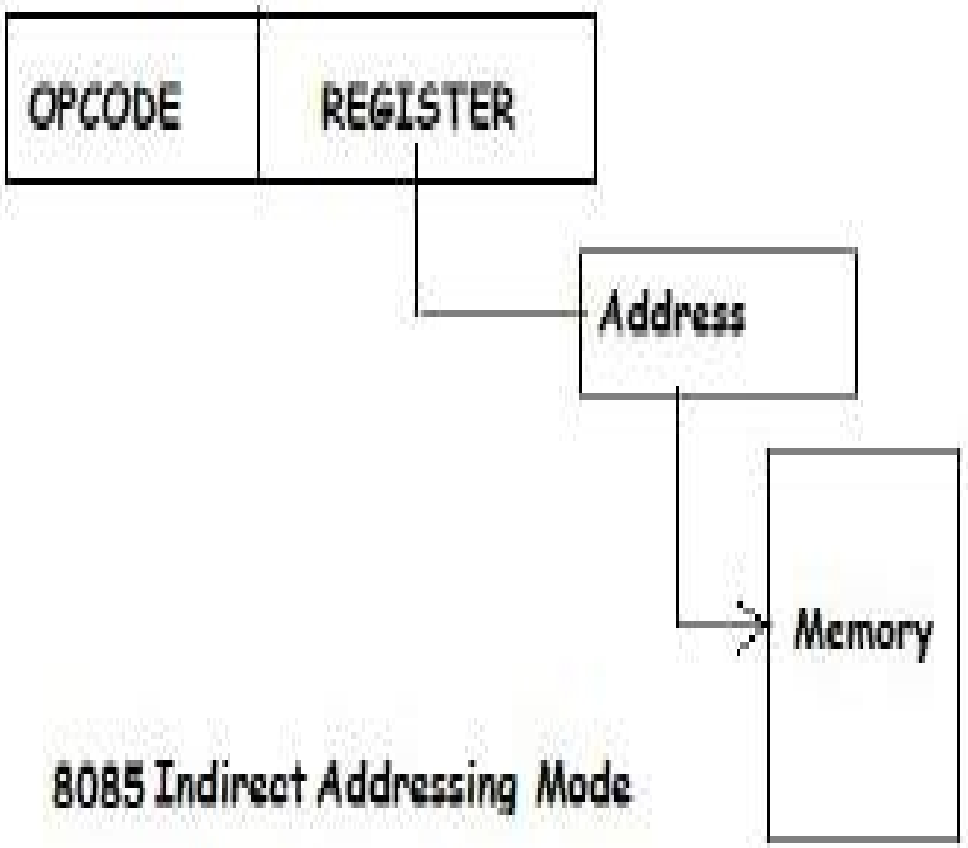
Addressing Modes

- **Register Direct Addressing**
- This type of addressing refers to transfer of the data byte or a word directly from one register
- to the other. For example, MOV A, C: it means copy the content of register C to register A.
- Fig.1.11 depicts the pictorial representation of this mode of operation



Addressing Modes

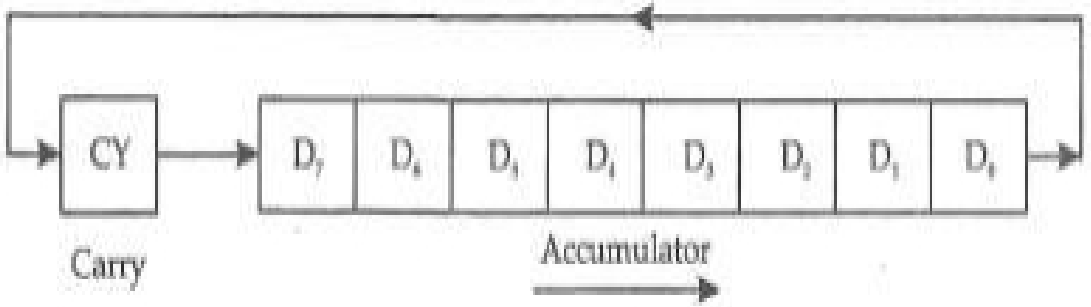
- **Indirect Addressing**
- Indirect addressing refers to data transfer (byte or a word) between a register and a memory location specified by register pair. For example, MOV B, M : copy the data byte into register B from the memory location specified by the address in the register pair HL. Fig.1.12 represents indirect mode of addressing



8085 Indirect Addressing Mode

Addressing Modes

- **Implied or Implicit Addressing**
- In this type of addressing no operand is needed. The operand is implicitly present there in the opcode/ instruction itself. For example,
- CMA : complement the content of the accumulator,
- RAL : Rotate the content of accumulator to the left
- RLC : Rotate the accumulator content left through carry. Following picture illustrates the same.



Instruction Set

- **Instruction Set**
- Instruction is a binary pattern or a format to perform a specific task by the processor. The entire group of instructions that are supported by a microprocessor is known as the instruction set.
- The 8085 microprocessor has a total of 74 opcodes that result in 246 instructions (with all variants). An instruction has two parts-the opcode and the operand.
- Opcode specifies the function to be executed and operand indicates the data to be operated on. **These instructions can be broadly classified into three groups.**

- Data transfer or data movement instructions
- Data Processing (ALU operations)
    - o Arithmetic instructions
    - o Logic instructions
- Program Control
    - o Branch instruction
    - o Machine control instructions

Instruction Set

- **Data Transfer Instructions**
- These are a group of instructions used to copy data from one location (source) to another (destination).
- Such an operation does not modify the source register content.
- Following are the various **types of instructions under data processing** depending upon the type of source and destination.

• **Instruction Types:**

– Between registers (example, B to D)

– A specific data byte to a register or a memory location (example, load B with 32H)

– Between a memory location and a register (example, from location 2000H to B)

– Between I/O device and accumulator (example: Keyboard to accumulator)

– Between registers and stack memory

Instruction Set

- **Data Processing (Arithmetic and Logic) Instructions**
- This group includes all the instructions needed for arithmetic operations and logical operations which comprise of the following instruction types.

• Arithmetic Operations

– Addition (any 8-bit number, contents of register or contents of memory location can be added with accumulator content)

– Subtraction (performed in 2's complement)

– Increment/Decrement: contents of a register or register pair can be incremented/decremented by 1

• Deals with one register or one location

• Logical Operations

– AND, OR, Exclusive-OR

• any 8-bit number, contents of register or contents of memory location

can be logically ANDed, ORed or XORed with accumulator content

– Rotate: each bit of accumulator can be shifted left or right

– Compare: for equality, greater than or less than

– Complement: Each bit of accumulator contents

Program Control Instructions

- **Program Control Instructions**
- This group includes all the instructions needed for branching operations as well as for machine control operations. These can be further classified as follows:
- Jump:
– Conditional Jump
- Test for certain condition (e.g. zero or carry flag)
– Unconditional jump
- Call, Return and Restart: change the sequence of instruction execution
– By calling a subroutine
– Returning from a subroutine
- Machine Control Operation
– Halt, Interrupt or do nothing

Instruction Timing diagram and Machine Cycle

- Each instruction in 8085 microprocessor consists of two part- operation code (opcode) and operand.
- **The opcode** is a command such as ADD and **the operand** is an object to be
- Any operation involving read/write operation or data transfer with the action of control signals, IO/*M*, S1 and S0 can be displayed in the form of a **timing diagram.**

- **A machine cycle** is the time taken by the microprocessor to complete the task of accessing memory or I/O devices.
- Various operations like opcode fetch, memory read, memory write, I/O read, I/O write etc. are performed in a machine cycle. Each cycle of the clock is known as **T-state**.
- **T-State**: Time corresponding to one clock period. It is the basic unit to calculate execution of instructions or programs in a processor.d

Instruction Timing diagram and Machine Cycle

- **Instruction Cycle:** The time taken by the processor to complete the execution of an instruction. An instruction cycle consists of one to six machine cycles.

8085 INTERRUPTS

- **Interrupt Structure:**
- Interrupt is the mechanism by which the processor is made to transfer control from its current program execution to another program having higher priority.
- The interrupt signal may be given to the processor by any external peripheral device
- The program or the routine that is executed upon interrupt is called **interrupt service routine (ISR)**.
- After execution of ISR, the processor must return to the interrupted program. The address of the memory where the ISR is located for a particular interrupt signal.
- This address is called I**nterrupt Vector Address (IVA)**.

8085 INTERRUPTS

Types of Interrupts:

Interrupts are classified based on their maskability, IVA and source. They are classified as:

i. Vectored and Non-Vectored Interrupts

Vectored interrupts In this type of interrupt, the interrupt address is known to the processor. For example: RST7.5, RST6.5, RST5.5, TRAP

Non-vectored interrupts In this type of interrupt, the interrupt address is not known to the processor so, the interrupt address needs to be sent externally by the device to perform interrupts. For example: INTR.s

8085 INTERRUPTS

ii. Maskable and Non-Maskable Interrupts

Maskable interrupts In this type of interrupt, we can disable the interrupt by writing some instructions into the program. For example: RST7.5, RST6.5, RST5.5

Non-maskable interrupts In this type of interrupt, we cannot disable the interrupt by writing some instructions into the program. For example: TRAP..

iii**. Software and Hardware Interrupts**

In this type of interrupt, the programmer has to add the instructions into the program to execute the interrupt. There are 8 software interrupts in 8085, i.e. RST0, RST1, RST2, RST3, RST4, RST5, RST6, and RST7.

8085 INTERRUPTS

- **Hardware interrupts** In this type of interrupt,it is caused by any peripheral device by sending a signal through a specified pin to the microprocessor
- There are 5 interrupt pins in 8085 used as hardware interrupts, i.e. TRAP, RST7.5, RST6.5, RST5.5, INTA.