Noida Institute of Engineering and Technology, Greater Noida

Introduction to Microprocessor

(ACSE0405)

Unit: 1

Microprocessor

B.Tech 4th Semester

Khushboo
Assistant Professor
ECE

Evaluation Scheme

Sl. No.	Subject Codes	Subject Name	Periods			Evaluation Scheme				End Semester		Total	Credit
			L	T	P	CT	TA	TOTAL	PS	TE	PE		
1	AAS0402	Engineering Mathematics-IV	3	1	0	30	20	50		100		150	4
2	AASL0401	Technical Communication	2	1	0	30	20	50		100		150	3
3	ACSE0405	Microprocessor	3	0	0	30	20	50		100		150	3
4	ACSE0403A	Operating Systems	3	0	0	30	20	50		100		150	3
5	ACSE0404	Theory of Automata and Formal Languages	3	0	0	30	20	50		100		150	3
6	ACSE0401	Design and Analysis of Algorithm	3	1	0	30	20	50		100		150	4
7	ACSE0455	Microprocessor Lab	0	0	2				25		25	50	1
8	ACSE0453A	Operating Systems Lab	0	0	2				25		25	50	1
9	ACSE0451	Design and Analysis of Algorithm Lab	0	0	2				25		25	50	1
10	ACSE0459	Mini Project using Open Technology	0	0	2				50			50	1
11	ANC0402 / ANC0401	Environmental Science*/ Cyber Security*(Non Credit)	2	0	0	30	20	50		50		100	0
12		MOOCs** (For B.Tech. Hons. Degree)											
		GRAND TOTAL										**1100**	**24**

Subject Syllabus

Course Contents / Syllabus		
UNIT-I	**8085 Microprocessor**	**8 Hours**
Introduction to Microprocessor, Microprocessor evolutionandtypes, Microprocessorarchitectureand its operation, Logic devices for interfacing,Pin diagram and internal architecture of 8085 Microprocessor,Example of an 8085 based computer, Instructionanddataflow,timerandtimingdiagram,interruptandmachinecycle, Addressingmodes.		
UNIT-II	**8085 Instructions and Programming Techniques**	**8 Hours**
Instructionsets,InstructionClassification:datatransfer operations, arithmeticoperations,logicaloperations,branchingoperations,machinecontrolandassemblerdirectives, writing assembly language programs, Programming techniques: looping, counting and indexing		
UNIT-III	**Code Conversion and BCD Arithmetic**	**8 Hours**
Counter and time delays, Illustrative program: Hexadecimal counter, zero-to-nine, (module ten) counter, generating pulse waveforms, Stack, Subroutine, Restart, Conditional call and return instructions, Advance subroutine concepts, Program: BCD-to-Binary conversion, Binary-to-BCD conversion, BCD-to-Seven segment code converter, Binary-to-ASCII and ASCII-to-Binary code conversion, BCD Addition, BCD Subtraction, Introduction to Advance instructions and Application, Multiplication		
UNIT-IV	**Interfacing of I/O devices**	**8 Hours**
Basic interfacing concepts,Memoryinterfacing,Interfacing output displays, Interfacing input devices, Memory mapped I/O,Interfacing keyboard and seven segment displays, The 8085 Interrupts, 8085 vector interrupts,8259 programmable interrupt controller,		
UNIT-V	**Programmable Peripheral IC's and 8086 Microprocessor**	**8 Hours**
Peripheral Devices: 8255 programmable peripheral interface,8253/8254 programmable timer/counter, 8237 DMA Controller, 8251 USART and RS232C.Introduction to 8086 microprocessors: Architecture of 8086 (Pin diagram, Functional block diagram, register organization), Addressing Modes		

Branch Wise Application

- **Microprocessor**, any of a type of miniature electronic device that contains the arithmetic, logic, and control circuitry necessary to perform the functions of a digital computer's central processing unit.
- This kind of integrated circuit can interpret and execute program instructions as well as handle arithmetic operations.

Course Objective

- The objective of this course is to understand basic concepts of Microprocessor based systems and able to do programming in Assembly Language of 8085. They will be able to learn and program Peripheral IC's.

Course Outcome

At the end of this course students will able to:

- Apply a basic concept of digital fundamentals to Microprocessor based personal computer system.
- Analyze a detailed s/w & h/w structure of the Microprocessor.
- Illustrate how the different peripherals (8085/8086) are interfaced with Microprocessor.
- Analyze the properties of Microprocessors (8085/8086).
- Evaluate the data transfer information through serial & parallel ports.

Program Outcomes

- **Program Outcomes** are narrow statements that describe what the students are expected to know and would be able to do upon the graduation.

- These relate to the skills, knowledge, and behavior that students acquire through the programmed.

1. Engineering knowledge
2. Problem analysis
3. Design/development of solutions
4. Conduct investigations of complex problems
5. Modern tool usage
6. The engineer and society
7. Environment and sustainability
8. Ethics
9. Individual and team work
10. Communication
11. Project management and finance
12. Life-long learning

CO-PO Mapping

Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
ACSE0405.1	3	1	2		1	-	-	-	-	-	-	3
ACSE0405.2	3	1	-		1	-	-	-	-	-	-	2
ACSE0405.3	3	1	3		1	-	-	-	-	-	-	2
ACSE0405.4	3	1	-		1	-	-	-	-	-	-	2
ACSE0405.5	3	1	2		1	-	-	-	-	-	-	2
Average	3	1	2.3		1	-	-	-	-	-	-	2.1

PSOs Mapping

Sr. No.	Course Outcome	PSO1	PSO2	PSO3
1	**ACSE0405.1**	3	2	1
2	**ACSE0405.2**	3	2	1
3	**ACSE0405.3**	3	2	1
4	**ACSE0405.4**	3	2	1
5	**ACSE0405.5**	3	2	2
	Average	**3**	**2**	**1.2**

End Semester Question Paper Template

B.Tech (Semester IV Theory Examination 2021-22)			**Total Marks : 100**
Note: Attempt all sections. If require any missing data, then choose suitably. **Time: 3 hours**			
Section A			
1. Attempt all questions in brief.		2 X 10 = 20	
Q. No.	Question	Marks	CO
a. to j		2	
Section B			
2. Attempt any three of the following		3 X 10 = 30	
Q. No.	Question	Marks	CO
a to e		10	
Section C			
Question no. 3,4,5,6,7. Attempt any one of the following		1 X 10 = 10	
a		10	
b		10	

Prerequisites

- Basic knowledge of digital logic gates
- Knowledge of Boolean Algebra
- Basic concept of Flip-Flops

Unit Content

- Course Objective
- Unit Objective
- Course Outcome
- CO and PO Mapping
- Topic Objective
- Prerequisite
- Introduction to Microprocessor
- Microprocessor evolution and types
- Microprocessor architecture and its operations
- Logic devices for interfacing
- Pin Diagram
- Internal architecture of 8085 Microprocessor
- Example of an 8085 based computer
- Instruction and data flow
- Timer and timing diagram
- Interrupt and Machine Cycle
- Addressing modes
- Daily quiz & MCQs
- Old Question Papers
- Recap
- Video Links
- Weekly Assignments
- References

Unit Objective

1. To study the Introduction of microprocessors.
2. To understand Memory and I/O devices.
3. To learn microprocessor 8085 architecture.
4. To discuss concept of Timing Diagram.
5. To acquire the knowledge of Addressing Modes.

Topic-1 Objective

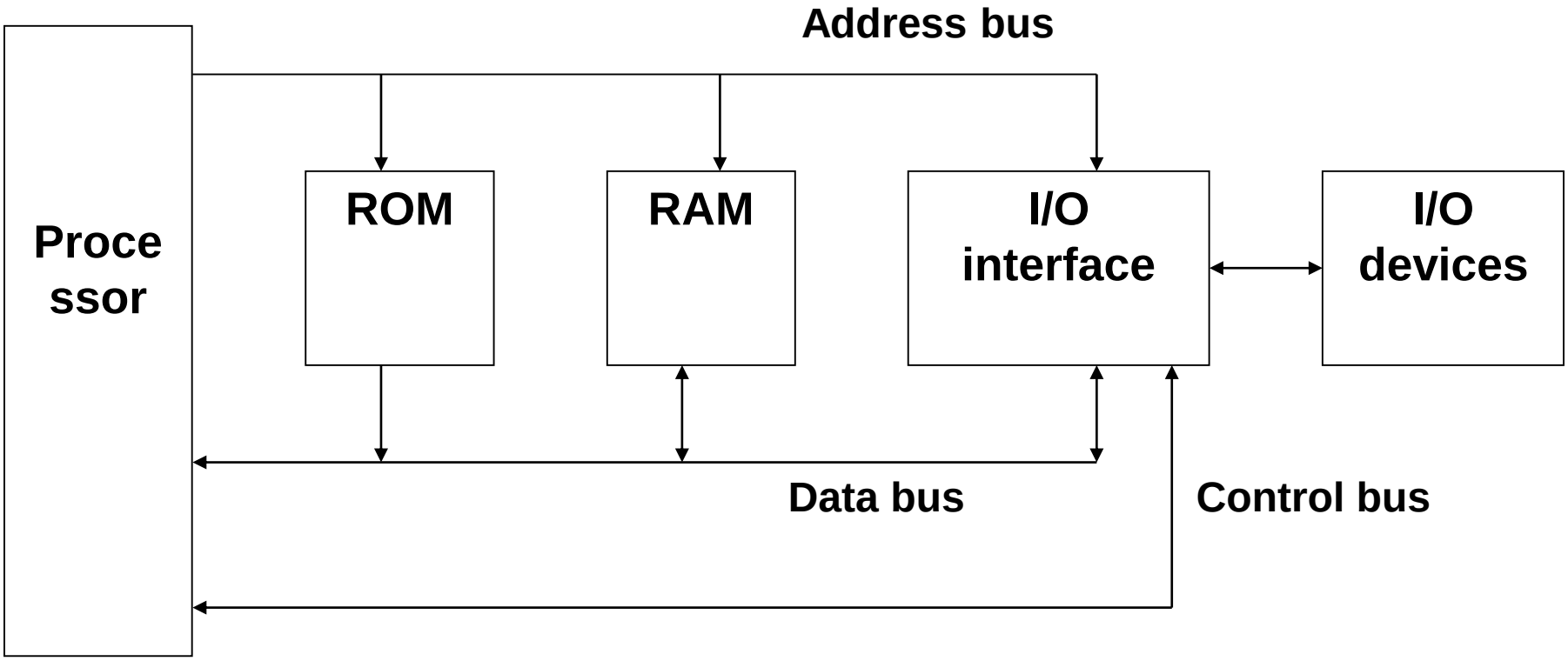
Name of the Topic	Objective of the topic	Mapping with CO
Introduction to Microprocessor	To understand the concept of microprocessor.	CO1

Introduction to Microprocessor

Block Diagram of a Basic Computer

A **computer** is a programmable machine that receives input, stores and manipulates data/information, and provides output in a useful format.

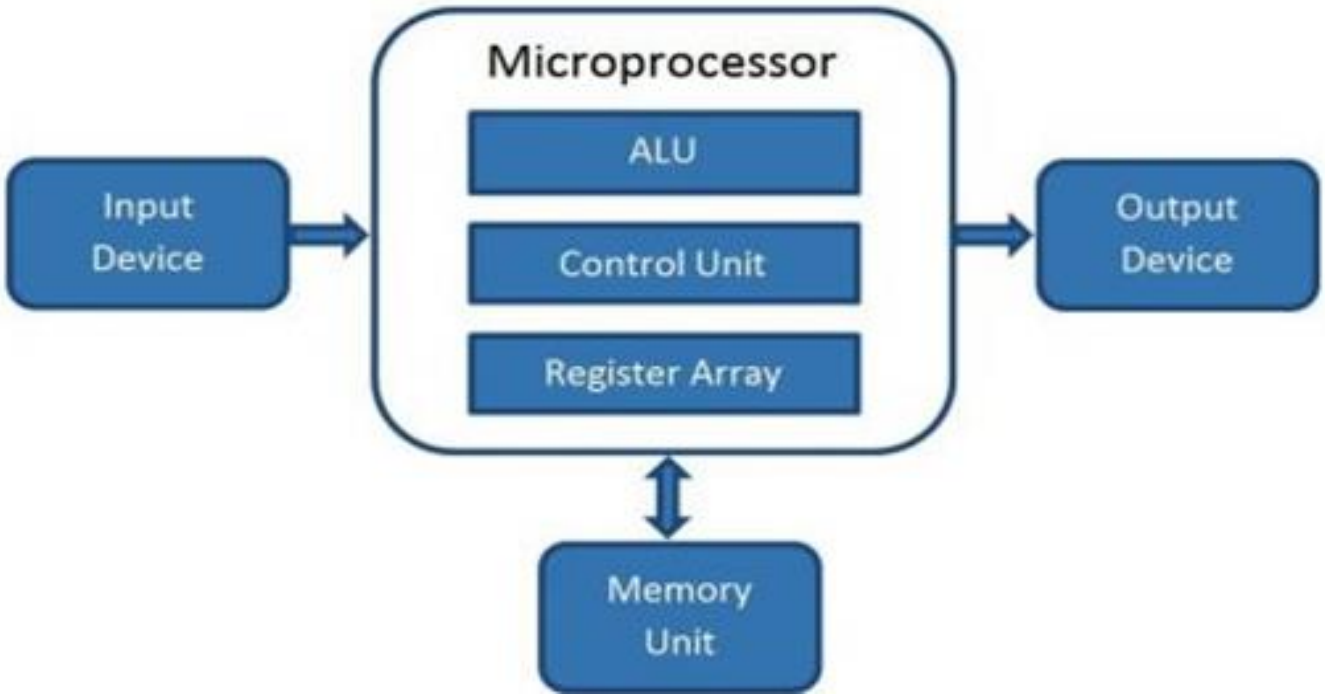
Basic computer system consist of a Central processing unit (CPU), memory (RAM and ROM), input/output (I/O) unit.



Introduction to Microprocessor

A microprocessor is a controlling unit of a micro-computer, fabricated on a small chip capable of performing Arithmetic Logical Unit (ALU) operations and communicating with the other devices connected to it.

Block Diagram of a Basic Microcomputer



Cont..

1. **CPU - Central Processing Unit**
   - The portion of a computer system that carries out the instructions of a computer program
   - The primary element carrying out the computer's functions. It is the unit that **reads and executes program instructions**.
   - The data in the instruction tells the processor what to do.
2. **Memory**
   - Physical devices used to store data or programs.
   - Computer main memory comes in two principal varieties: random-access memory (RAM) and read-only memory (ROM).
   - RAM can be read and written to anytime the CPU commands it, but ROM is pre-loaded with data and software that never changes, so the CPU can only read from it.
   - ROM is typically used to store the computer's initial start-up instructions.

   In general, the contents of RAM are erased when the power to the computer is turned off, but ROM retains its data indefinitely.

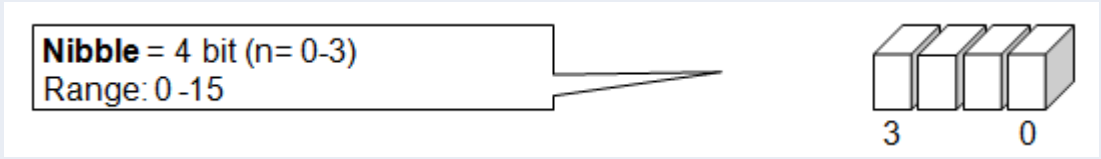
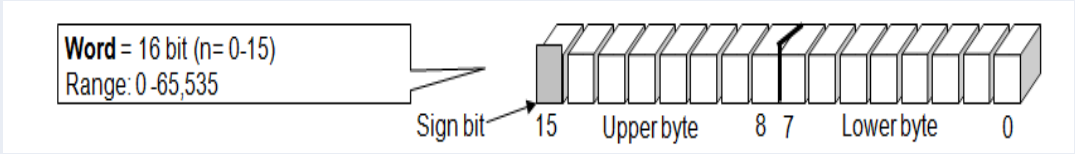
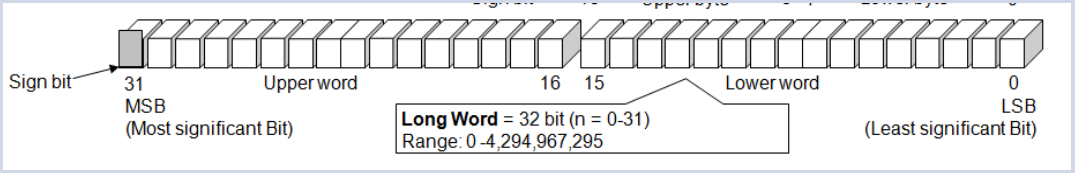
Cont..

3. I/O Unit

- **Input/output (I/O)**, refers to the communication between an information processing system (such as a computer), and the outside world possibly a human, or another information processing system.
- Inputs are the signals or data received by the system, and outputs are the signals or data sent from it
- Devices that provide input or output to the computer are called peripherals.
- On a typical personal computer, peripherals include input devices like the keyboard and mouse, and output devices such as the display and printer. Hard disk drives, floppy disk drives and optical disc drives serve as both input and output devices. Computer networking is another form of I/O.

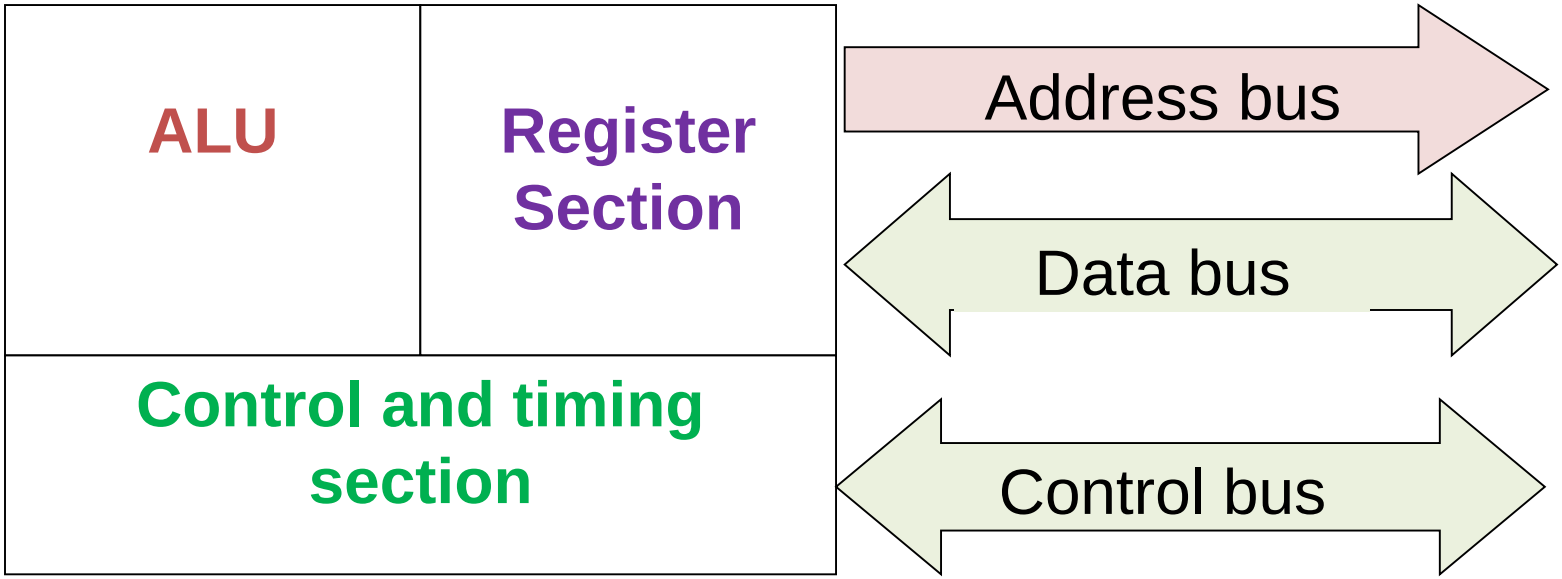
Cont..

Data Size

Nibble	**4 bit**	
Byte	**8 bit**	
Word	**16 bit**	
Long word	**32 bit**	

Cont..

Internal structure and basic operation of microprocessor



Block diagram of a microprocessor

Arithmetic and logic unit (ALU):

- The component that performs the arithmetic and logical operations
- The most important components in a microprocessor, and is typically the part of the processor that is designed first.
- Able to perform the basic logical operations (AND, OR), including the addition operation.

Cont..

Control unit:

- The circuitry that controls the flow of information through the processor, and coordinates the activities of the other units within it.
- In a way, it is the "brain within the brain", as it controls what happens inside the processor, which in turn controls the rest of the PC.

On a regular processor, the control unit performs the tasks of fetching, decoding, managing execution and then storing results.

Register sets:

- The register section/array consists completely of circuitry used to **temporarily store data or program codes until they** are sent to the ALU or to the control section or to memory.
- The number of registers are different for any particular CPU and the more register a CPU have will result in easier programming tasks.

Registers are normally measured by the number of bits they can hold, for example, an "8-bit register" or a "32-bit register".

Daily Quiz

Problems

1. Convert $(25)_{10}$ to binary number.
2. Convert $(128)_{10}$ to octal number.
3. Convert 128_{10} to hex.
4. Convert $(1101)_2$ into decimal number.
5. Convert 22_8 to decimal number.
6. Convert 121_{16} to decimal number.
7. Convert $(89)_{16}$ into a binary number.
8. Convert $(214)_8$ into a binary number.

Daily Quiz

Answers

1. $(25)_{10} = (11001)_2$
2. 200_8
3. 80_{16}
4. $(1101)_2 = (13)_{10}$
5. $22_8 = 18_{10}$
6. $121_{16} = 289_{10}$
7. $(89)_{16} = (10001001)_2$
8. $(214)_8 = (010001100)_2$

Weekly Assignment

- **Addition of two hexadecimal number.**
  1. 2345H+9854H
  2. 4A6H + 1B3H
  3. 182ABH + 5FCAA H
  4. A6CCH+ 4AC7H
  5. 90A6H + DF91H
  6. 1712H + 4753H
  7. AFC2H+ 4ACCH

- **Subtraction of two hexadecimal number.**
  1. 923H – 257H
  2. 4A6H - 1B3H
  3. 180BFH – FFDFH
  4. 7E6CH - 5DFAH
  5. 15DA2H - 884FH
  6. 1C582H - E706H

Topic Links

https://www.youtube.com/watch?v=Xl2nWDcy0To

https://slideplayer.com/slide/3944912/

https://www.bu.edu.eg/portal/uploads/Engineering,%20Shoubra/Electrical%20Engineering/3515/crs-14315/Files/Lec%201%20Intro%20to%20mp.ppt

Topic-2 Objectives

Name of the Topic	Objective of the topic	Mapping with CO
Microprocessor Evolution and Types	To understand the concept of microprocessor & its architecture.	CO1

Microprocessor Evolution

Intel 4004

Year of introduction 1971

- 4-bit microprocessor
- 4 KB main memory
- 45 instructions
- PMOS technology
- as first programmable device which was used in calculators

Intel 8008

Year of introduction 1972

- 8-bit version of 4004
- 16 KB main memory
- 48 instructions
- PMOS technology

Cont..

Intel 8080

Year of introduction 1973

- 8-bit microprocessor, 64 KB main memory
- 2 microseconds clock cycle time
- 500,000 instructions/sec
- **10X faster than 8008**
- NMOS technology
- Drawback was that it needed **three power supplies**.
- Small computers (Microcomputers) were designed in mid 1970's using 8080 as CPU

Intel 8085

Year of introduction 1975

- 8-bit microprocessor-upgraded version of 8080
- 64 KB main memory
- 1.3 microseconds clock cycle time
- 246 instructions
- Intel sold 100 million copies of this 8-bit microprocessor
- uses **only one +5V power supply.**

Cont..

Intel 8086/8088

Year of introduction 1978 for 8086 and 1979 for 8088

- **16-bit microprocessors**
- Data bus width of 8086 is 16 bit and 8 bit for 8088
- **1 MB main memory**
- **400 nanoseconds clock cycle time**
- 6 byte instruction **cache** for 8086 and 4 byte for 8088
- Other improvements included **more registers and additional instructions**
- In 1981 IBM decided to use 8088 in its personal computer

Intel 80386

- Year of introduction 1986
- Intel's first practical **32-bit microprocessor**
- **4 GB main memory**
- Improvements include page handling in virtual environment
- Includes hardware circuitry for memory management and memory assignment
- Memory paging and enhanced I/O permissions

Cont..

Pentium

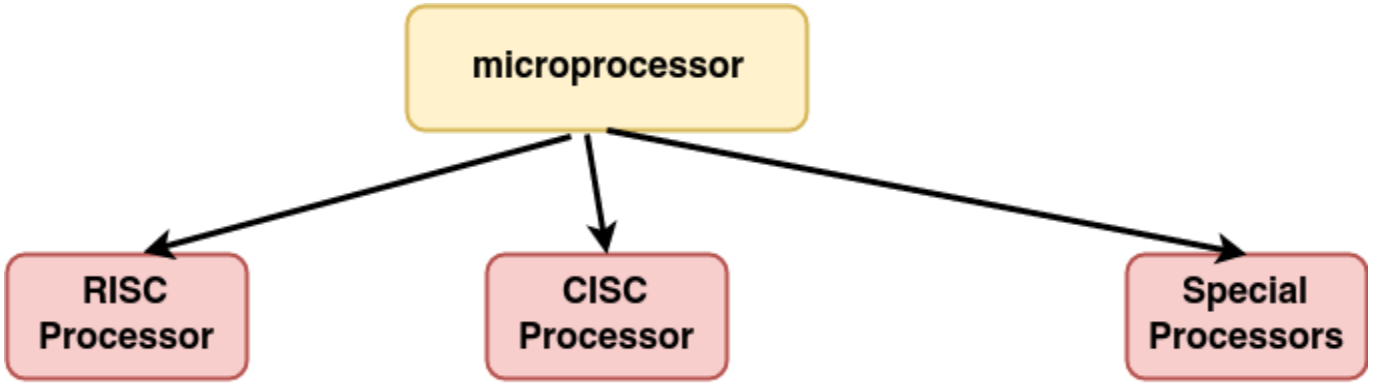
Year of introduction 1993

- 32-bit microprocessor, 32/64-bit data bus and 32-bit address bus
- 4 GB main memory, Double clocked 120 and 133MHz versions
- Fastest version is the 233MHz, Dual integer processor
- 16 KB L1 cache (split instruction and data: 8 KB each)

Processor	Year Of Introduction	No. Of Transistors	Intial Clock Speed	Address Bus	Data Bus(in bit)	Addressable Memory
4004	1971	2300	108 kHz	10 bit	4	640 bytes
8008	1972	3500	200 kHz	14 bit	8	16 k
8080	1974	6000	2 MHz	16 bit	8	64 k
8085	1976	6500	5 MHz	16 bit	8	64 k
8086	1978	29000	5 MHz	20 bit	16	1 M
8088	1979	29000	5 MHz	20 bit	8	1 M
80286	1982	134000	8 MHz	24 bit	16	16 M
80386	1985	275000	16 MHz	32 bit	32	4 G
80486	1989	1.2 M	25 MHz	32 bit	32	4 G
Pentium	1993	3.1 M	60 MHz	32 bit	32/64	4 G
Pentium Pro	1995	5.5 M	150 MHz	36 bit	32/64	64 G
Pentium II	1997	8.8 M	233 MHz	36 bit	64	64 G
Pentium III	1999	9.5 M	650 MHz	36 bit	64	64 G
Pentium 4	2000	42 M	1.4 GHz	36 bit	64	64 G

Microprocessor Types

A microprocessor can be classified into three categories:



RISC Processor

RISC stands for **Reduced Instruction Set Computer**. It is designed to reduce the execution time by simplifying the instruction set of the computer. Using RISC processors, each instruction requires only one clock cycle to execute results in uniform execution time. This reduces the efficiency as there are more lines of code, hence more RAM is needed to store the instructions. The compiler also has to work more to convert high-level language instructions into machine code.

Some of the RISC processors are:

- Power PC: 601, 604, 615, 620
- DEC Alpha: 210642, 211066, 21068, 21164
- MIPS: TS (R10000) RISC Processor
- PA-RISC: HP 7100LC

Cont..

Characteristics of RISC

The major characteristics of a RISC processor are as follows:

- It consists of **simple instructions**.
- It supports **various data-type formats**.
- It utilizes **simple addressing modes** and fixed length instructions for pipelining.
- It supports register to use in any context.
- **One cycle** execution time.
- "**LOAD" and "STORE**" instructions are used to access the memory location.
- It consists of **larger number of registers**.
- It consists of less number of transistors.

CISC Processor

CISC stands for **Complex Instruction Set Computer**. It is designed to minimize the number of instructions per program, ignoring the number of cycles per instruction.

The compiler has to do very little work to translate a high-level language into assembly level language/machine code because the length of the code is relatively short, so very little RAM is required to store the instructions.

Some of the CISC Processors are: IBM 370/168, VAX 11/780, Intel 80486

Cont..

Characteristics of CISC

- **Variety of addressing modes**.
- **Larger number of instructions** available.
- **Variable length of instruction** formats.
- Several cycles may be required to execute one instruction.
- **Instruction-decoding** logic is **complex**.
- One instruction is required to support **multiple addressing modes**.

Cont..

Special Processors

These are the processors which are designed for some special purposes. Few of the special processors are briefly discussed:

Coprocessor:

A coprocessor is a specially designed microprocessor, which can handle its particular function many times faster than the ordinary microprocessor.

For example: Math Coprocessor.

Some Intel math-coprocessors are:

- 8087-used with 8086
- 80287-used with 80286
- 80387-used with 80386

Input/Output Processor:

It is a specially designed microprocessor having a local memory of its own, which is used to control I/O devices with minimum CPU involvement.

For example:

- DMA (direct Memory Access) controller
- Keyboard/mouse controller
- Graphic display controller

Cont..

DSP (Digital Signal Processor):

This processor is specially designed to process the analog signals into a digital form. This is done by sampling the voltage level at regular time intervals and converting the voltage at that instant into a digital form. This process is performed by a circuit called an analogue to digital converter, A to D converter or ADC.

Its applications are:

- Sound and music synthesis, Audio and video compression, Video signal processing, 2D and 3d graphics acceleration.

For example: Texas Instrument's TMS 320 series, e.g., TMS 320C40.

Daily Quiz

- Which of the following is a type of microprocessor?
  a) CISC
  b) RISC
  c) EPIC
  d) **All of the mentioned**
- Which of the following technology was used by Intel to design its first 8-bit microprocessor?
  a) NMOS
  b) HMOS
  c) **PMOS**
  d) TTL
- Which of the following is true about microprocessors?
  a) It has an internal memory
  b) It has interfacing circuits
  c) **It contains ALU, CU, and registers**
  d) It uses Harvard architecture

Daily Quiz

- What is Microprocessor?
  a) **A multipurpose PLD that accepts binary data as input**
  b) A multipurpose PLD that accepts an integer as input
  c) A multipurpose PLD that accepts whole numbers as input
  d) A multipurpose PLD that accepts prime numbers as input

- Which of the following is not true about 8085 microprocessor?
  a) It is an 8-bit microprocessor
  b) It is a 40 pin DIP chip
  c) **It is manufactured using PMOS technology**
  d) It has 16 address lines

Weekly Assignment

1. What is Microprocessor?
2. What are the different types of Microprocessor?
3. Explain the evolution of Microprocessor.
4. How many generation of Microprocessors are there?
5. Which is the fastest speed of microprocessor?

Topic Links

https://www.youtube.com/watch?v=cylTOKGdFJI

https://www.youtube.com/watch?v=mU7wpeThu7A

Topic-3 Objective

Name of the Topic	Objective of the topic	Mapping with CO
Microprocessor architecture and its operations	To understand the concept of microprocessor & its architecture.	CO1

Microprocessor Architecture

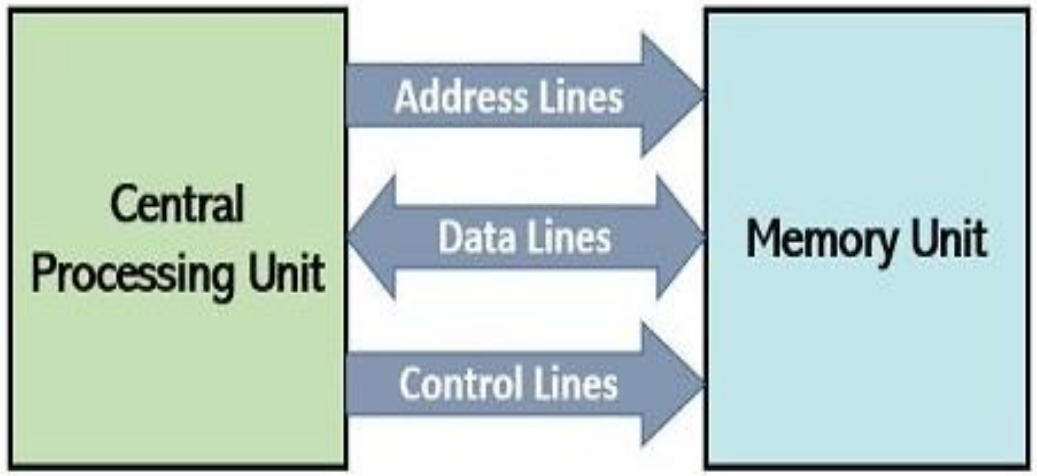
The microprocessor is a programmable logic device, designed with registers, flip-flops. The microprocessor has a set of instructions designed internally, to manipulate data and communicate with peripherals. This process of data manipulation and communication is determined by the logic design of the microprocessor, called the architecture.

Von Neumann Architecture

- In 1945, John von Neumann, who was a mathematician at the time, had delved into the study that, a computer could have a fixed simple structure and still be able to execute any kind of computation without hardware modification. This is providing that the computer is properly programmed with proper instructions, in which it is able to execute them.
- The von Neumann architecture describes a design model for a stored program digital computer that incorporates only **one single processing unit** and one single separate storage structure, which will **hold both instructions and data**.
- The von Neumann architecture refers to one that keeps the data as well as the programmed instructions **in read-write RAM** (Random Access Memory).

Cont..

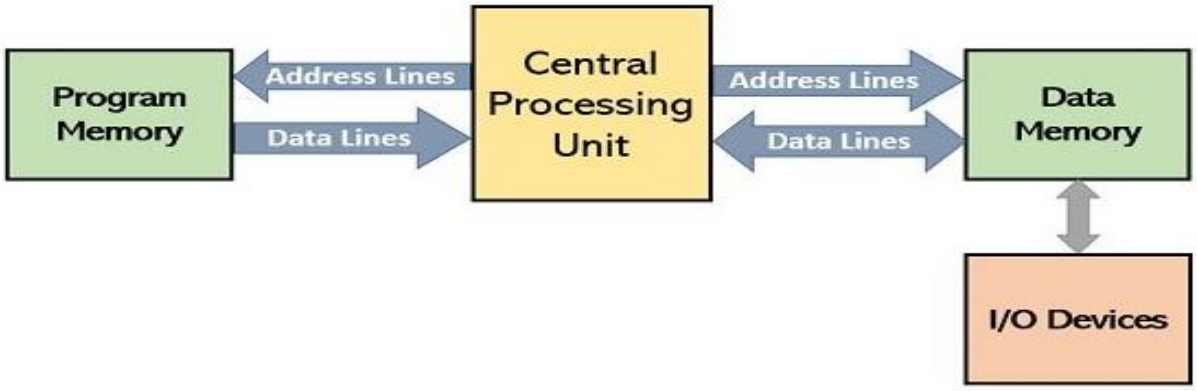
Von Neumann Architecture



Cont..

Harvard Architecture

- Harvard architecture is named after the "Harvard Mark I" relay-based computer, which was an IBM computer in the University of Harvard.
- The computer-stored instructions on "punched tape" (24 bits wide), furthermore the data was stored in electro mechanical counters.
- The CPU of these early computer systems contained the data storage entirely, and it provided no access to the instruction storage as data.
- Harvard architecture is a type of architecture, which stores the data and instructions separately, therefore splitting the memory unit.
- The CPU in a Harvard architecture system is enabled to fetch data and instructions simultaneously, due to the architecture having separate buses for data transfers and instruction fetches.



Cont..

Difference between Harvard Architecture & Von Neumann Architecture

Basis for Comparison	Von Neumann Architecture	Harvard Architecture
Basic	Data and instructions reside within a single memory unit.	Data and instruction are provided 2 different memory units.
Based on	Stored program computer concept	Harvard Mark I relay based model
Memory system	Single	Dual
Required space	Less	Comparatively more
Set of address/ data bus	One	Two
Development cost	Low	Comparatively more
Efficiency	Less	More
Execution speed	Slow	Comparatively fast

Cont..

Difference between Harvard Architecture & Von Neumann Architecture

Basis for Comparison	Von Neumann Architecture	Harvard Architecture
Operation	Simple	Complex
Performance offered	Low	Comparatively high
Clock cycle	Single instruction is executed in minimum two clock cycles.	Single instruction is executed in one clock cycle.
Feature	Data transfer and instruction fetching do not occur simultaneously.	Data transfer and instruction fetch take place at the same time.
Space utilization	Good	Not so good
Applications	PCs, workstations, notebooks, etc.	Microcontrollers, digital signal processing, etc.

Operations

3. Peripheral or Externally-Initiated Operations

- These are operations initiated by external devices.
- These are usually :
  - Reset
  - Interrupt
  - Ready
  - Hold

Daily Quiz

1. There are _____ general purpose registers in 8085 processor
   a) 5
   b) 6
   c) 7
   d) 8
2. Flag register is an 8-bit register having _____ 1-bit flip-flops.
   a) 3
   b) 4
   c) 5
   d) 6
3. What is true about Program counter?
   a) It is an 8-bit register, which holds the temporary data of arithmetic and logical operations.
   b) When an instruction is fetched from memory then it is stored in the program counter.
   c) It provides timing and control signal to the microprocessor
   d) It is a 16-bit register used to store the memory address location of the next instruction to be executed.

Daily Quiz

4. The microprocessor _____ the instructions from the memory.
   - **A. Fetch**
   - B. Decode
   - C. Execute
   - D. None of the above
5. An 8-bit microprocessor can process _____ data at a time.
   - A. 4-bit
   - B. **8-bit**
   - C. 16-bit
   - D. All of the above
6. There are primarily two types of register:
   - A. general purpose register
   - B. dedicated register
   - **C. A and B**
   - D. none of these

Weekly Assignment

1. Differentiate Von Neuman and Harvard Architecture.
2. Which architecture is used in microprocessor and explain it.
3. What are the different types of Flags in 8085?
4. Explain Microprocessor and its operations.
5. What is the length of stack pointer in 8085 microprocessor? And what is its use?

Topic Links

https://www.slideshare.net/RamaPrabha24/8085-microprocessor-architecture-and-its-operations

https://www.javatpoint.com/microprocessor-architecture

Topic-4 Objectives

Name of the Topic	Objective of the topic	Mapping with CO
Logic devices for interfacing	To understand the concept of interfacing with 8085 microprocessor.	CO1

Logic Devices for Interfacing

Logic Devices consists of :

- Tri-State devices
- Buffer
- Bidirectional Buffer
- Decoder
- Encoder
- D Flip Flop : Latch and Clocked

Cont..

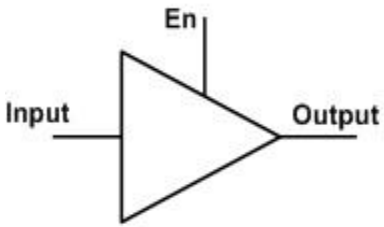
Tri-State devices:

Tri-state logic devices have three states:

1. Logic 1 or High
2. Logic 0 or Low
3. High impedance

- A tri-state logic device has a extra input line called **Enable**.
- When this line is active (Enabled), a tri-state device functions in the same way as ordinary logic devices.
- When this line is not active (disabled), the logic device goes into a high impedance state, as if it is disconnected from the system and practically no current is drawn from the system.

Symbol



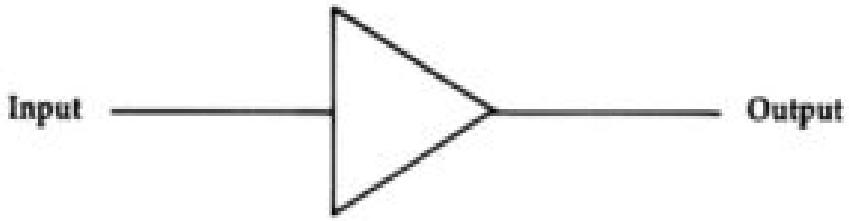
Truth Table

En	Input	Output
0	X	Hi-Z
1	0	0
1	1	1

Cont..

Buffer

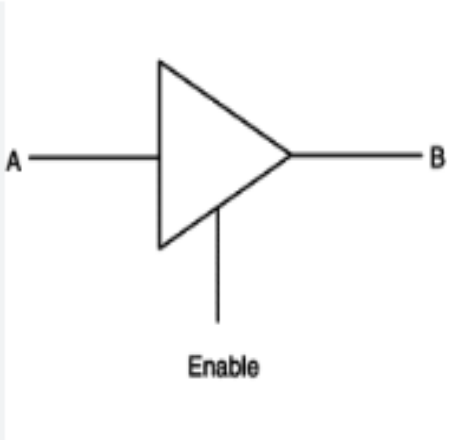
- A **Digital Buffer** is a single input device that does not invert or perform any type of logical operation on its input signal.
- In other words, the logic level of the output is same as that of the input.
- The buffer is a logic circuit that amplifies the current or power.
- The buffer is used primarily to increase the driving capability of a logic circuit.
- It is also known as driver.



Cont..

Tri-state Buffer

- A Tri-state Buffer can be thought of as an input controlled switch which has an output that can be electronically turned "ON" or "OFF" by means of an external "Enable" signal input.
- This Enable signal can be either a logic "0" or a logic "1" type signal.
- When Enable line is high (logic “1“), the circuit functions as a buffer.

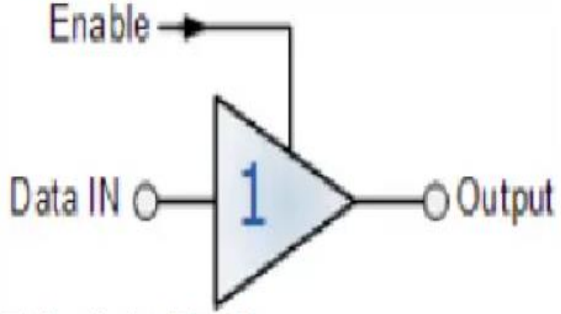


Enable	A	B
0	0	Z
0	1	Z
1	0	0
1	1	1

Cont..

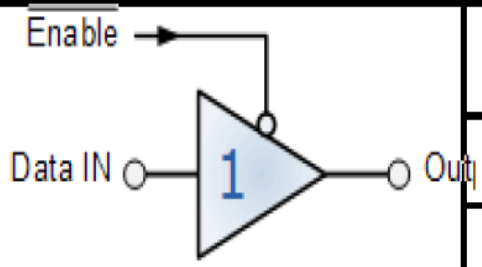
- There are two different types of Tri-state Buffer, one whose output is controlled by an "**Active-HIGH**" Enable signal and the other which is controlled by an "**Active-LOW**" Enable signal, as shown below

Active "HIGH" Tri-state Buffer

Symbol	Truth Table		
 Tri-state Buffer	Enable	A	Q
	1	0	0
	1	1	1
	0	0	Hi-Z
	0	1	Hi-Z
Read as Output = Input if Enable is equal to "1"			

Cont..

Active "LOW" Tri-state Buffer

Symbol	Truth Table		
 **Tri-state Buffer**	Enable	A	Q
	0	0	0
	0	1	1
	1	0	Hi-Z
	1	1	Hi-Z
Read as Output = Input if Enable is **NOT** equal to "1"			

Cont..

Encoder

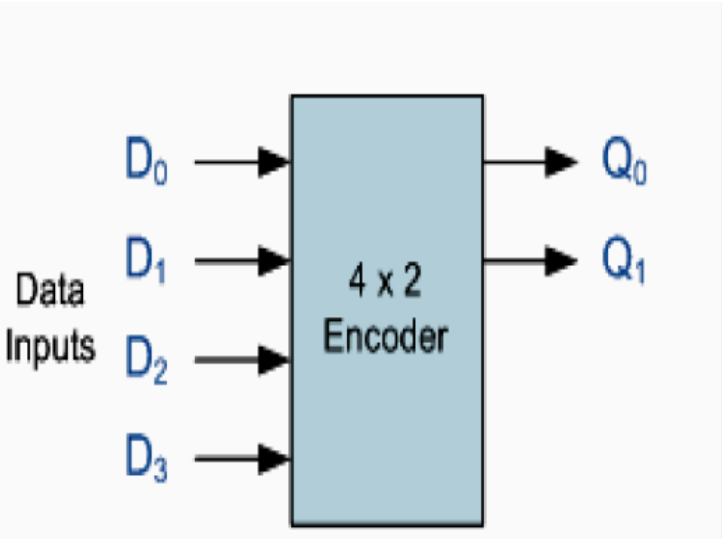
- The encoder is a logic circuit that provides the appropriate code (binary, BCD, etc.) as output for each input signal.

Binary Encoder

- A binary encoder, is a multi-input combinational logic circuit that converts the logic level "1" data at its inputs into an equivalent binary code at its output.
- Generally, digital encoders produce outputs of 2-bit, 3-bit or 4-bit codes depending upon the number of data input lines.
- An "n-bit" binary encoder has 2n input lines and n-bit output lines with common types that include 4-to-2, 8-to-3 and 16-to-4 line configurations.
- The output lines of a digital encoder generate the binary equivalent of the input line whose value is equal to "1" and are available to encode either a decimal or hexadecimal input pattern to typically a binary or B.C.D. output code.

Cont..

4-to-2 Bit Binary Encoder:



Inputs				Outputs	
D_3	D_2	D_1	D_0	Q_1	Q_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1
0	0	0	0	x	x

Disadvantages of standard digital encoders

- They can generate the wrong output code when there is more than one input present at logic level "1".

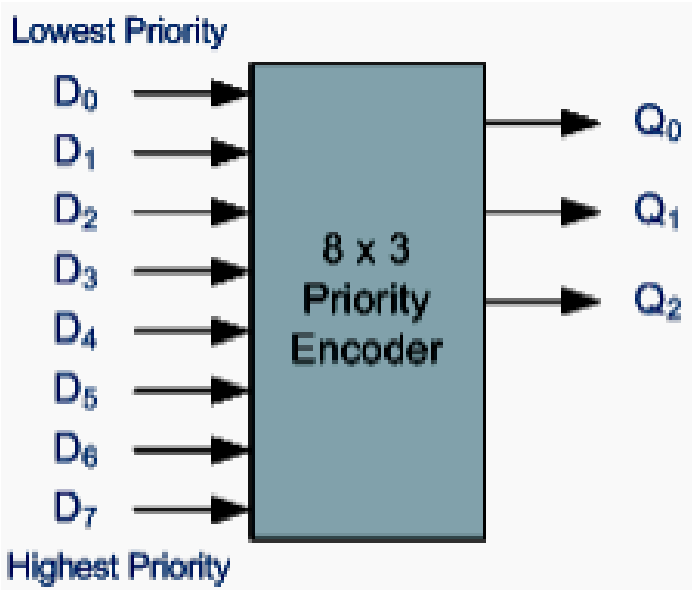
Cont..

Priority Encoder

- The Priority Encoder solves the problems mentioned above by allocating a priority level to each input.
- The ***priority encoders*** output corresponds to the currently active input which has the highest priority.
- So when an input with a higher priority is present, all other inputs with a lower priority will be ignored.
- The priority encoder comes in many different forms with an example of an 8-input priority encoder along with its truth table shown below.

Cont..

8-to-3 Bit Priority Encoder:



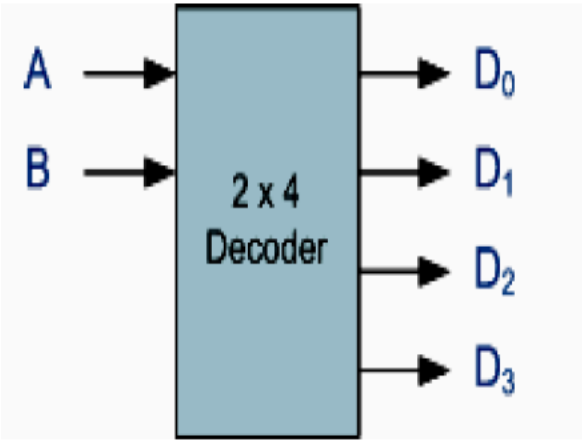
Inputs								Outputs		
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	x	0	0	1
0	0	0	0	0	1	x	x	0	1	0
0	0	0	0	1	x	x	x	0	1	1
0	0	0	1	x	x	x	x	1	0	0
0	0	1	x	x	x	x	x	1	0	1
0	1	x	x	x	x	x	x	1	1	0
1	x	x	x	x	x	x	x	1	1	1

X = dont care

Cont..

Decoder: A **Decoder** is the exact opposite to that of an "Encoder". It is basically, a combinational type logic circuit that converts the binary code data at its input into an equivalent decimal code at its output. **Binary Decoders** have inputs of 2-bit, 3-bit or 4-bit codes depending upon the number of data input lines, and a n-bit decoder has 2n output lines. Therefore, if it receives n inputs (usually grouped as a binary or Boolean number) it activates one and only one of its 2n outputs based on that input with all other outputs deactivated. A decoders output code normally has more bits than its input code and practical binary decoder circuits include, 2-to-4, 3-to-8 and 4-to-16 line configurations.

2-to-4 Binary Decoder :



A	B	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

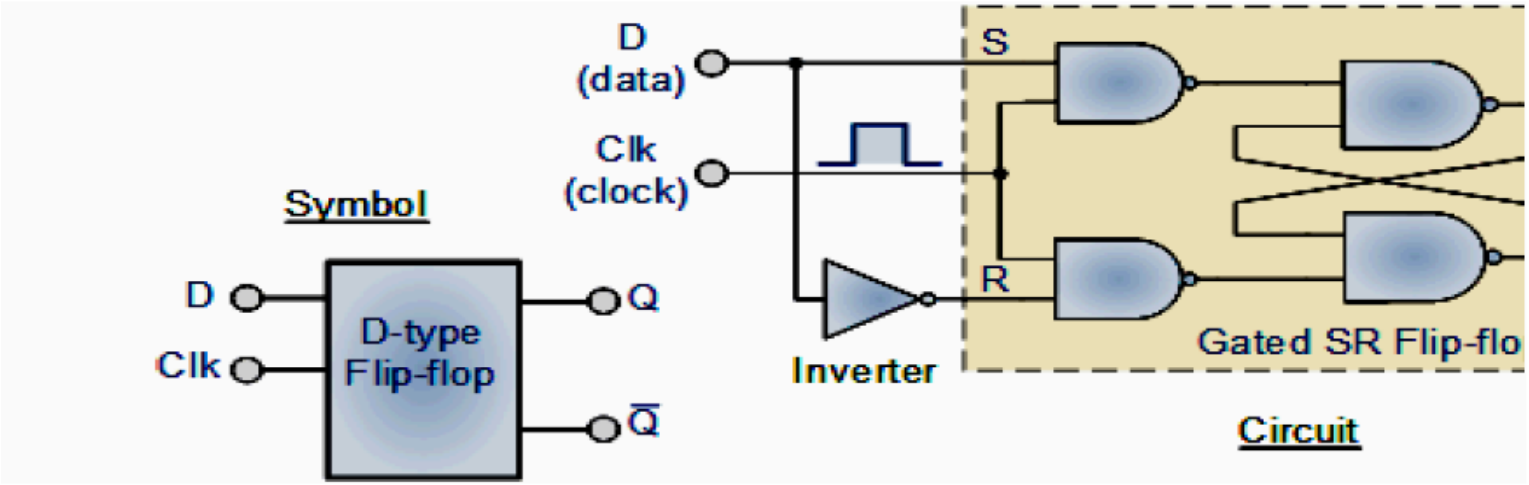
Cont..

In this simple example of a 2-to-4 line binary decoder, the binary inputs A and B determine which output line from D0 to D3 is "HIGH" at logic level "1" while the remaining outputs are held "LOW" at logic "0" so only one output can be active (HIGH) at any one time. Therefore, whichever output line is "HIGH" identifies the binary code present at the input, in other words it "de-codes" the binary input and these types of binary decoders are commonly used as **Address Decoders** in microprocessor memory applications.

The D flip-flop:

The **D flip-flop** is by far the most important of the clocked flip-flops as it ensures that inputs S and R are never equal to one at the same time. D-type flip-flops are constructed from a gated SR flip-flop with an inverter added between the S and the R inputs to allow for a single D (data) input. This single data input D is used in place of the "set" signal, and the inverter is used to generate the complementary "reset" input thereby making a level-sensitive D-type flip-flop from a level-sensitive RS-latch as now S = D and R = not D as shown.

Cont..



A simple SR flip-flop requires two inputs, one to "SET" the output and one to "RESET" the output. By connecting NOT gate to the SR flip-flop one can "SET" and "RESET" the flip-flop using just one input as now the two input signals are complements of each other. This complement avoids “forbidden state” in the SR latch when both inputs are LOW, since that state is no longer possible.

Thus the single input is called the "DATA" input. If this data input is HIGH the flip-flop would be "SET" and when it is LOW the flip-flop would be "RESET". However, this would be rather pointless since the flip-flop's output would always change on every data input. To avoid this an additional input called the "CLOCK" or "ENABLE" input is used to isolate the data input from the flip-flop after the desired data has been stored. The effect is that D is only copied to the output Q when the clock is active. This forms the basis of a **D flip-flop**.

Cont..

The **D flip-flop** will store and output whatever logic level is applied to its data terminal so long as the clock input is HIGH. Once the clock input goes LOW the "set" and "reset" inputs of the flip-flop are both held at logic level "1" so it will not change state and store whatever data was present on its output before the clock transition occurred. In other words the output is "latched" at either logic "0" or logic "1".

Truth Table for the D Flip-flop:

Clk	D	Q	Q	Description
↓ » 0	X	Q	Q	Memory no change
↑ » 1	0	0	1	Reset Q » 0
↑ » 1	1	1	0	Set Q » 1

Note: ↓ and ↑ indicates direction of clock pulse as it is assumed D flip-flops are edge triggered

Daily Quiz

- To avoid loading during read operation, the device used is
  a) latch
  b) flip flop
  c) buffer
  d) tristate buffer
- The input and output operations are respectively similar to the operations,
  a) read, read
  b) write, write
  c) read, write
  d) write, read
- Which is not the control bus signal:
  A. READ
  B. WRITE
  C. RESET
  D. None of these

Daily Quiz

- PROM stands for:
  - **A. Programmable read-only memory**
  - B. Programmable read write memory
  - C. Programmer read and write memory
  - D. None of these

- The method of connecting a driving device to a loading device is known as _____
  a) Compatibility
  b) **Interface**
  c) Sourcing
  d) Sinking

Weekly Assignment

1. What do you mean by tri-state devices?
2. With the help of a neat symbol explain tri-state buffer.
3. State function of buffer.
4. Explain different types of encoder.
5. With the help of neat block diagram explain 4 to 2 encoder.
6. What do you mean by priority encoder?
7. With the help of neat block diagram explain 8 to 3 encoder.
8. Explain decoder in detail.
9. With the help of neat block diagram explain 3 to 8 decoder.
10. Explain the working of D flipflop.

Topic Links

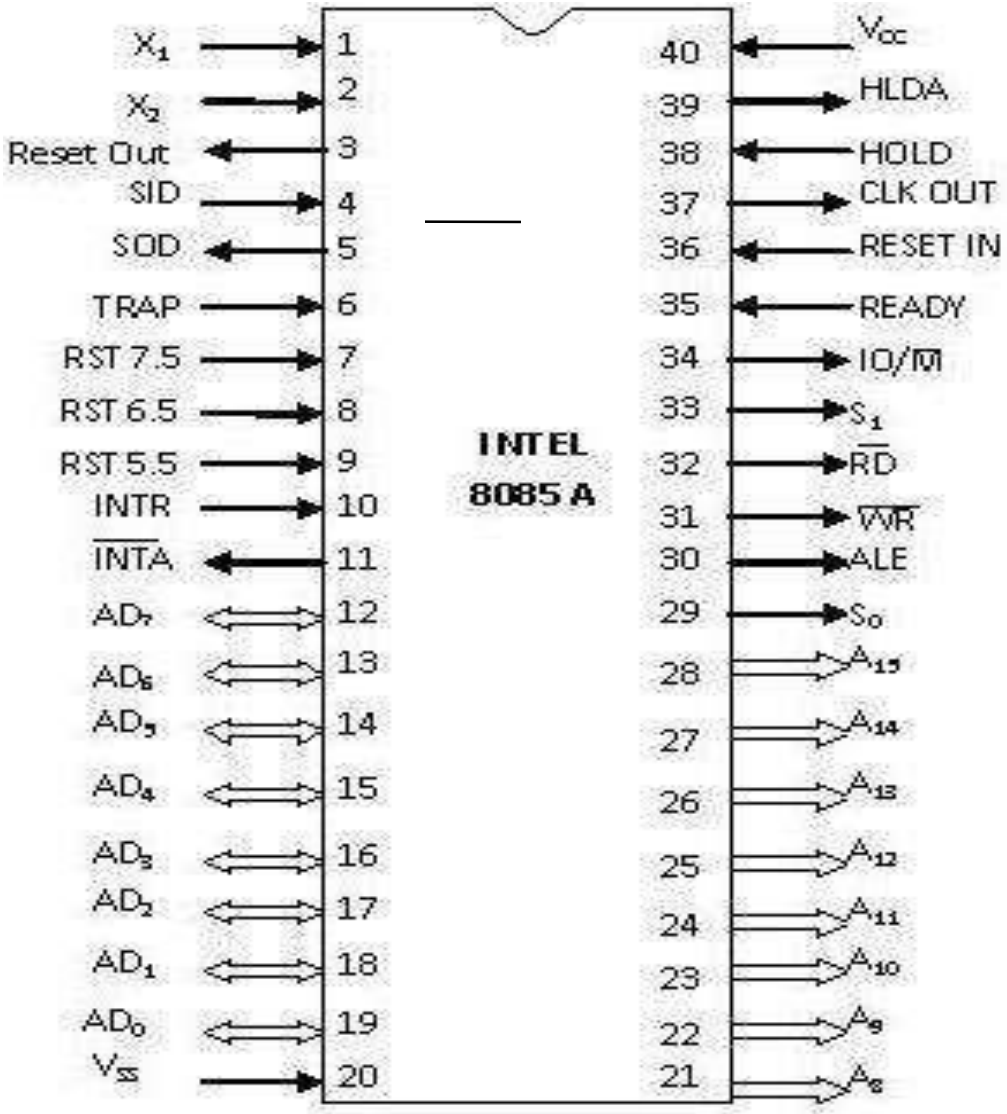
https://www.youtube.com/watch?v=_EJDoLnJ5BM

https://www.youtube.com/watch?v=bXOpC0z-8o8

Topic-5 Objectives

Name of the Topic	Objective of the topic	Mapping with CO
Pin Diagram and Internal Architecture of 8085	To understand the concept of Architecture and Pin configuration of 8085.	CO1

PIN Diagram


INTEL
8085 A
X1 1
X2 2
Reset Out 3
SID 4
SOD 5
TRAP 6
RST 7.5 7
RST 6.5 8
RST 5.5 9
INTR 10
INTA 11
AD7 12
AD6 13
AD5 14
AD4 15
AD3 16
AD2 17
AD1 18
AD0 19
VSS 20
40 VCC
39 HLDA
38 HOLD
37 CLK OUT
36 RESET IN
35 READY
34 IO/M
33 S1
32 RD
31 WR
30 ALE
29 S0
28 A15
27 A14
26 A13
25 A12
24 A11
23 A10
22 A9
21 A8

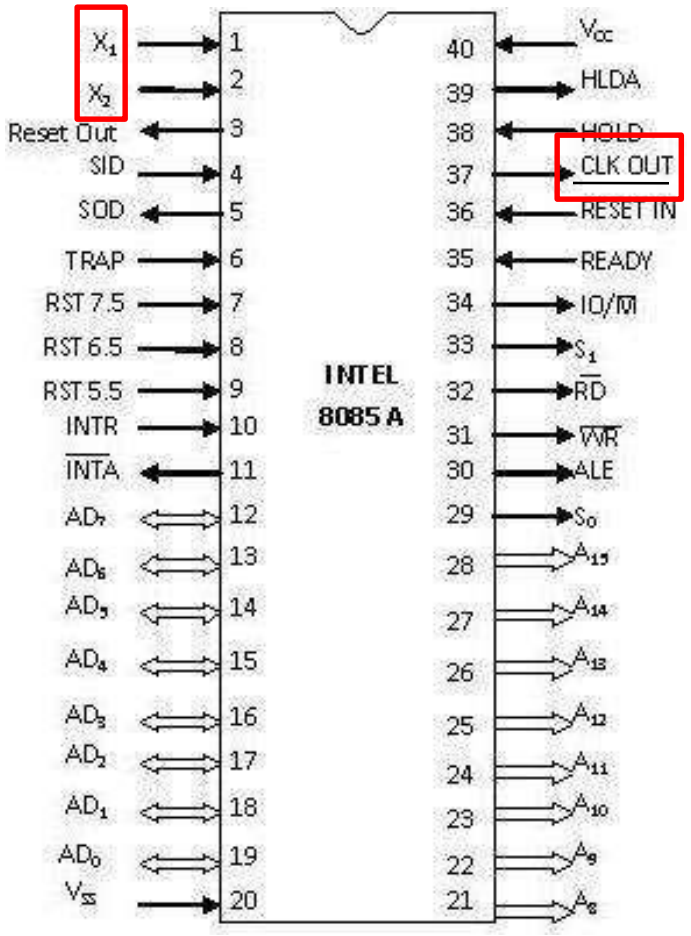
PIN Diagram

- The pins on the chip can be grouped into 6 groups:
  1. Address Bus.
  2. Data Bus.
  3. Control and Status Signals.
  4. Power supply and frequency.
  5. Externally Initiated Signals.
  6. Serial I/O ports.

PIN Diagram

Frequency Control Signals (X1 & X2) Pin 1 and Pin 2 (Input)

- There are 3 important pins in the frequency control group.
- **X0 and X1** are the inputs from the crystal or clock generating circuit.
- The frequency is internally divided by 2.
- So, to run the microprocessor at 3.072 MHz, a clock running at 6.144 MHz should be connected to the X0 and X1 pins.
- **CLK (OUT) Pin 37:** An output clock pin to drive the clock of the rest of the system.



Pins and signals

RESET IN and RESET OUT

Pin 36 (Input) and Pin 3 (Output)

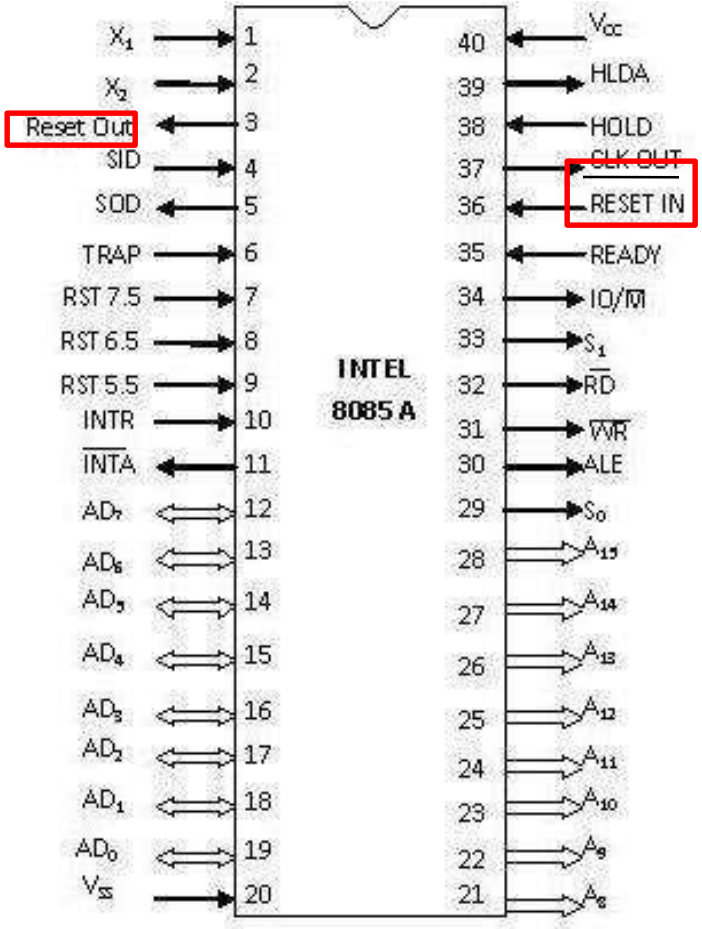
RESET IN

- It is used to reset the microprocessor.
- It is active low signal.
- When the signal on this pin is low

For at least 3 clocking cycles, it forces the microprocessor to reset itself.

– Resetting the microprocessor means:

1) Clearing the PC and IR.
2) Disabling all interrupts(except TRAP).
3) Disabling the SOD pin.
4) Gives HIGH output to RESET OUT pin



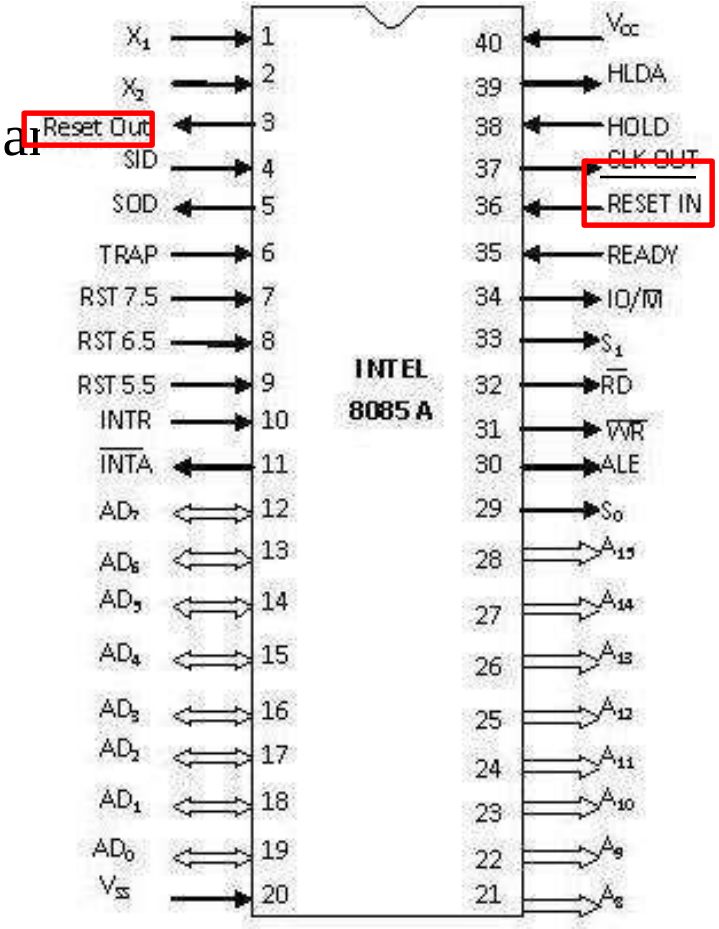
Pins and signals

RESET IN and RESET OUT

Pin 36 (Input) and Pin 3 (Output)

RESET OUT

- It is used to reset the peripheral devices a circuit.
- It is an active high signal.
- The output on this pin goes high
- whenever RESET IN is given low signal.

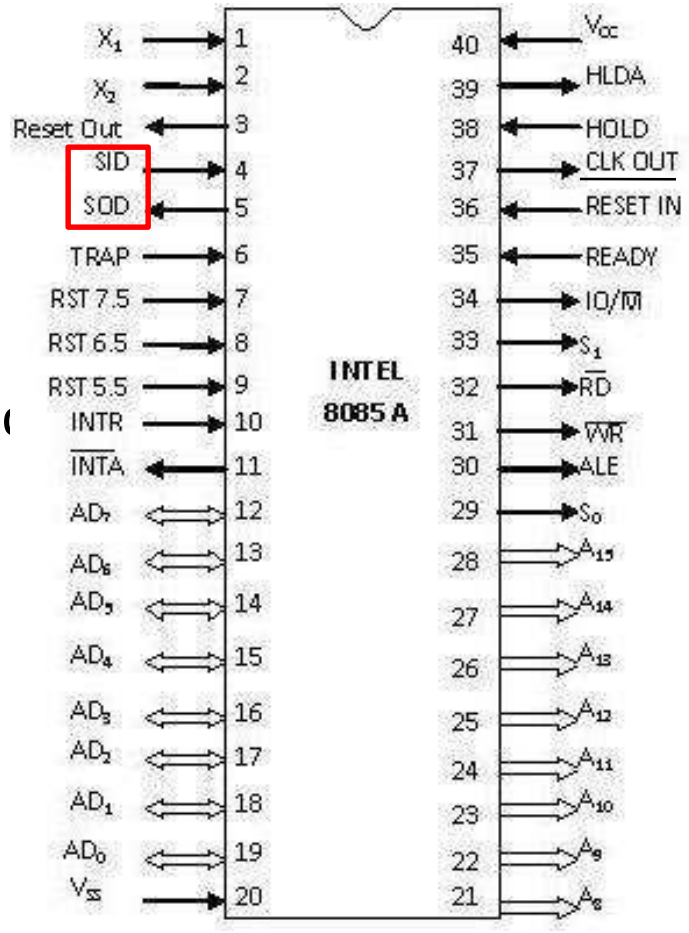


Pins and signals

SID and SOD Pin 4 (Input) and Pin 5 (Output)

SID (Serial Input Data)

- It takes 1 bit input from serial port of 8085.
- Stores the bit at the 8th position (MSB) of the Accumulator.
- RIM (Read Interrupt Mask) instruction is used to transfer the bit.

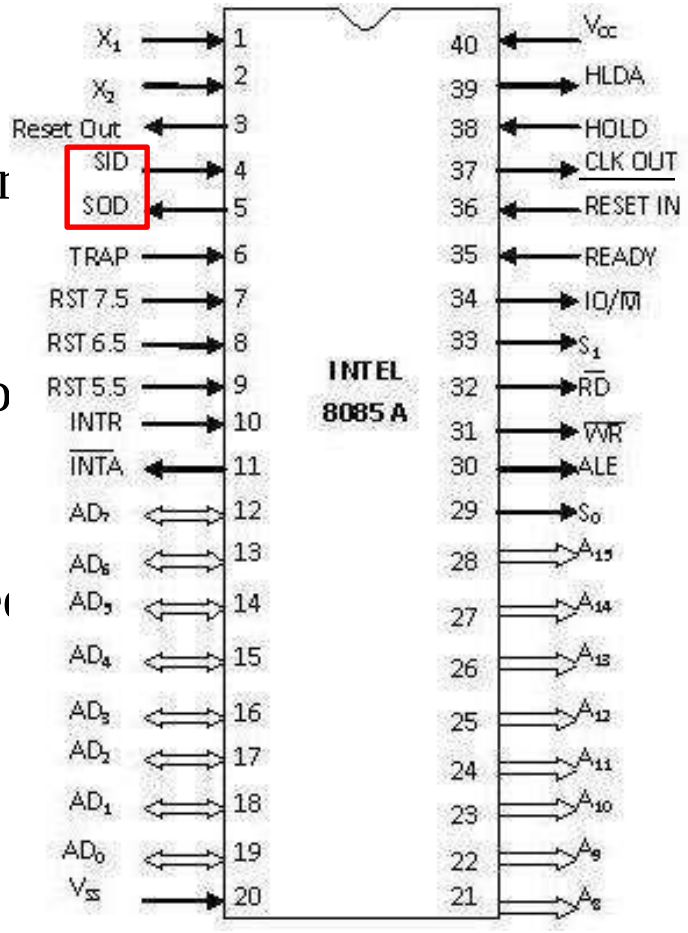


Pins and signals

SID and SOD Pin 4 (Input) and Pin 5 (Output)

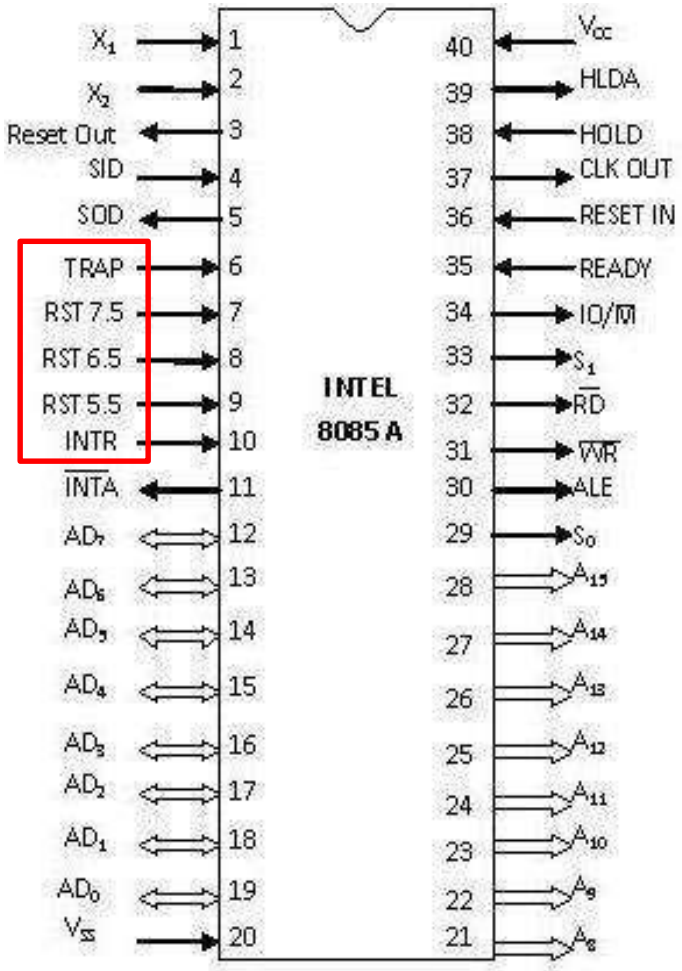
SOD (Serial Output Data)

- It takes 1 bit from Accumulator to serial por of 8085.
- Takes the bit from the 8th position (MSB) o the Accumulator.
- SIM (Set Interrupt Mask) instruction is use to transfer the bit.



Pins and signals

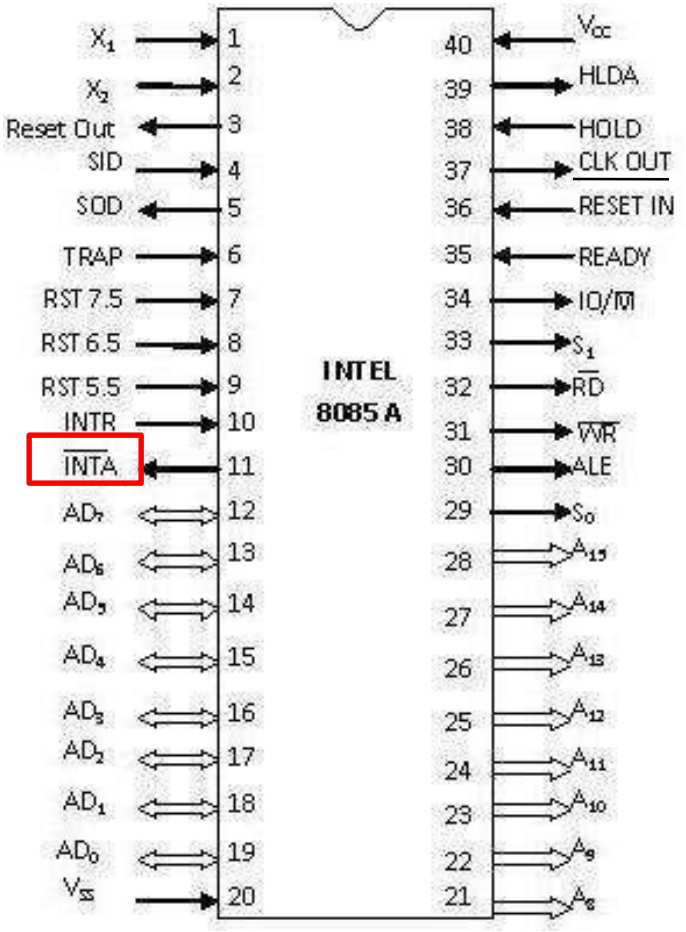
Interrupt Pin 6 to Pin 10



Pins and signals

$\overline{\text{INTA}}$ Pin 11 (Output)

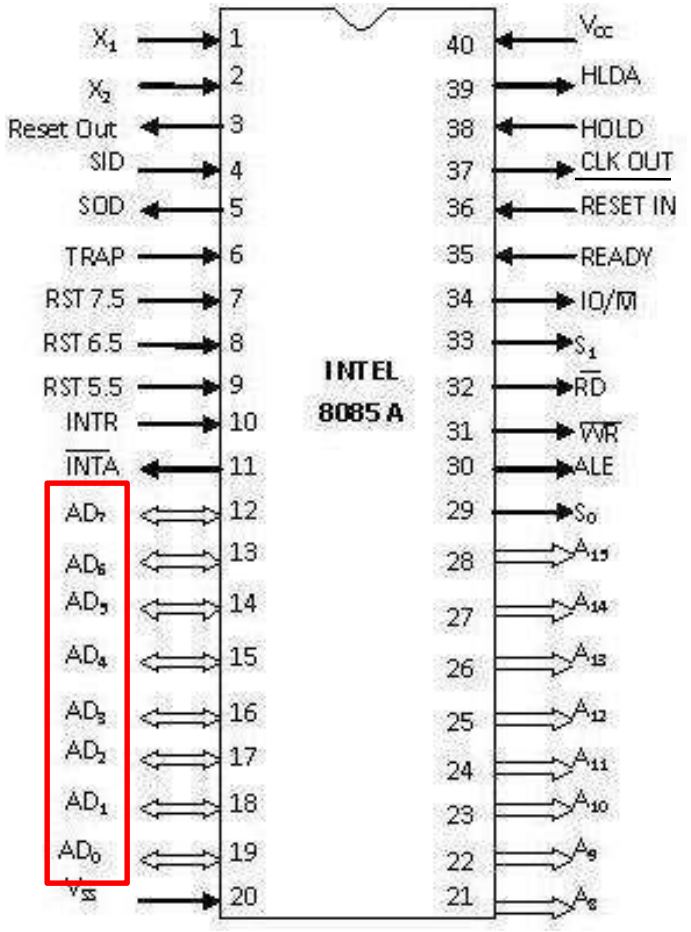
- It stands for interrupt acknowledge.
- It is an active low signal.
- Low output on this pin indicates that microprocessor has acknowledged the INTR request.



Pins and signals

AD0 – AD7 [Pin 19-12 (Bidirectional)]

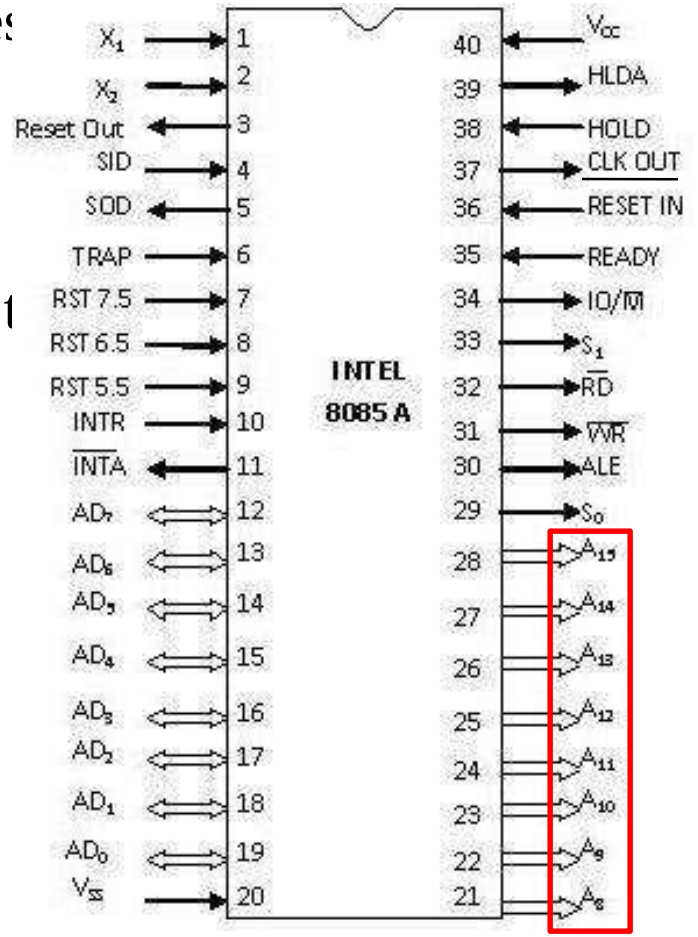
- These pins serve the dual purpose of transmitting lower order address and data byte.
- During 1st clock cycle, these pins act as lower half of address.
- In remaining clock cycles, these pins act as data bus.
- The separation of lower order address and data is done by address latch Enable (ALE).



Pins and signals

A8 – A15 [Pin 21-28 (Unidirectional)]

- These pins carry the higher order of addres bus.
- The address is sent from microprocessor t memory.



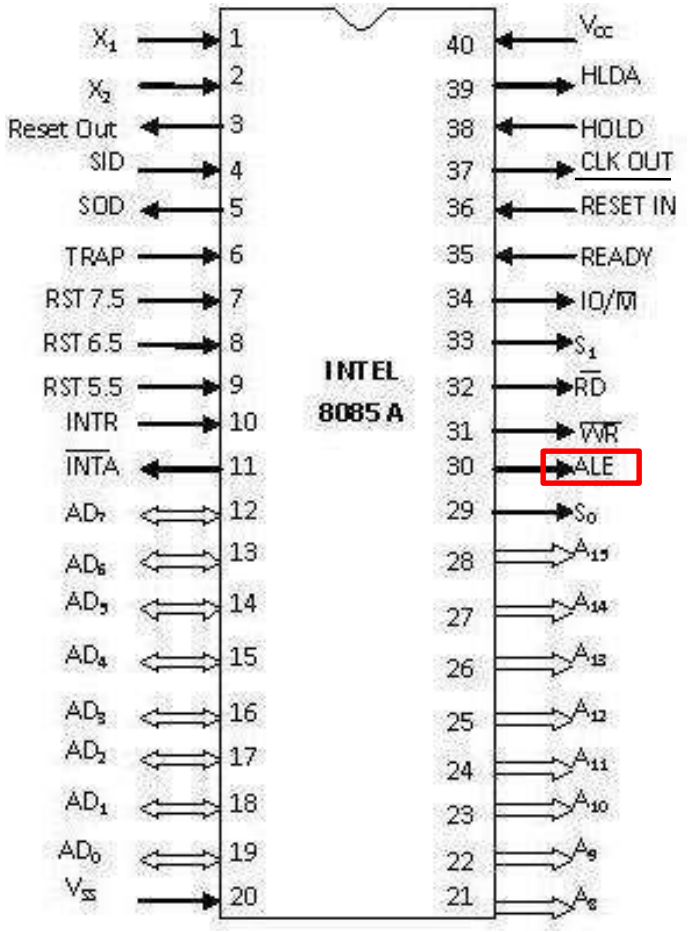
Pins and signals

ALE [Pin 30 (Output)]

- It is used to enable Address Latch.
- If ALE = 1 then

  Bus functions as address bus.
- If ALE = 0 then

  Bus functions as data bus.

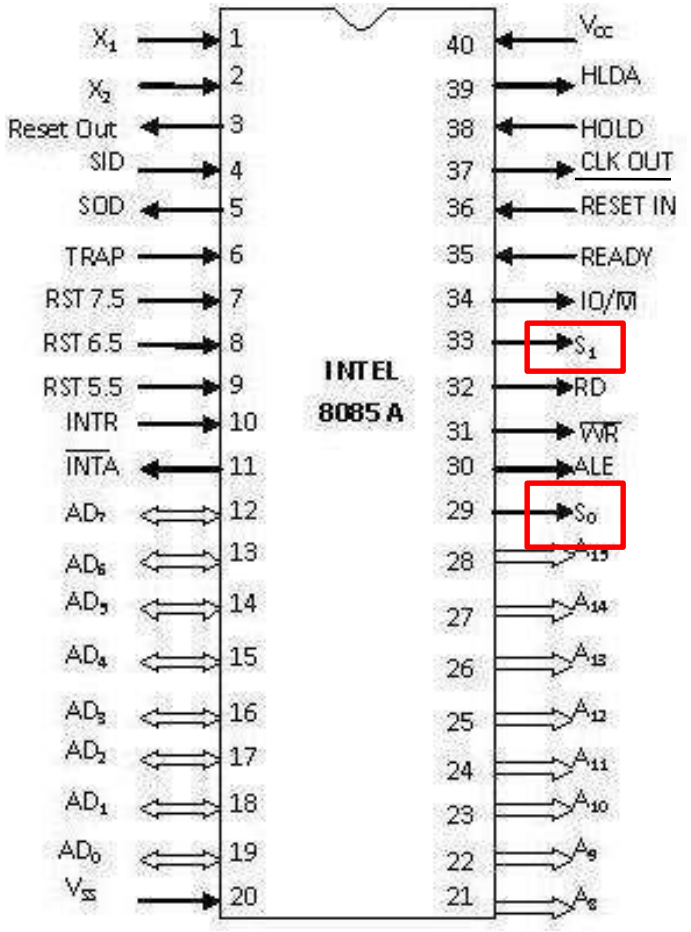


Pins and signals

S0 and S1 [Pin 29 (Output) and Pin 33 (Output)]

- S0 and S1 are called Status Pins.
- They tell the current operation which is in progress in 8085.

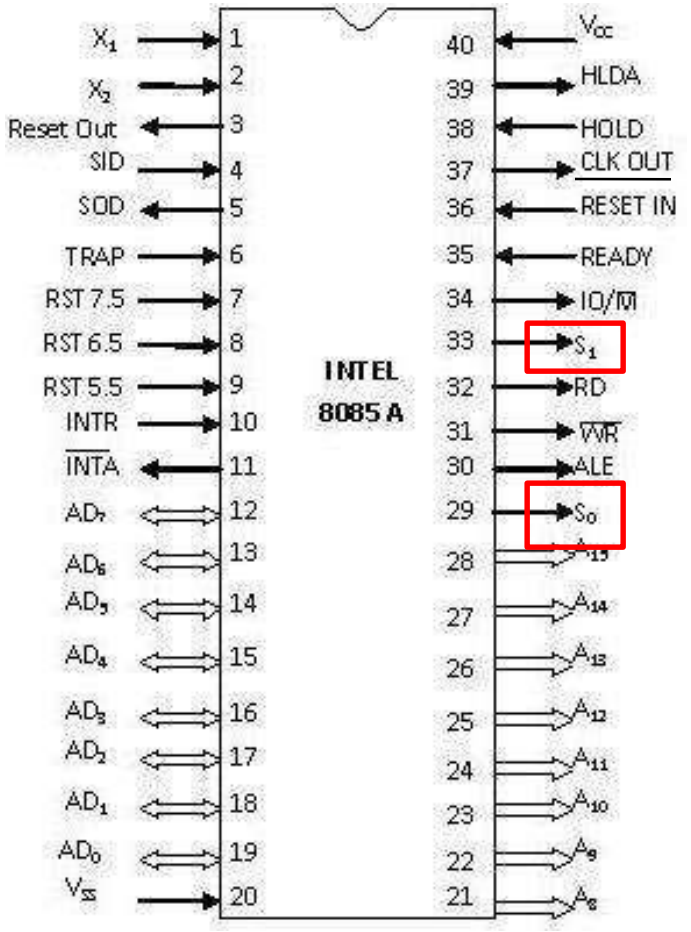
S_1	S_0	Operation
0	0	Halt
0	1	Write
1	0	Read
1	1	Opcode Fetch



Pins and signals

$\overline{IO/M}$ [Pin 34 (Output)]

- This pin tells whether I/O or memory operation is being performed.
- If $\overline{IO/M}$ = 1 then
  - I/O operation is being performed.
- If $\overline{IO/M}$ = 0 then
  - Memory operation is being
  - performed.



Pins and signals

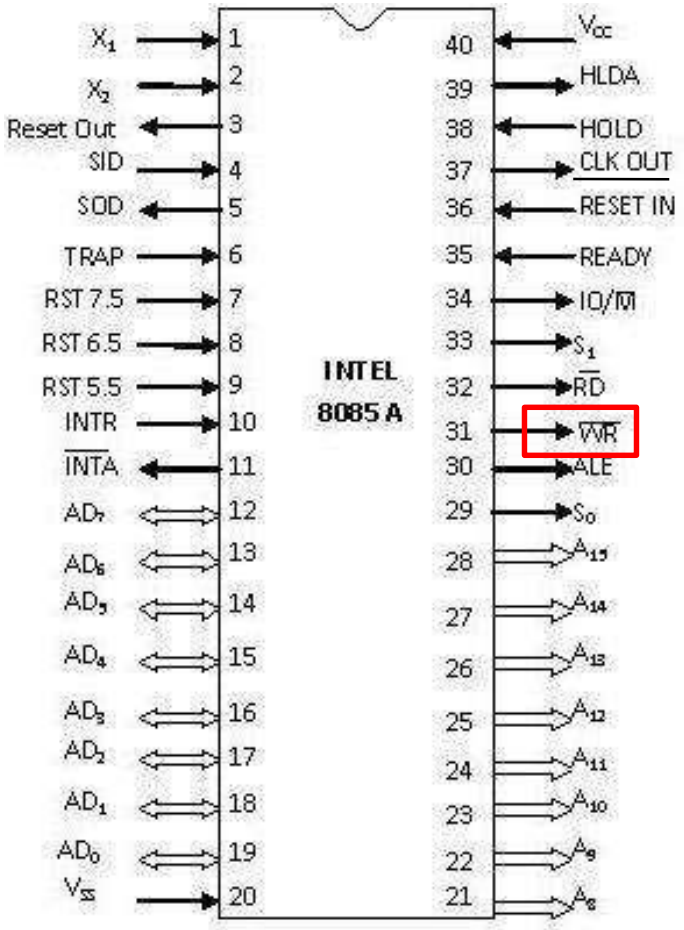
Table Showing $IO/\overline{M}$, S_0, S_1 and Corresponding Operations

Operations	$IO/\overline{M}$	S_1	S_0
Opcode Fetch	0	1	1
Memory Read	0	1	0
Memory Write	0	0	1
I/O Read	1	1	0
I/O Write	1	0	1
Interrupt Ack.	1	1	1
Halt	High Impedance	0	0

Pins and signals

$\overline{WR}$ Pin 31 (Output)

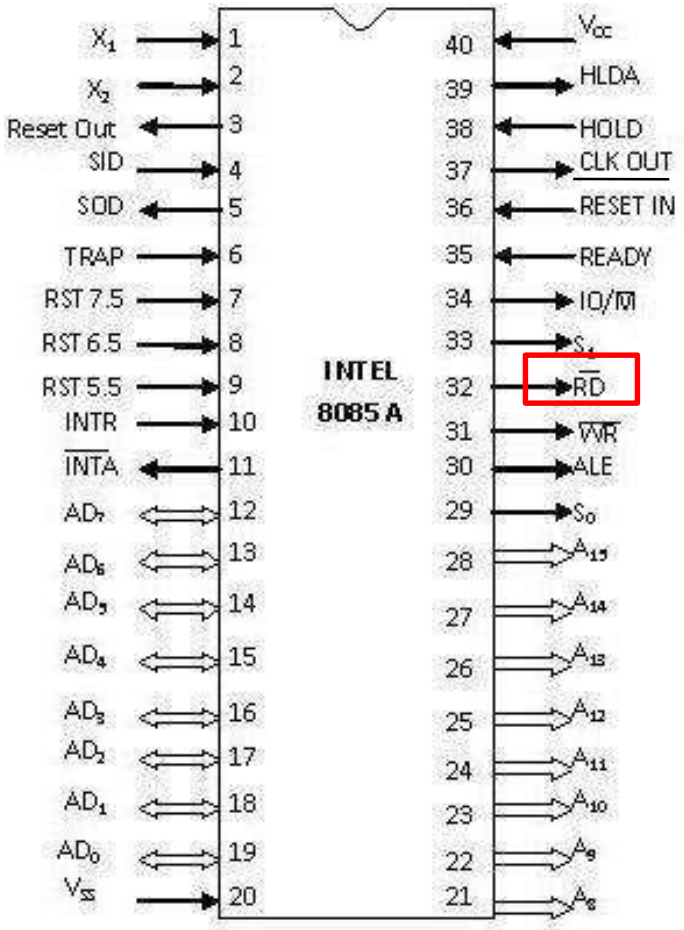
- WR stands for write.
- It is an active low signal.
- It is a control signal used for Write operation either into memory or into output device.



Pins and signals

$\overline{RD}$ Pin 32 (Output)

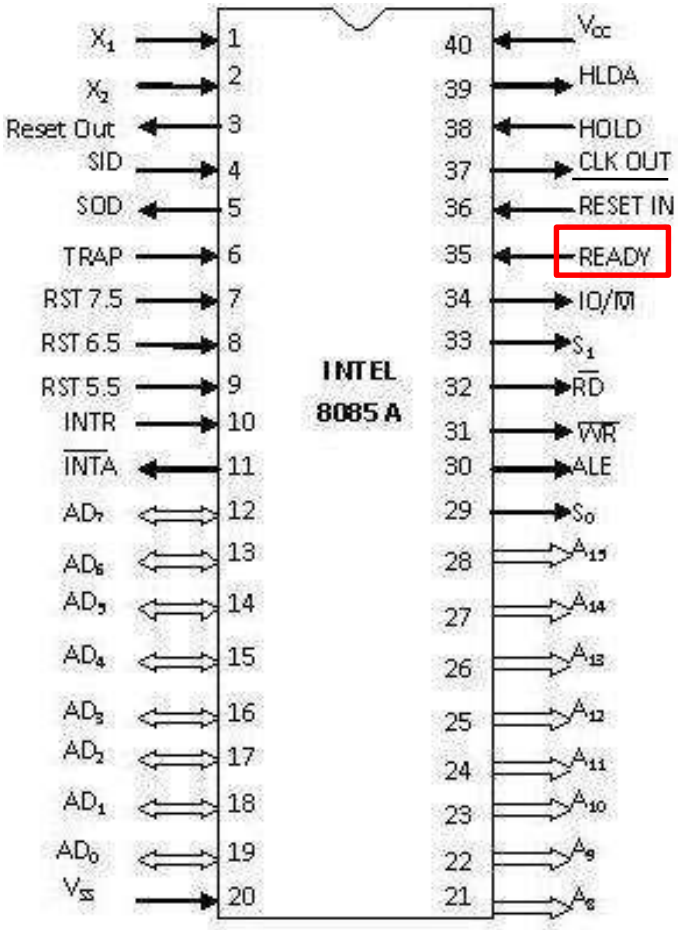
- RD stands for Read.
- It is an active low signal.
- It is a control signal used for Read operation either from memory or from Input device.



Pins and signals

READY Pin 35 (Input)

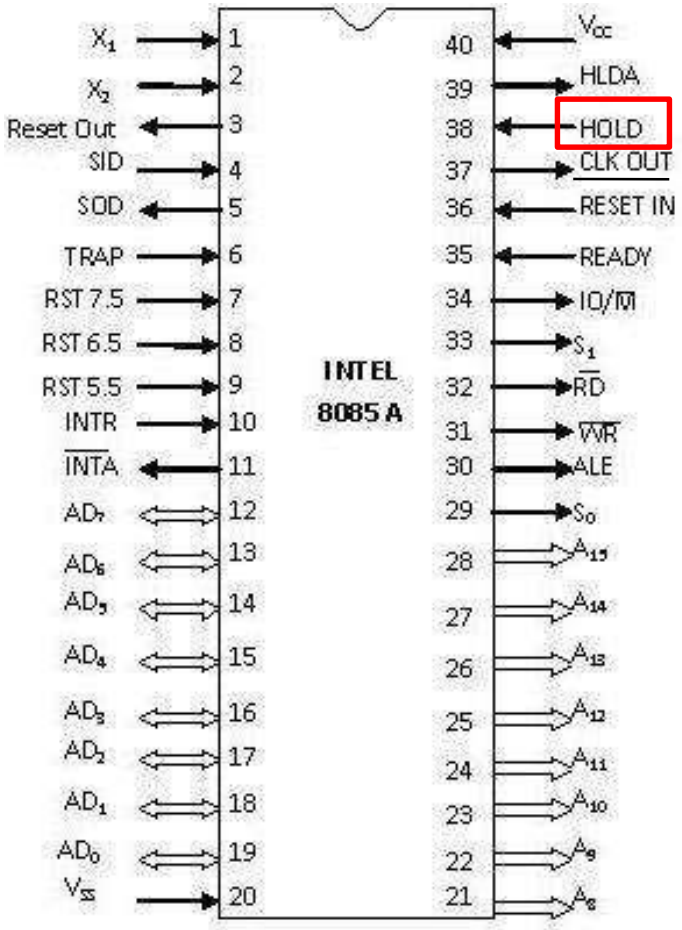
- This pin is used to synchronize slower peripheral devices with fast microprocessor.
- A low value causes the microprocessor to enter into ***wait state***.
- The microprocessor remains in wait state until the input at this pin goes high.



Pins and signals

HOLD [Pin 38 (Input)]

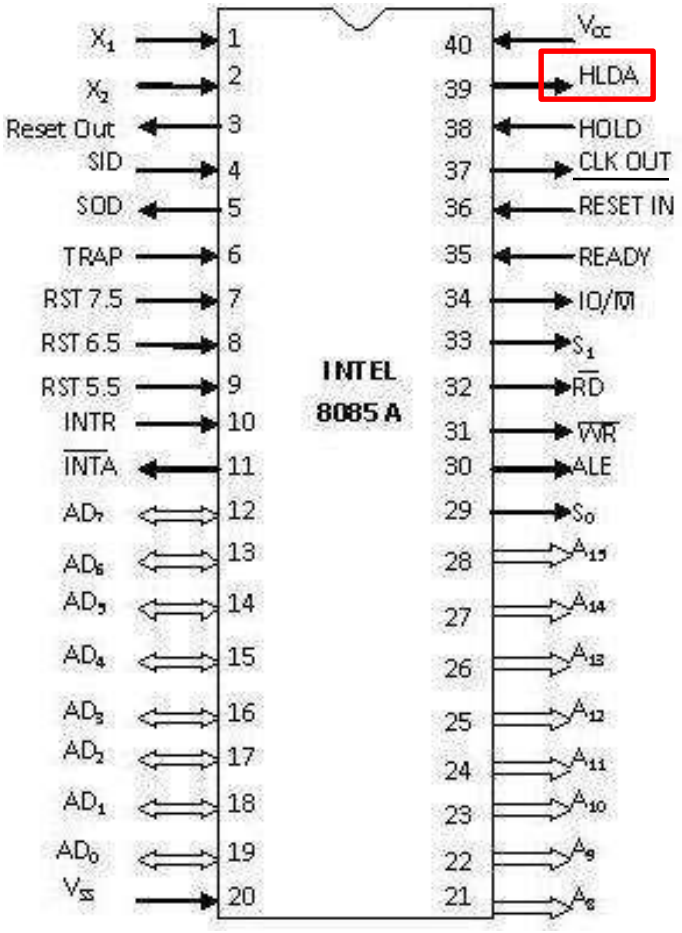
- HOLD pin is used to request the microprocessor for DMA transfer.
- A high signal on this pin is a request to microprocessor to relinquish the hold on buses.
- This request is sent by DMA controller.
- Intel 8257 and Intel 8237 are two DMA controllers.



Pins and signals

HLDA [Pin 39 (Output)]

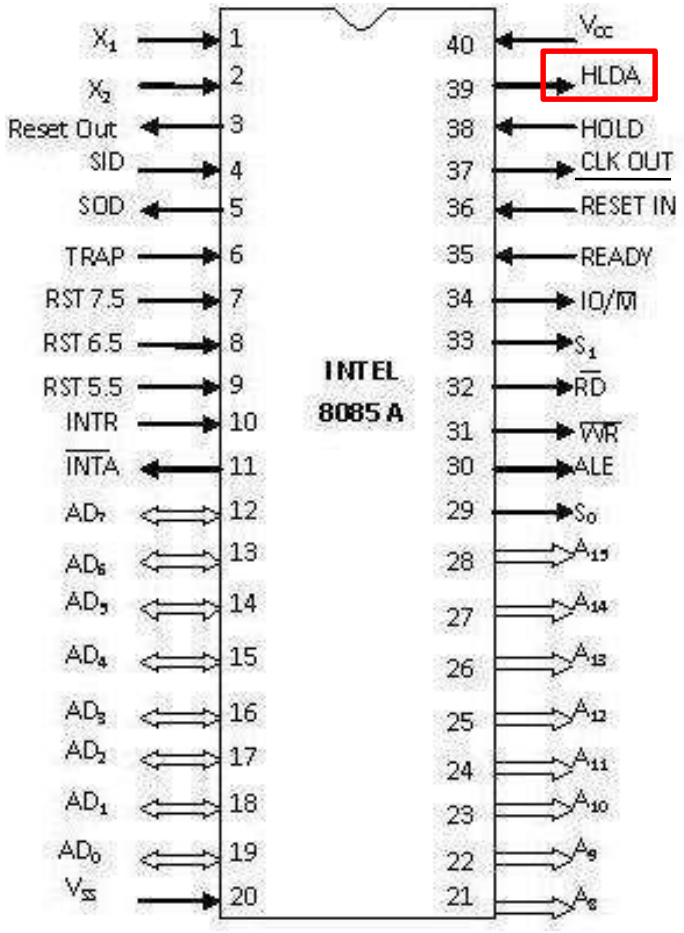
- HLDA stands for Hold Acknowledge.
- The microprocessor uses this pin to acknowledge the receipt of HOLD signal.
- When HLDA signal goes high, address bus, data bus, RD, WR, IO/$\overline{M}$ pins are tri-stated.
- This means they are cut-off from external environment.



Pins and signals

HLDA [Pin 39 (Output)]

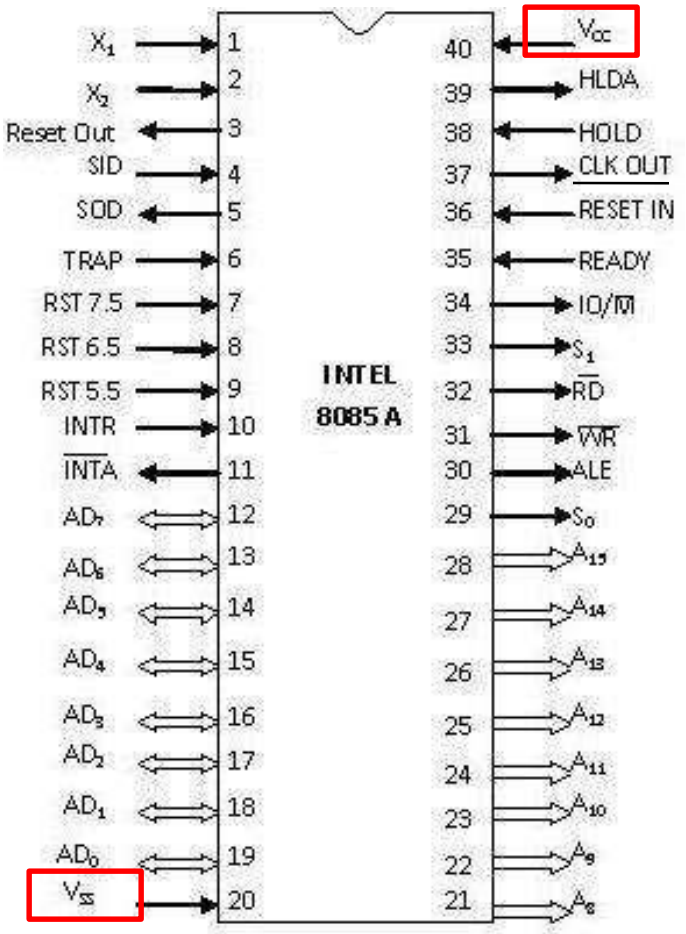
- The control of these buses goes to DMA Controller.
- Control remains at DMA Controller until HOLD is held high.
- When HOLD goes low, HLDA also goes low and the microprocessor takes control of the buses.



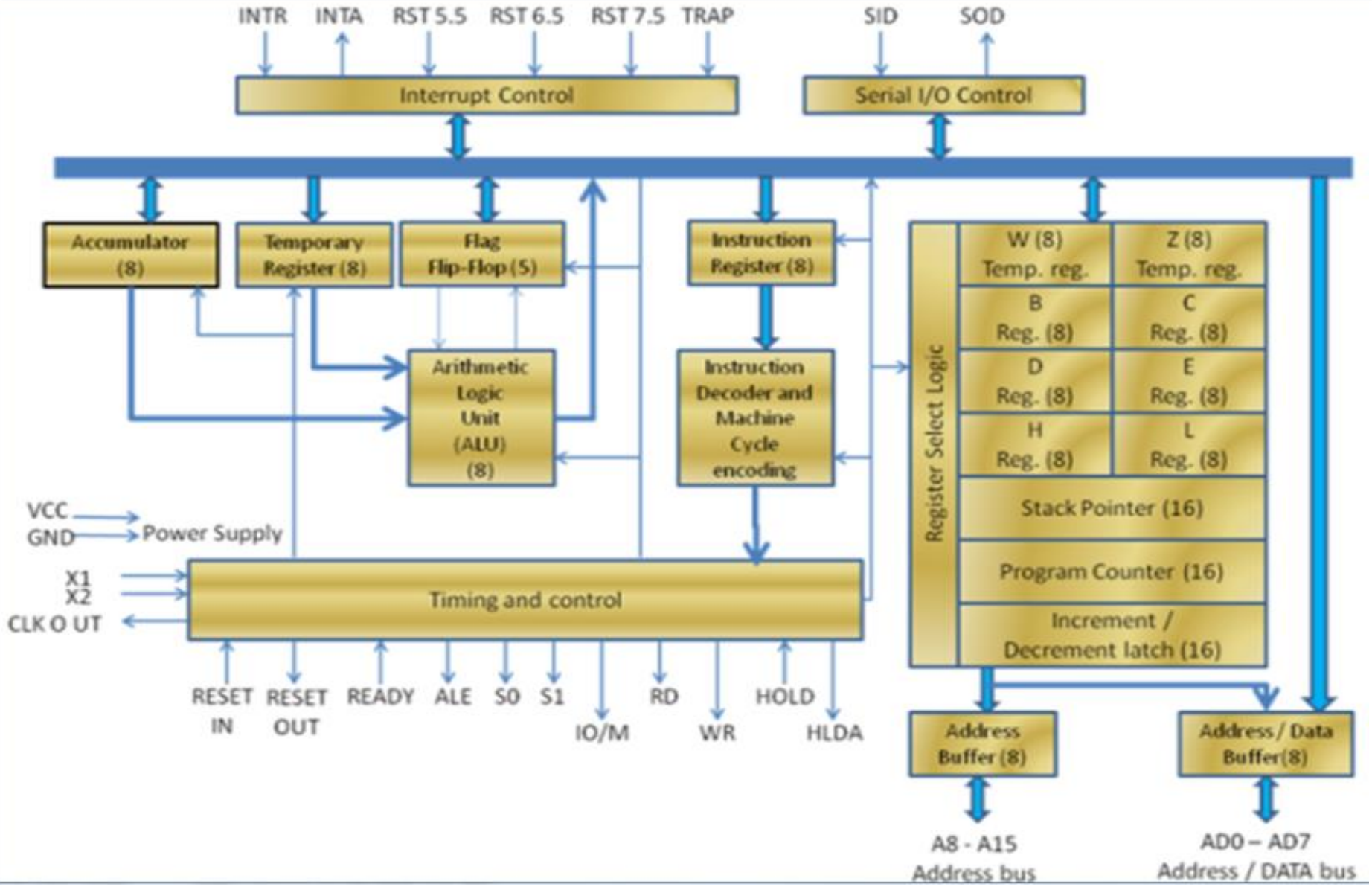
Pins and signals

Vss and Vcc [Pin 20 (Input) and Pin 40 (Input)]

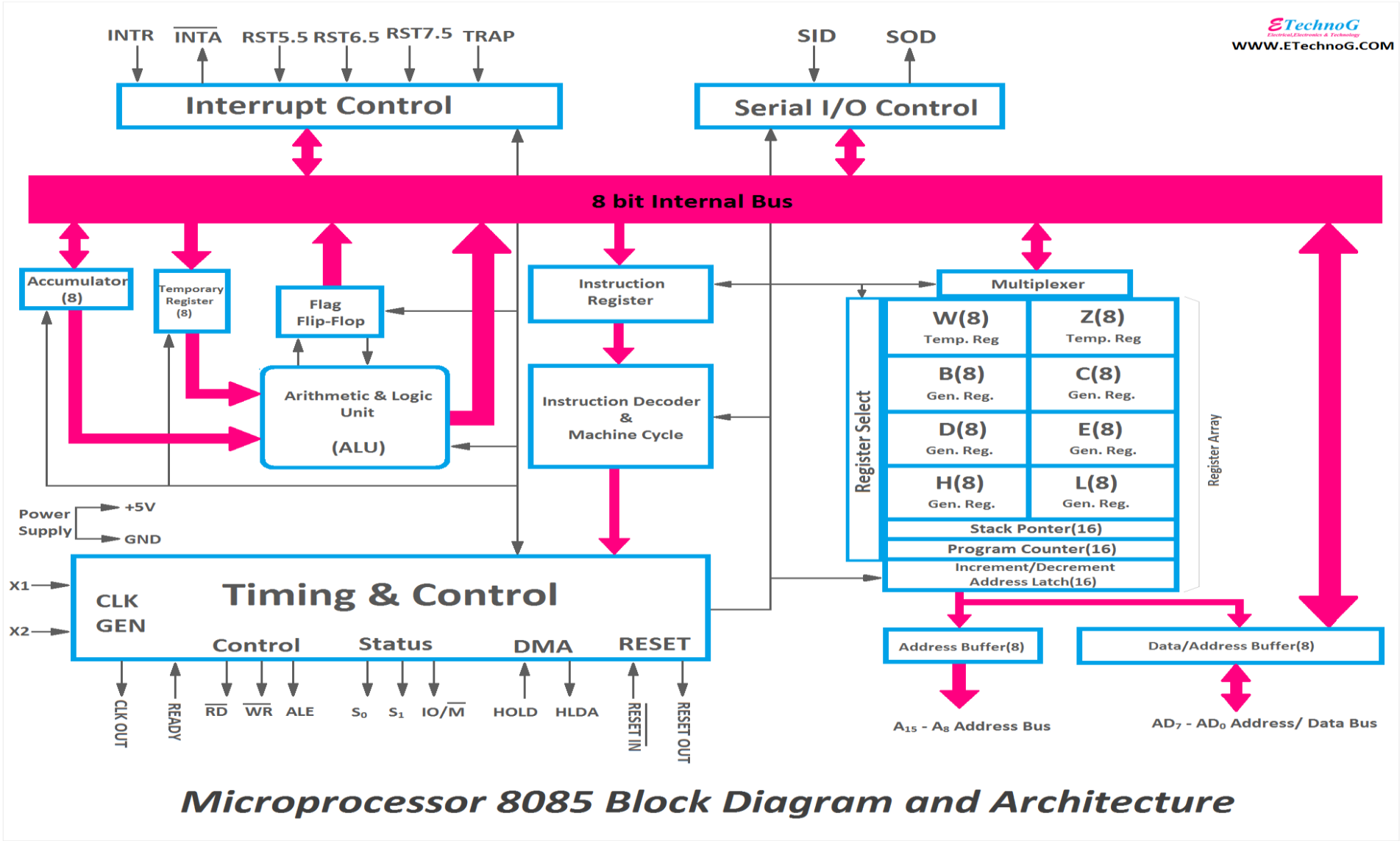
- +5V power supply is connected to VCC.
- Ground signal is connected to VSS.



8085 Microprocessor Architecture



8085 Microprocessor Architecture



Microprocessor 8085 Block Diagram and Architecture

8085 Microprocessor Architecture

- The 8085 uses three separate busses to perform its operations:
  - **Address bus**.
  - **Data bus.**
  - **Control bus**.

16 bits wide (A0 A1…A15) Address bus:

- Therefore, the 8085 can access locations with numbers from 0 to 65,535. Or, the 8085 can access a total of 64K addresses.
- "Unidirectional".
- Information flows out of the microprocessor and into the memory or peripherals.
- When the 8085 wants to access a peripheral or a memory location, it places the 16-bit address on the address bus and then sends the appropriate control signals.

8085 Microprocessor Architecture

The Data Bus

- 8 bits wide (D_0 D_1…D_7)
- "Bi-directional".
  - Information flows both ways between the microprocessor and memory or I/O.
- The 8085 uses the data bus to transfer the binary information.
- Since the data bus has 8-bits only, then the 8085 can manipulate data 8 bits at-a-time only.

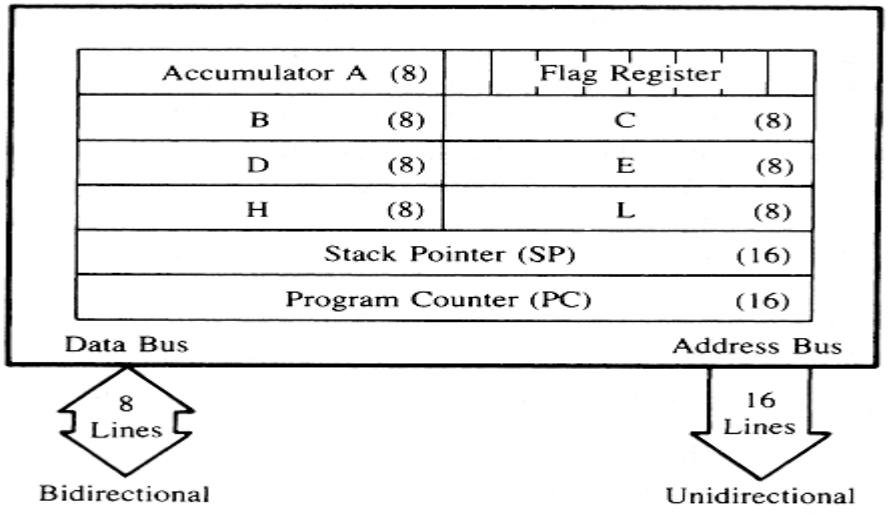
8085 Microprocessor Architecture

The Control Bus

- The control bus is made up of a number of single bit control signals.

Registers

- Six general purpose registers used to store 8-bit data during a program execution.
- The registers are identified as B, C, D, E, H, and L.
- They can be combined as register pairs: BC, DE, and HL to perform 16-bit operations.



8085 Microprocessor Architecture

Program Counter (PC) [16]

- This is used to control the sequencing of the execution of instructions.
- This register always holds the address of the next instruction.

Stack pointer [16]

- It is used to point into memory (stack).
- The stack is an area of memory used to hold data that will be retreived soon.
- The stack is usually accessed in a Last In First Out (LIFO) fashion.

Accumulator [8]

- 8-bit register that is part of the ALU.
- Used to store 8-bit data and in performing 8-bit arithmetic and logical operations, and in storing the results of operations.

8085 Microprocessor Architecture

FLAGS OF 8085

- Flag Register: It is a group of 5 flip flops used to know status of various operations done and is given by:

S	Z	X	AC	X	P	X	CY

- S: Sign flag is set when result of an operation is negative.
- Z: Zero flag is set when result of an operation is 0.
- AC: Auxiliary carry flag is set when there is a carry out of lower nibble or lower four bits of the operation.
- CY: Carry flag is set when there is carry generated by an operation.
- P: Parity flag is set when result contains even number of 1's.

  Rest(X) are don't care flip flops.
- 8085 uses these flags in decision-making process.

8085 Microprocessor Architecture

Interrupts Control Unit

- Interrupt is a signal sent by a peripheral interface or a software program to microprocessor to perform a particular task or work.
- Intel 8085 Microprocessor has 5 hardware interrupts and 8 software interrupts.

- **Hardware interrupts** in the descending order of their priority :
  TRAP (RST 4.5) , RST 7.5 , RST 6.5 , RST 5.5 , INTR
- **Software interrupts :**
  RST 0 , RST 1 , RST 2, RST 3, RST 4 , RST 5 , RST 6 , RST 7.

8085 Microprocessor Architecture

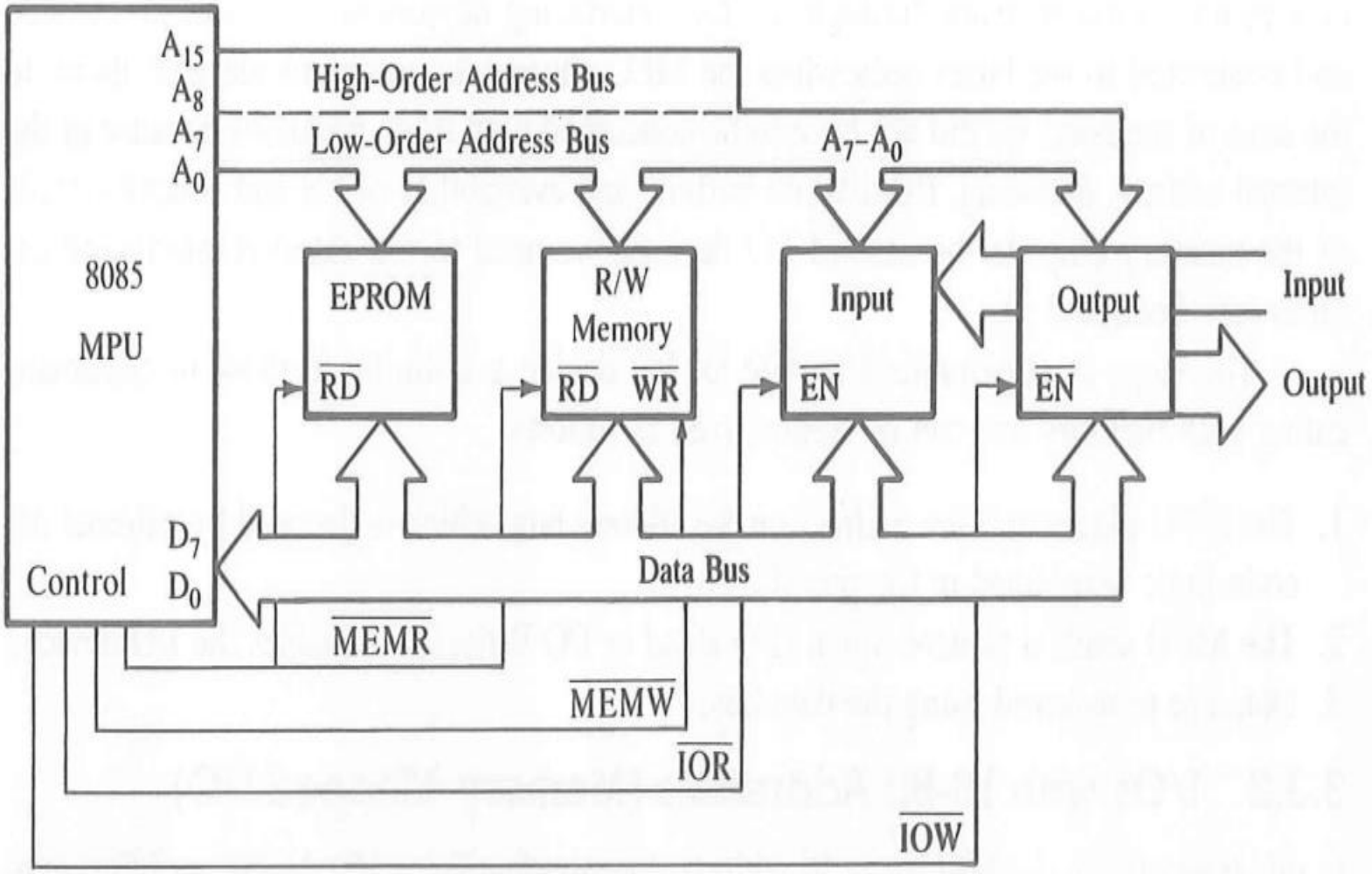
Timing & Control Unit

- This unit synchronizes all the microprocessors operation with the clock & generates the control signal necessary for communication between the microprocessors & peripherals.
- The RD & WR signals are indicating the availability of the data on the data bus.

Instruction Register & Decoder

- This is a part of ALU.
- When an instruction is fetched from memory, it is loaded in the instruction register.
- The decoder decodes the instruction & establishes the sequence of events to follow.

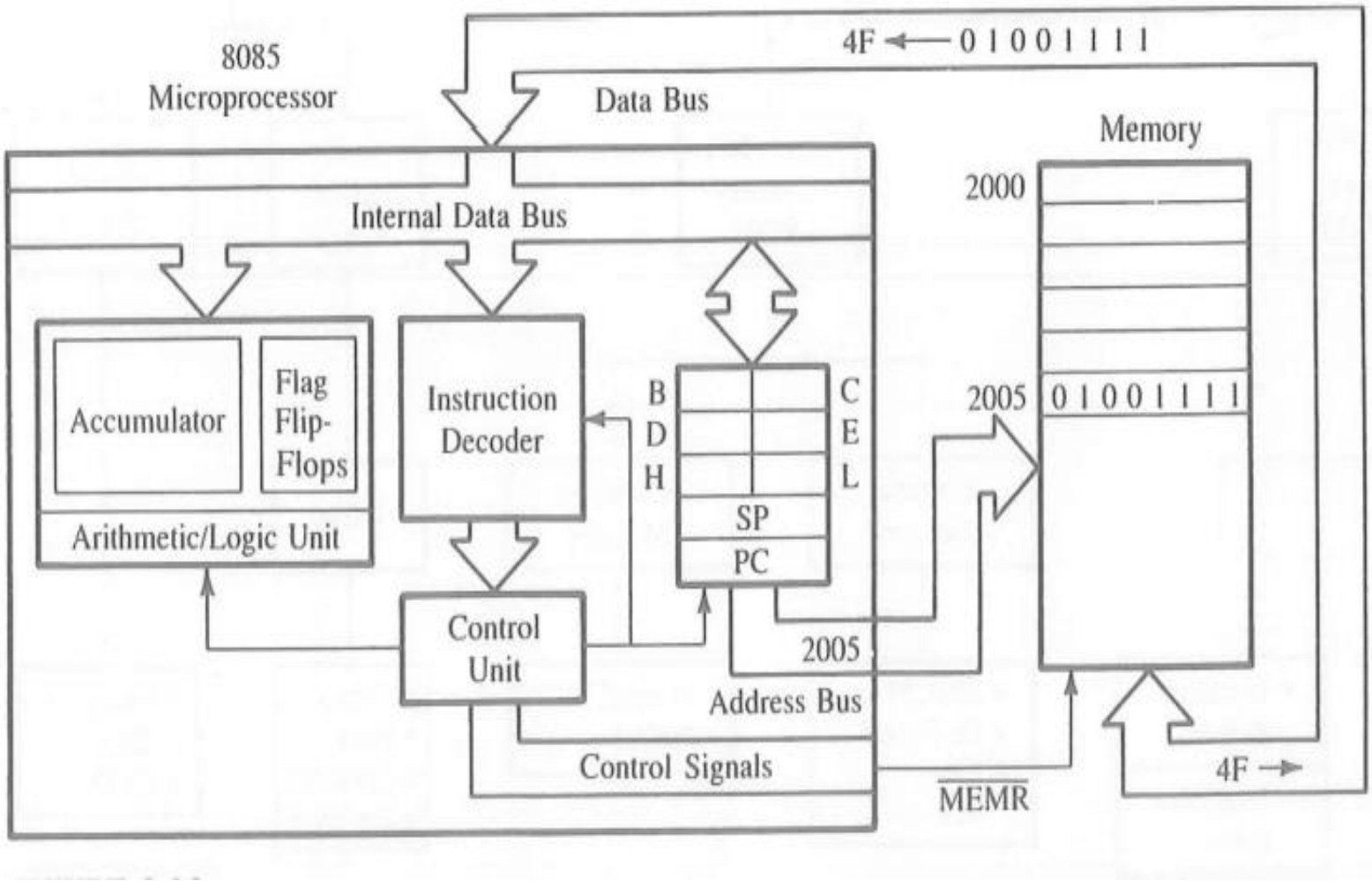
Example of an 8085 based Computer



Instruction and Data Flow

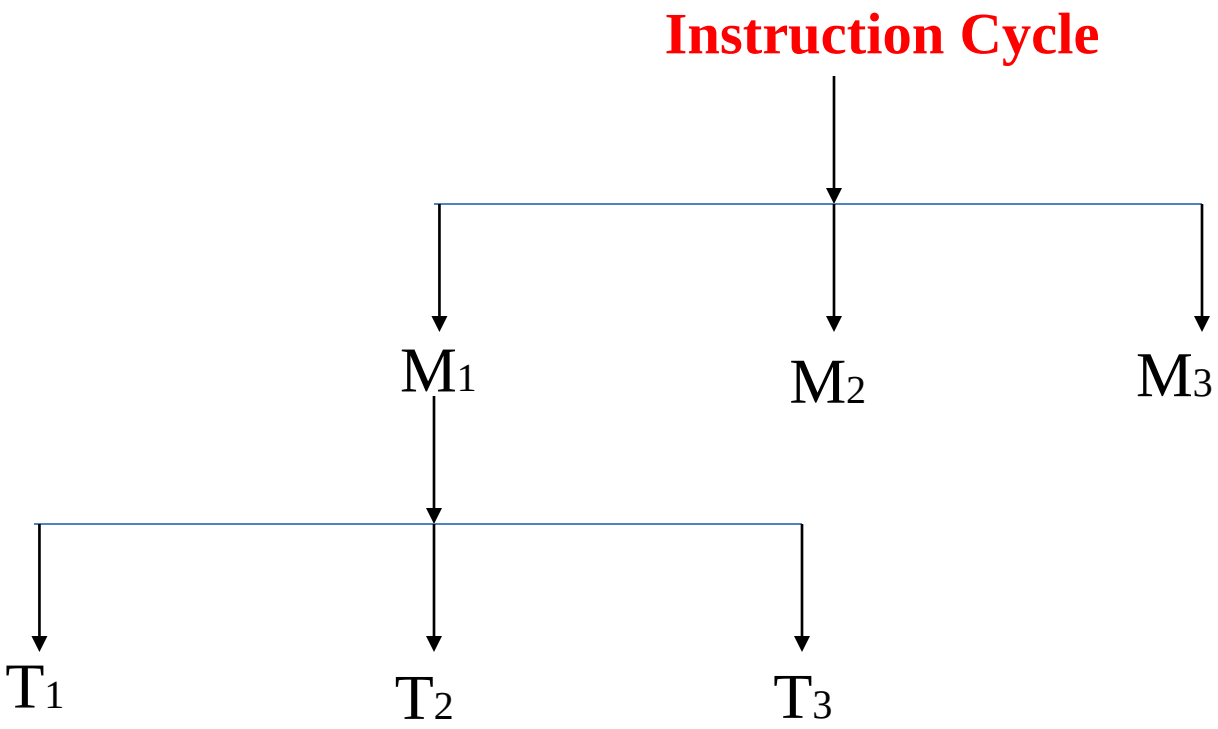
- The instruction code 0100 1111 (4FH) is stored in memory location 2005H. Illustrate the data flow and list the sequence of events when the instruction code is fetched by the MPU.

Instruction and Data Flow



Timer and Timing Diagram

MACHINE CYCLES AND THEIR TIMING OF 8085



M = Machine Cycle
T = T States

Timing Diagrams

MACHINE CYCLES AND THEIR TIMING OF 8085

- **Timing Diagram** is a graphical representation. It represents the execution time taken by each instruction.

T- State:

Microprocessor performs an operation in specific time period i.e clock cycles. Each clock cycles is called T-State.

Machine Cycle:

The time required to complete one operation of accessing memory, I/O, or acknowledging an external request. This cycle may consist of 3 to 6 T-states.

Instruction Cycle:

The time required to complete the execution of an instruction. In the 8085, an instruction cycle may consist of 1 to 6 machine cycles.

Timing Diagrams

MACHINE CYCLES AND THEIR TIMING OF 8085

Fetch Cycle:

In this cycle, the CPU fetches opcode of the instruction of the memory.

Execute Cycle:

- In this cycle, the data acquired from the memory & complete the operation.
- Total time taken to execute an instruction = FC + EC.

Timing Diagrams

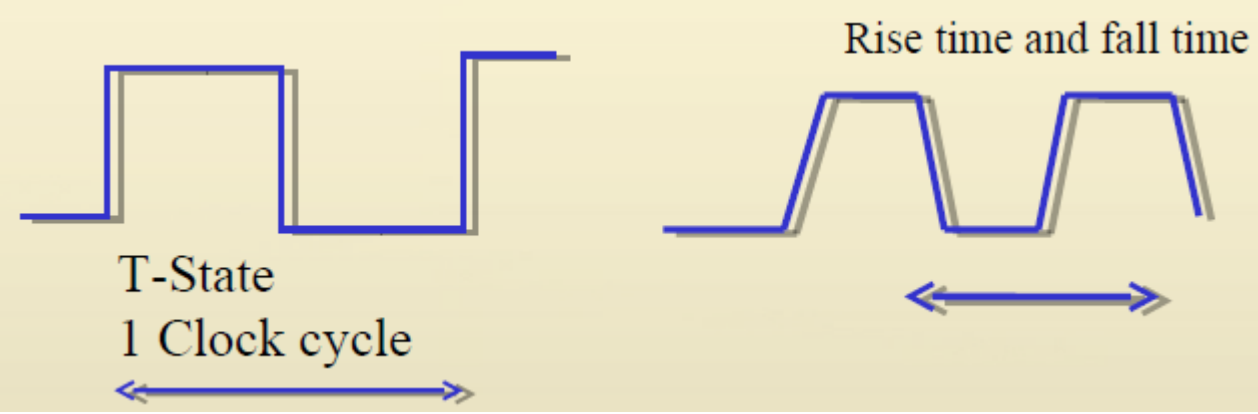
The 8085 microprocessor has 5 basic machine cycles.

1. Opcode fetch cycle (4T)
2. Memory write cycle (3 T)
3. Memory read cycle (3 T)
4. I/O read cycle (3 T)
5. I/O write cycle (3 T)

Timing Diagrams

Clock Signal

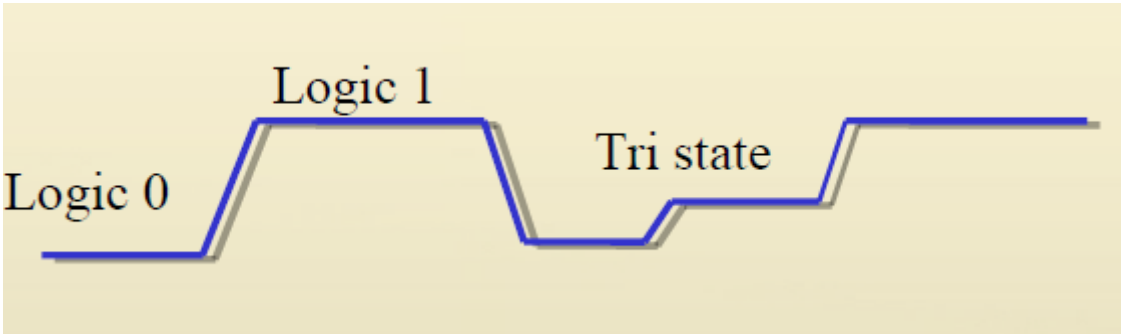
- The 8085 divides the clock frequency provided at x1 and x2 inputs by 2 which is called operating frequency.



Timing Diagrams

Single Signal

- Single signal status is represented by a line. It may have status either logic 0 or logic 1 or tri-state.

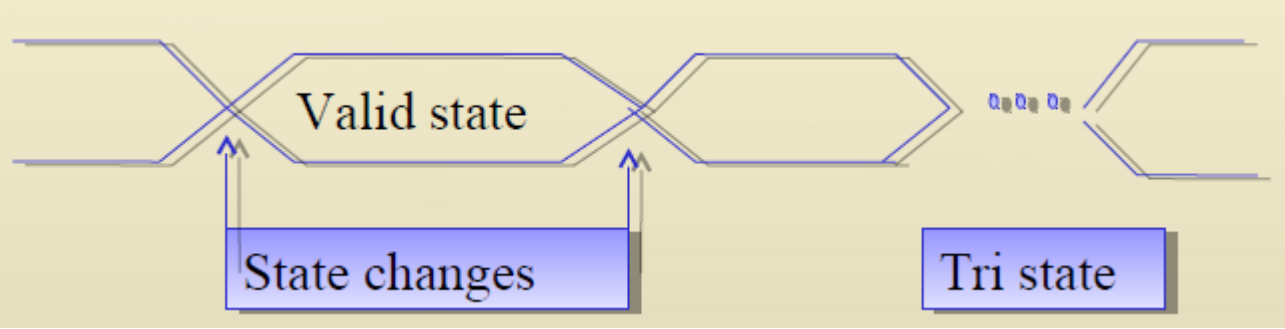


Timing Diagrams

Group of signals

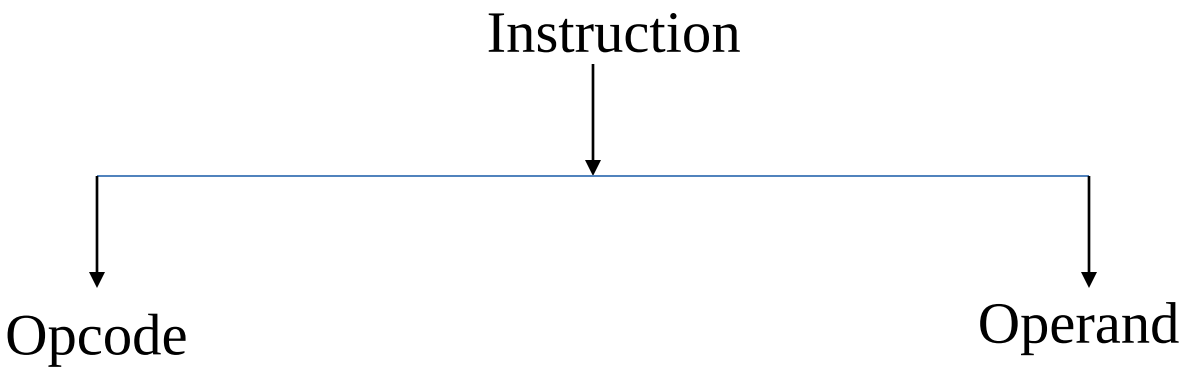
- Group of signals is also called a bus.

  Eg: Address bus, data bus



Timing Diagrams

Instruction



Example

MOV A, B

MVI B, 64H

Opcode Fetch Machine Cycle (4T or 6T)

- The first step of executing any instruction is the Opcode fetch cycle.
- In this cycle, the microprocessor brings in the instruction's Opcode from memory.

1) T1

Microprocessor uses IO/M , S0, S1 to instruct processor to fetch opcode, 16 bit address & ALE signal.

2) T2

Uses read signal & make data ready from that memory location to read the opcode & PC is incremented by one. And also check READY signal. If READY signal = 0: wait state between T2 & T3.

Opcode Fetch Machine Cycle (4T or 6T)

3) T3

Microprocessor reads opcode & store into the instruction register to decode it further.

4) T4

Microprocessor performs internal operation.

1) Decoding opcode (T5 or T6 is required or not)
2) Providing necessary action

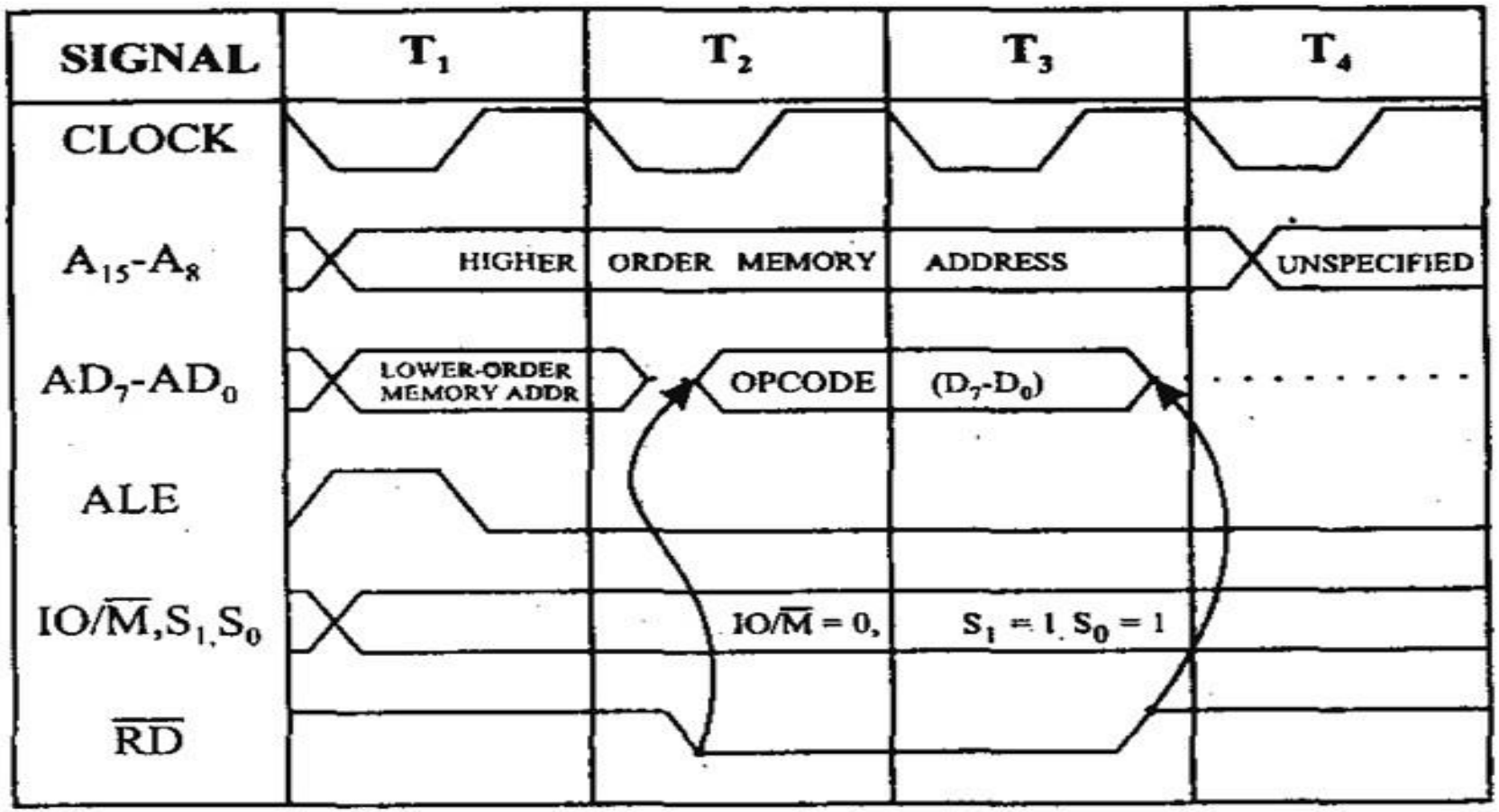
Timing Diagrams

Opcode Fetch Machine Cycle (4T or 6T)

- Instructions having opcode fetch machine cycle is of 6T states.
    - INX & DCX
    - Ccondtional, Rconditional & CALL
    - PCHL & SPHL
    - RSTn
    - PUSH
- Rest of instructions required 4T states to execute opcode fetch machine cycle.

Timing Diagrams

Timing Diagram for Opcode Fetch Machine Cycle



Timing Diagrams

Memory Read Machine Cycle

- The memory read machine cycle is exactly the same as the opcode fetch except:
  - It only has 3 T-states.
  - The s0 signal is set to 0 instead.
- The memory read machine cycle is executed by the processor to read a data byte from memory.
- The processor takes 3T states to execute this cycle.
- The instructions which have more than one byte word size will use the machine cycle after the opcode fetch machine cycle.

Timing Diagrams

Memory Read Machine Cycle

- To understand the memory read machine cycle, let's study the execution of the following instruction:

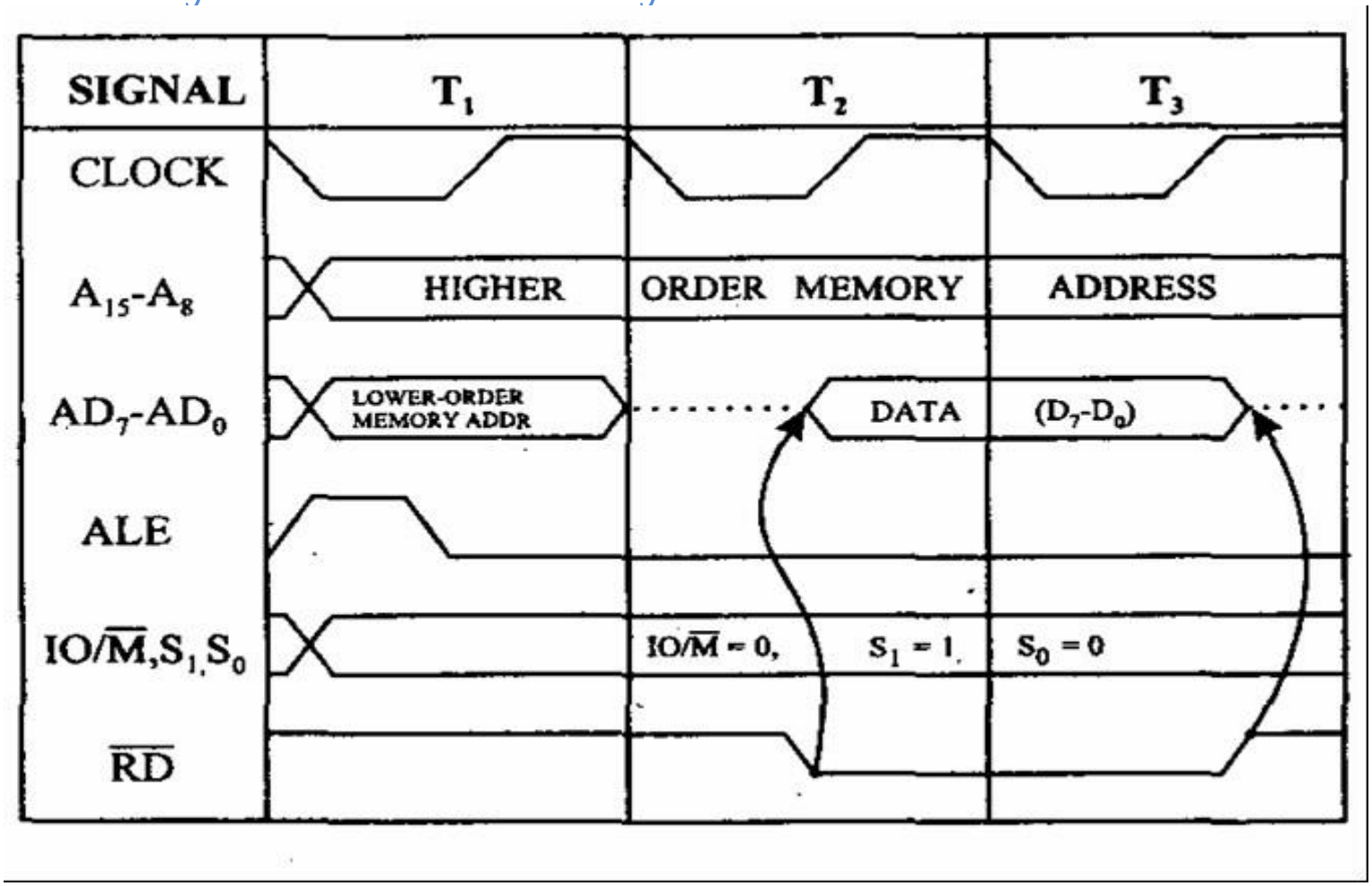
  MVI A, 32H

In memory, this instruction looks like:

- The first byte 3EH represents the opcode for loading a byte into the accumulator (MVI A), the second byte is the data to be loaded.
- The 8085 needs to read these two bytes from memory before it can execute the instruction. Therefore, it will need at least two machine cycles.
- The first machine cycle is the opcode fetch discussed earlier.
- The second machine cycle is the Memory Read Cycle.

Timing Diagrams

Memory Read Machine Cycle

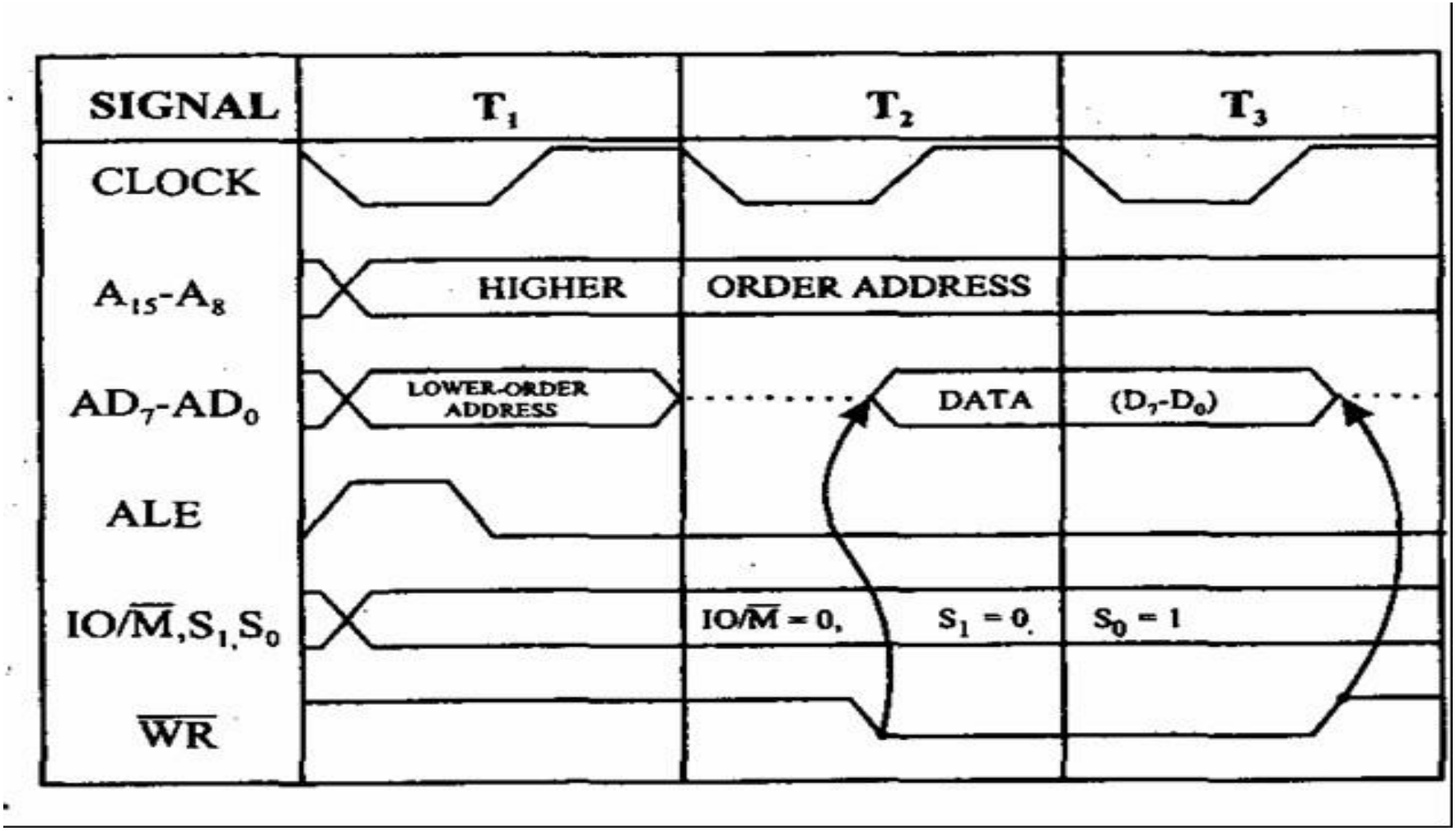


Timing Diagrams

Memory Write Machine Cycle

- The memory write machine cycle is executed by the processor to write a data byte in a memory location.
- The processor takes, 3T states to execute this machine cycle.
- In a memory write operation:
  - The 8085 places the address (2065H) on the address bus Identifies the operation as a memory write (IO/M=0, s1=0, s0=1).
  - Places the contents of the accumulator on the data bus and asserts the signal WR.
  - During the last T-state, the contents of the data bus are saved into the memory location.

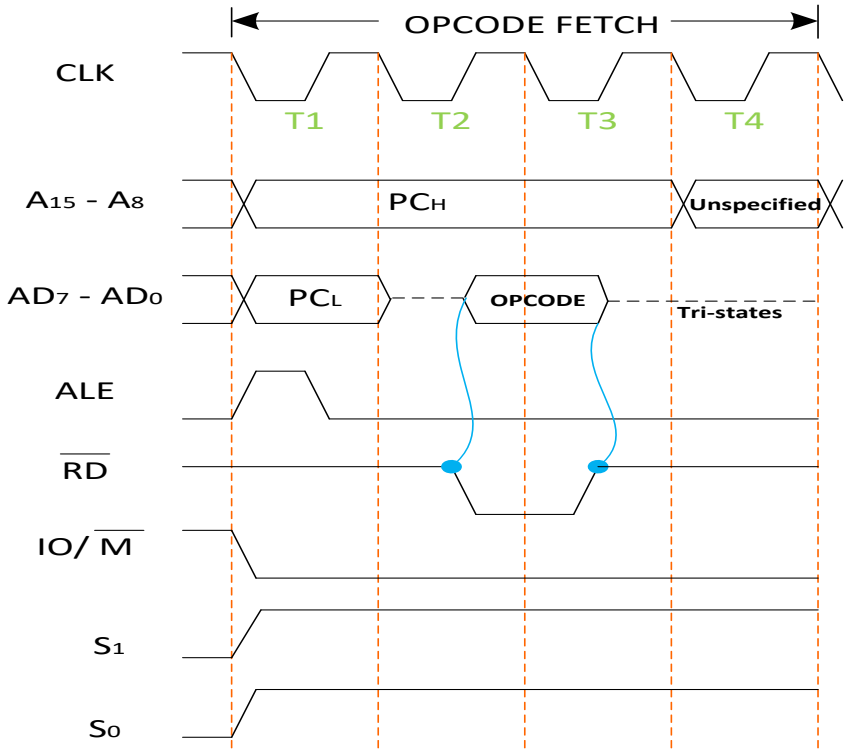
Timing Diagrams



Timing Diagram for Memory Write Machine Cycle

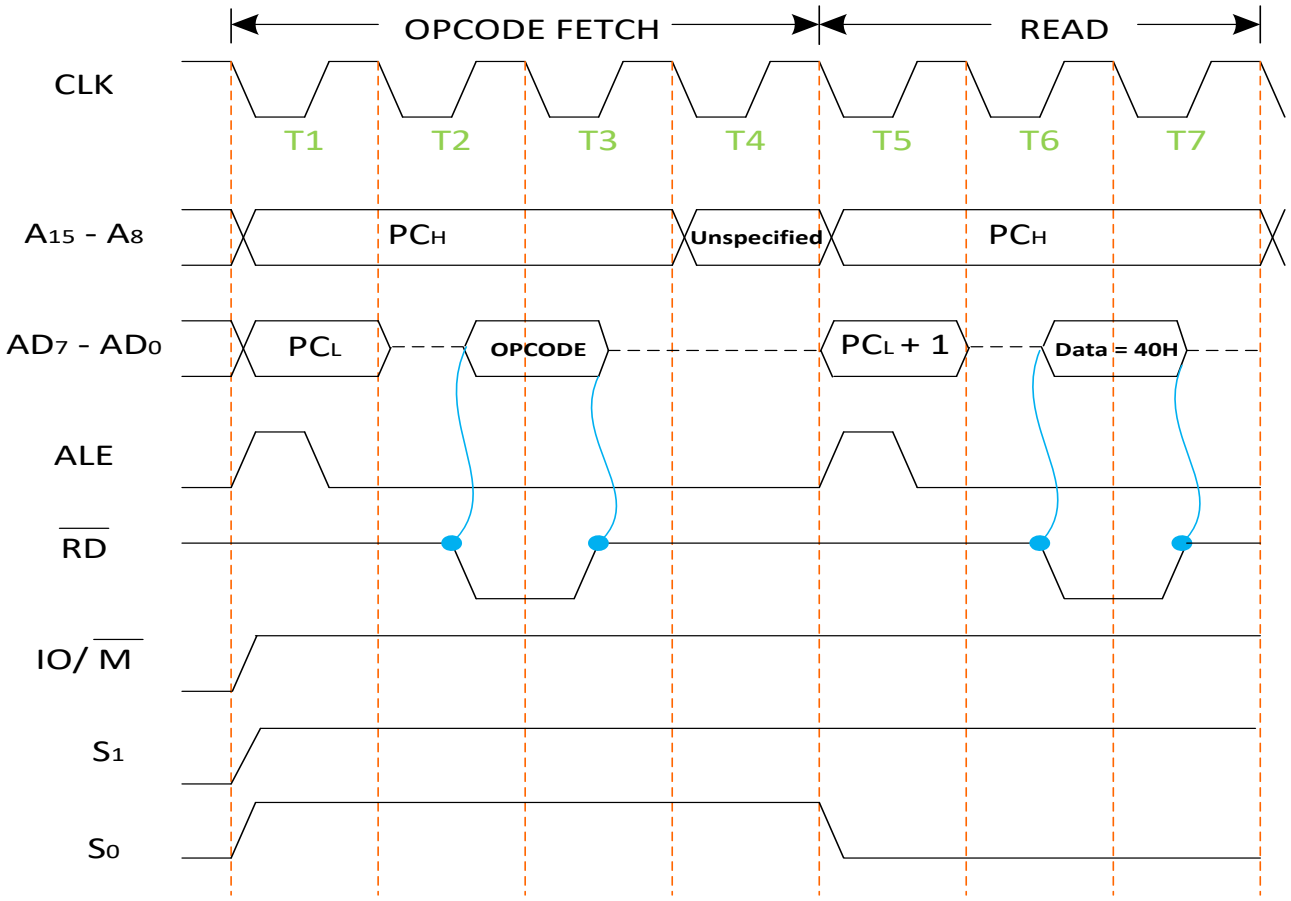
Timing Diagrams

MOV A,B	, ANA B	, ORA B	, XRA B
ADD B	, ADC B	, SUB B	, SBB B
RAL	, RAR	, RLC	, RRC
RIM	, SIM	, EI	, DI
NOP	, CMA	, CMC	, STC
INR B	, DCR B	, CMP B	, XCHG
DAA			



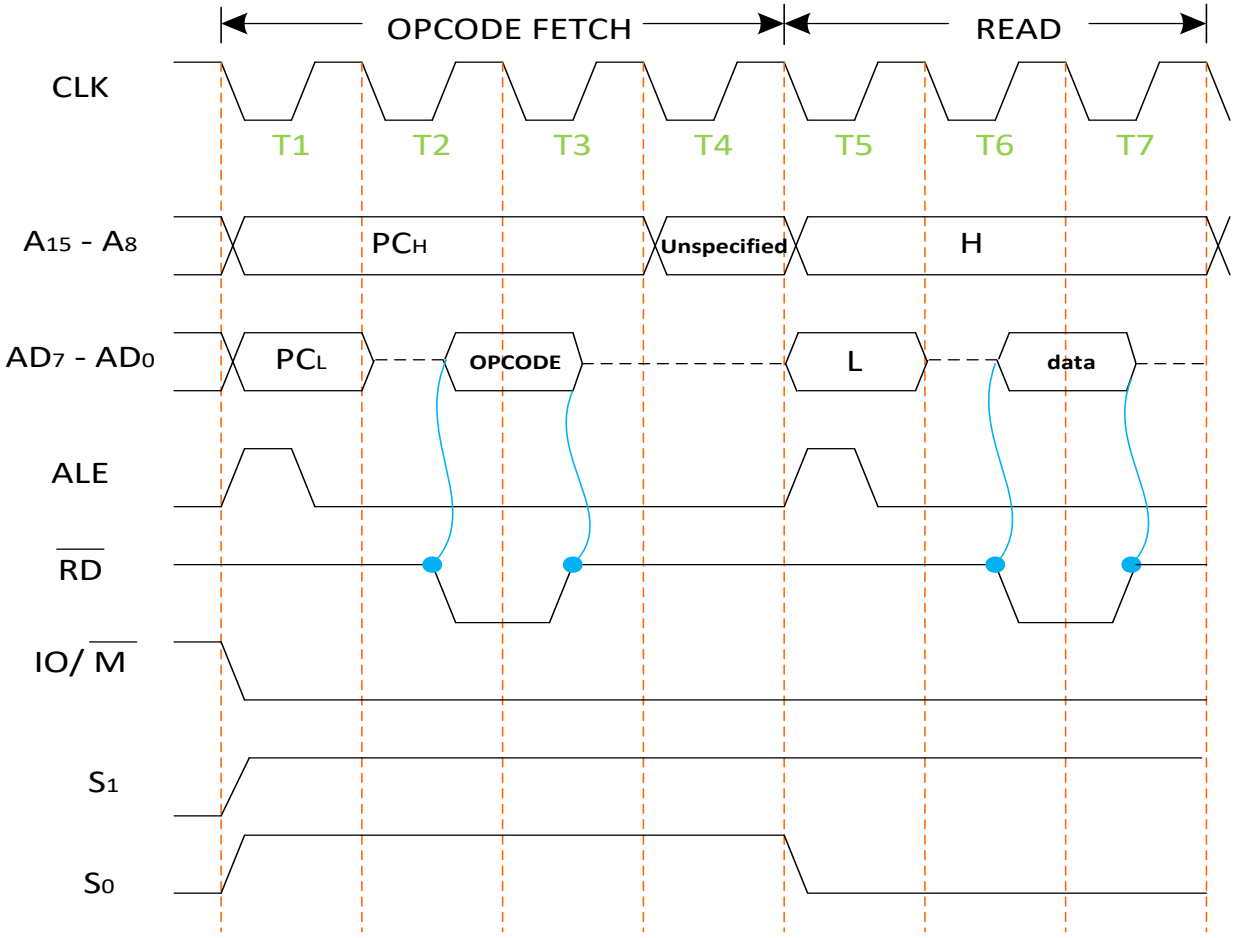
Timing Diagrams

MVI A,40H , ADI 80H , ACI 30H , SUI 40H , SBI 20H
ANI 30H , ORI 40H , XRI 75H , CPI 90H



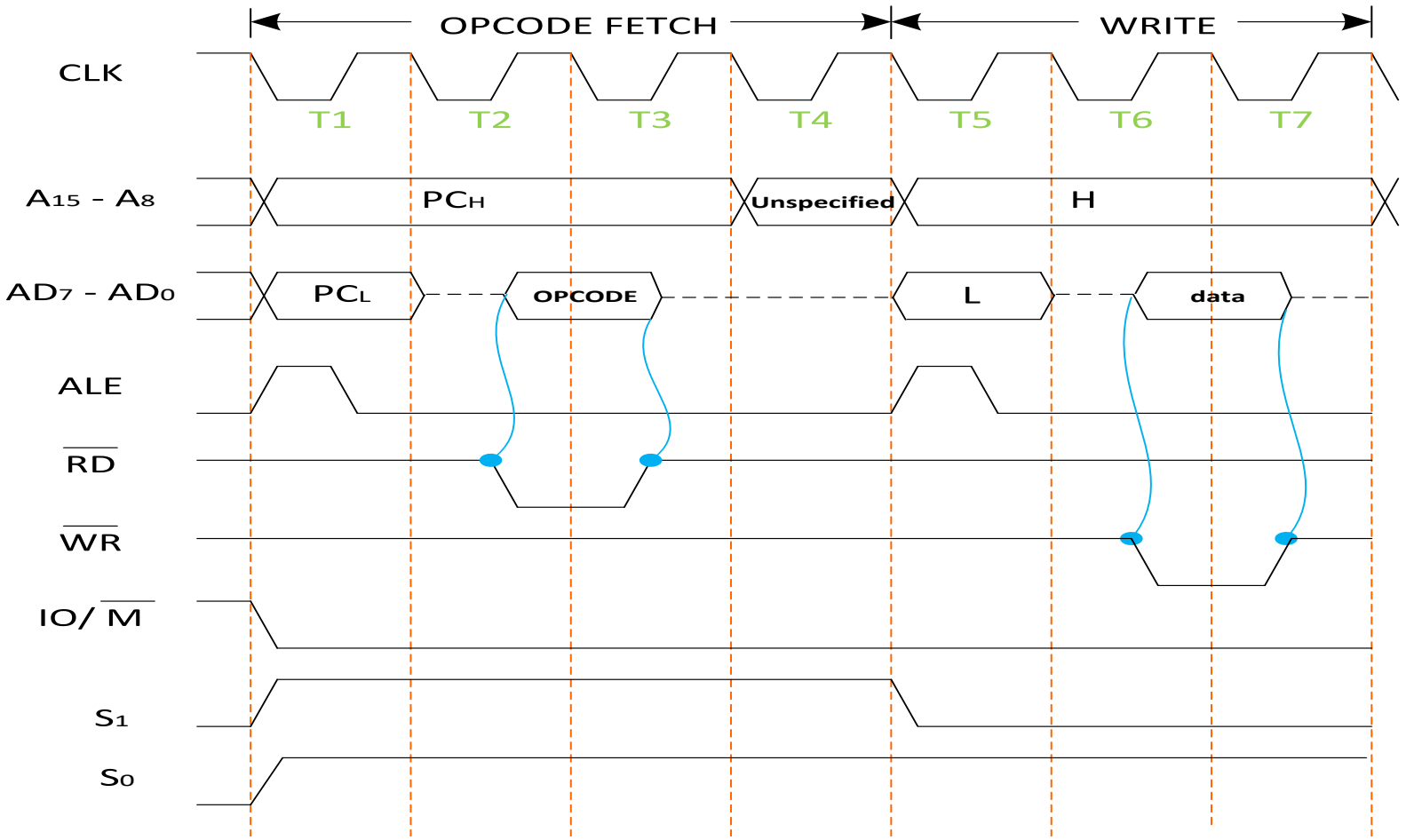
Timing Diagrams

MOV A,M , ADD M , ADC M , SUB M , SBB M
ANA M , ORA M , XRA M , CMP M



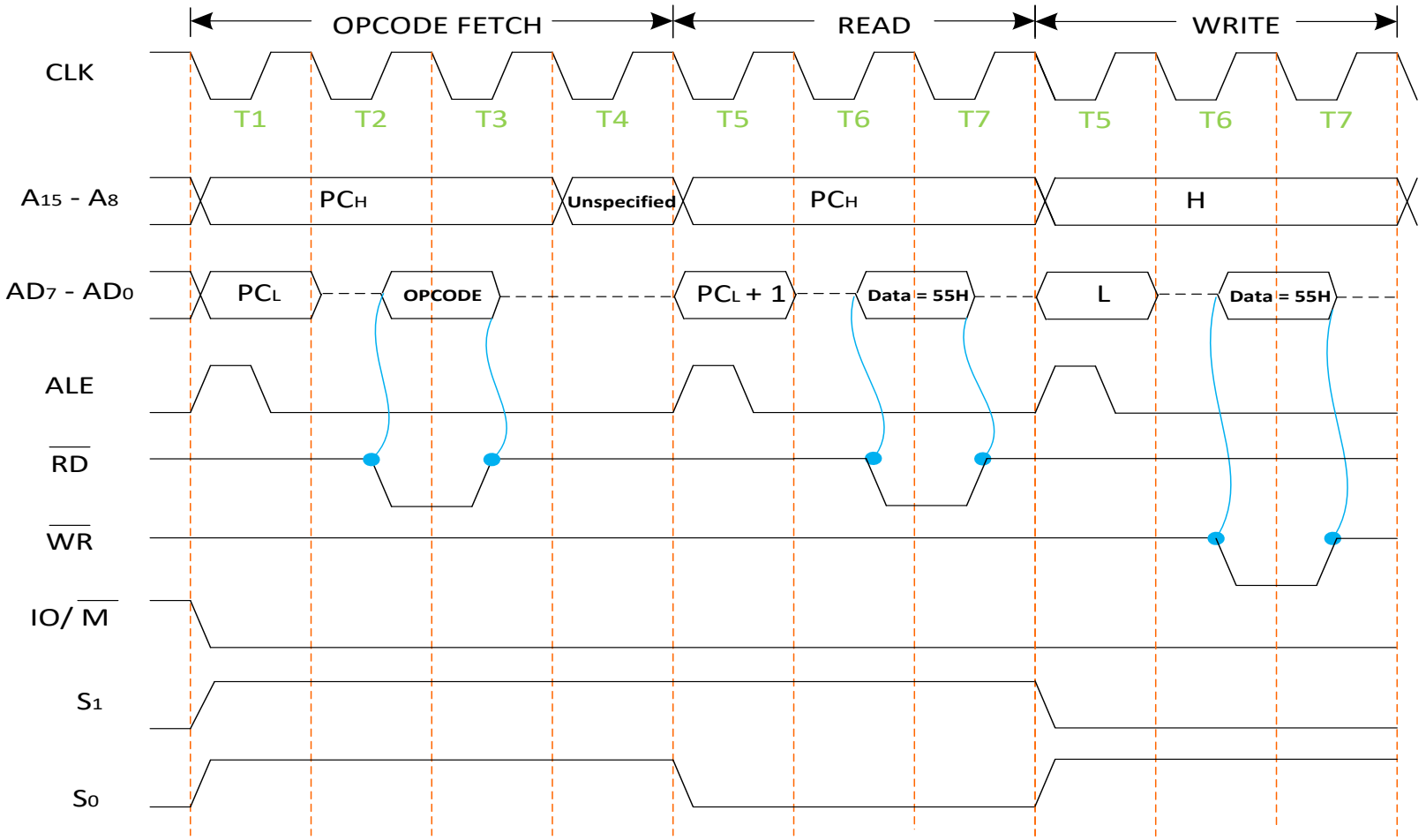
Timing Diagrams

MOV M,A

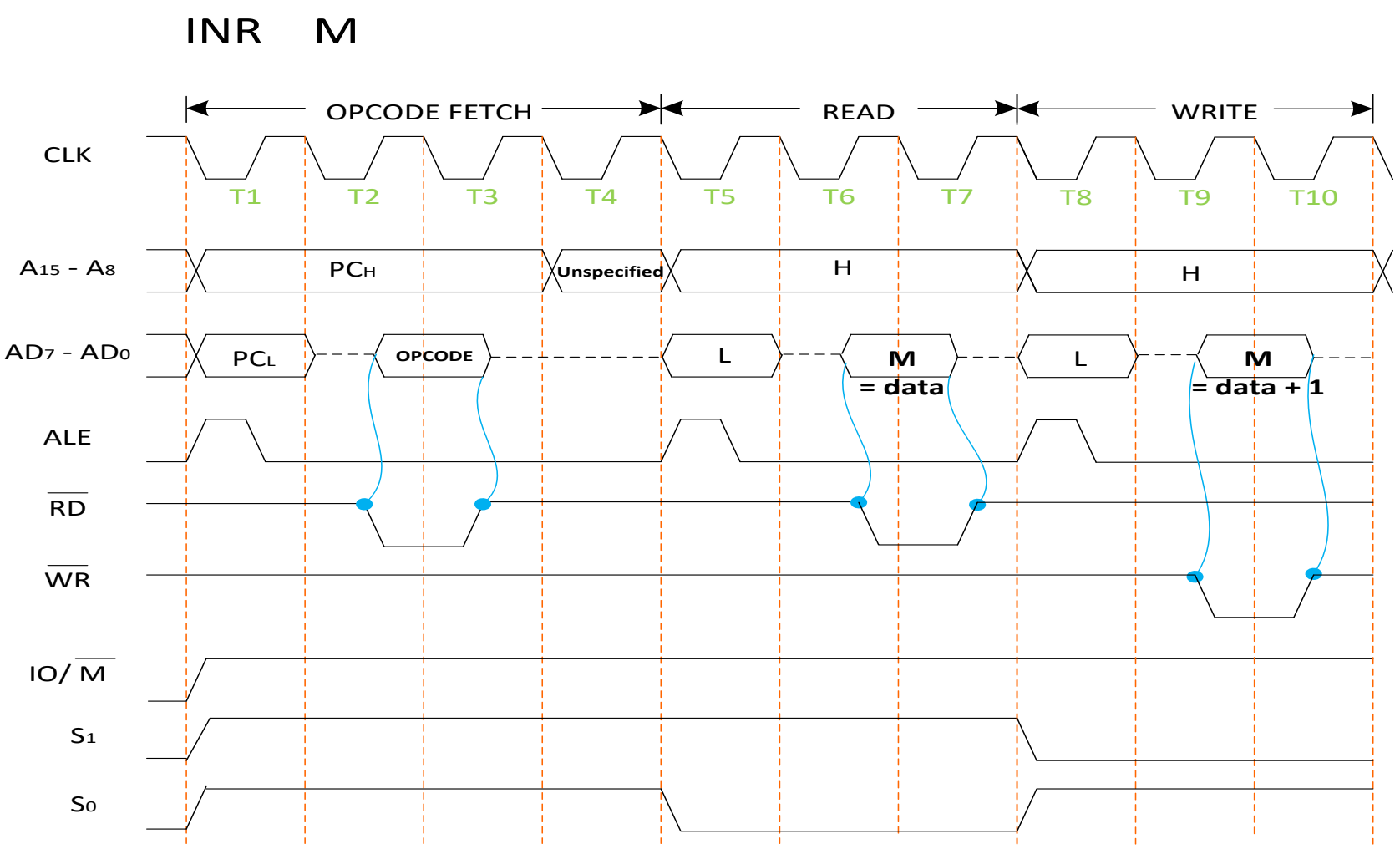


Timing Diagrams

MVI M,55H

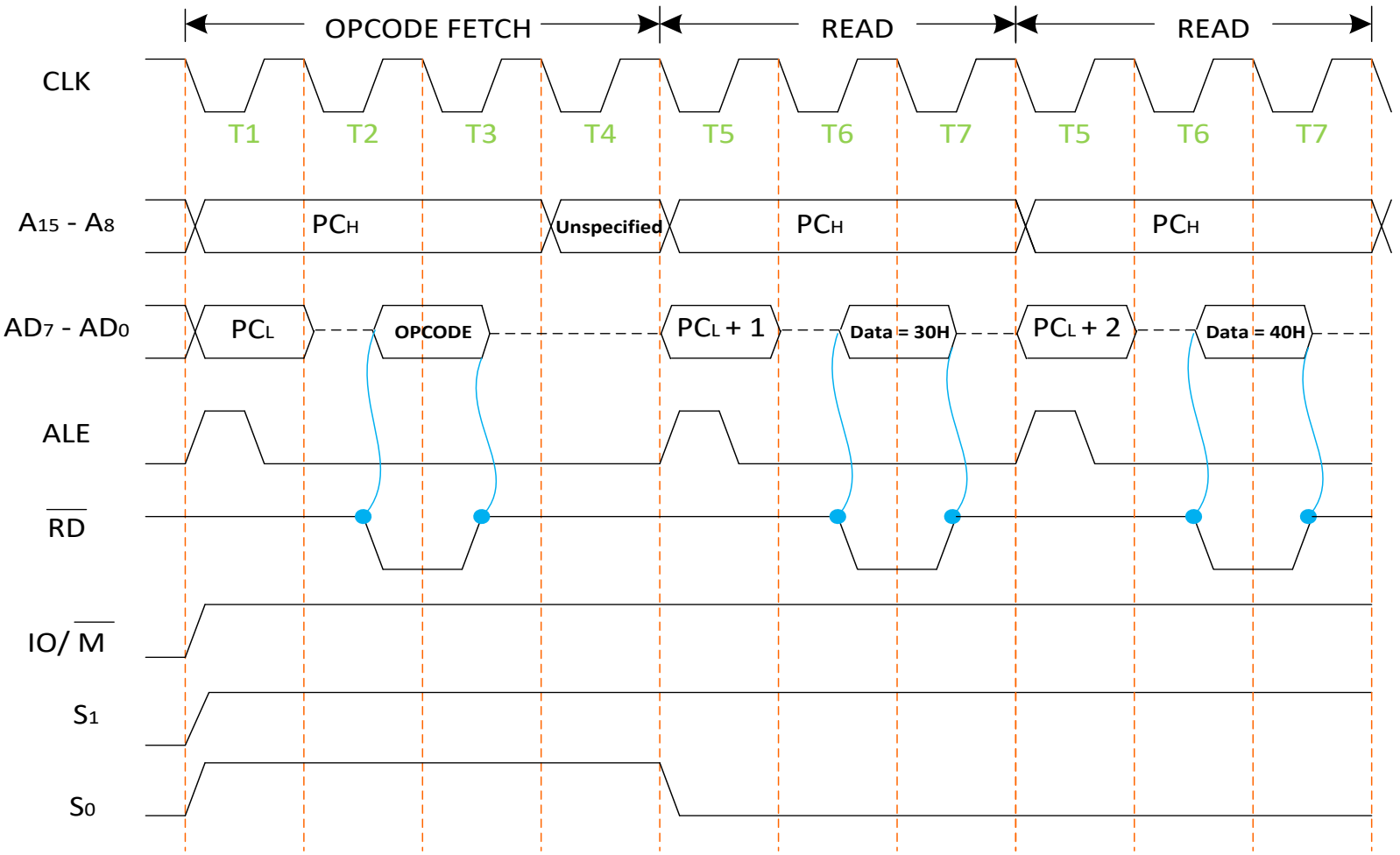


Timing Diagrams



Timing Diagrams

LXI H,4030H



Topic-8 Objectives

Name of the Topic	Objective of the topic	Mapping with CO
Interrupt	To understand the Interrupt function of 8085.	CO1

Interrupt

- Interrupts are signals send by an external device to the processor, to request the processor to perform a particular task or work.
- Mainly in the microprocessor based system the interrupts are used for data transfer between the peripheral and the microprocessor.
- The processor will check the interrupts always at the 2nd T-state of last machine cycle.
- If there is any interrupt it accept the interrupt and send the INTA (active low) signal to the peripheral.
- The vectored address of particular interrupt is stored in program counter.
- The processor executes an interrupt service routine (ISR) addressed in program counter.
- It returned to main program by RET instruction.

Interrupt

Types of Interrupts: It supports two types of interrupts.

1. Hardware interrupts
2. Software interrupts

- **Hardware interrupts** in the descending order of their priority :
  TRAP (RST 4.5) , RST 7.5 , RST 6.5 , RST 5.5 , INTR

- **Software interrupts :**
  RST 0 , RST 1 , RST 2, RST 3, RST 4 , RST 5 , RST 6 , RST

Types of Interrupt

Based on address:

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

Based on execution:

- **Maskable interrupt** − 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 interrupt** − In this type of interrupt, we cannot disable the interrupt by writing some instructions into the program.
  For example: TRAP.

Interrupts

Calculation of vector address

Vector Address = 8 * no of interrupt

Example

RST 5 = 8 * 5= 40_{10} = 28_{16}

Cont..

Software interrupts: The software interrupts are program instructions. These instructions are inserted at desired locations in a program.

- The 8085 has eight software interrupts from RST 0 to RST 7. The vector address for these interrupts can be calculated as follows.

The Table shows the vector addresses of all interrupts.

Interrupt	Vector address
RST 0 RST 1	0000_H 0008_H
RST 2 RST 3	0010_H 0018_H
RST 4 RST 5	0020_H 0028_H
RST 6 RST 7	0030_H 0038_H

Cont..

Hardware interrupts:

- An external device initiates the hardware interrupts and placing an appropriate signal at the interrupt pin of the processor.
- If the interrupt is accepted then the processor executes an interrupt service routine.

The 8085 has five hardware interrupts:

(1) TRAP (2) RST 7.5 (3) RST 6.5 4) RST 5.5 5) INTR

Interrupt	Vector address
RST 7.5	$003C_H$
RST 6.5	0034_H
RST 5.5	$002C_H$
TRAP	0024_H

Cont..

TRAP:

- This interrupt is a non-maskable interrupt. It is unaffected by any mask or interrupt enable.
- TRAP has the highest priority and vectored interrupt.
- TRAP interrupt is **edge and level triggered**. This means that the TRAP must go high and remain high until it is acknowledged.

Cont..

RST 7.5:

- The RST 7.5 interrupt is a maskable interrupt.
- It has the second highest priority.
- It is **edge sensitive**. ie. Input goes to high and no need to maintain high state until it recognized.
- Maskable interrupt. It is disabled by,
  1. DI, SIM instruction
  2. System or processor reset.
  3. After reorganization of interrupt with high priority .
- Enabled by EI instruction.

Cont..

RST 6.5 and 5.5:

- The RST 6.5 and RST 5.5 both are **level triggered**. ie. Inputs goes to high and stay high until it recognized.
- Maskable interrupt. It is disabled by,
  1. DI, SIM instruction
  2. System or processor reset.
  3. after reorganization of interrupt with high priority.
- Enabled by EI instruction.
- The RST 6.5 has the third priority whereas RST 5.5 has the fourth priority.

Cont..

INTR:

- INTR is a maskable interrupt. It is disabled by,
  1. DI, SIM instruction
  2. System or processor reset.
  3. After reorganization of interrupt with high priority .
- Enabled by EI instruction.
- Non- vectored interrupt
- It has lowest priority.
- It is a **level sensitive interrupts**. ie. Input goes to high and it is necessary to maintain high state until it recognized.

.

Daily Quiz

- The INTR interrupt may be masked using the flag
  a) direction flag
  b) overflow flag
  c) **interrupt flag**
  d) sign flag
- NMI stands for
  a) **non-maskable interrupt**
  b) non-multiple interrupt
  c) non-movable interrupt
  d) none of the mentioned
- The Programmable interrupt controller is required to
  a) handle one interrupt request
  b) **handle one or more interrupt requests at a time**
  c) handle one or more interrupt requests with a delay
  d) handle no interrupt request

Daily Quiz

- An interrupt breaks the execution of instructions and diverts its execution to
  a) **Interrupt service routine**
  b) Counter word register
  c) Execution unit
  d) control unit
- If any interrupt request given to an input pin cannot be disabled by any means then the input pin is called
  a) maskable interrupt
  b) **non-maskable interrupt**
  c) maskable interrupt and non-maskable interrupt
  d) none of the mentioned
- The INTR interrupt may be
  a) **maskable**
  b) non-maskable
  c) maskable and non-maskable
  d) none of the mentioned

Weekly Assignment

1. The 8085 microprocessor is executing a program given below:

```
2000:  MVI A, 10 H
       MVI B, 04 H
LOOP:  ADD B
       RLC
       RLC
       DCR B
       JNZ LOOP
       HLT
```

How many times ADD B operation takes place and final result of Accumulator?

2. An 8085 is executing the following program:

```
2000: LXI H 4325H
LXI SP 3000H
MOV A, H
ADD L
PUSH PSW
HLT
```

At the end of the program execution, what will be the contents of the PSW with stack location.

Weekly Assignment

3. Read the following program and answer the questions

```
2000 LXI SP, 2100H
2003 LXI B,0000H
2006 PUSH B
2007 POP PSW
2008 LXI H,200BH
200B CALL 2064H
200E OUT 01H
2010 HLT

DELAY: 2064 PUSH H
       2065 PUSH B
       2066 LXI B,80FFH
LOOP:  2069 DCX B
       206A MOV A,B
       206B ORA C
       206C JNZ LOOP
       206F POP B
       2070 RET
```

a). What is the status of the flags and the contents of the accumulator after the execution of the POP instruction located at 2007H?

b). What are the contents of the stack pointer and program counter registers after the execution of the CALL instruction?

Topic Links

https://www.youtube.com/watch?v=79icCUmqyPc

https://www.youtube.com/watch?v=1aG3aFEKxyA

Topic-9 Objective

Name of the Topic	Objective of the topic	Mapping with CO
Addressing Modes	To understand the role of Addressing Modes in 8085 microprocessor.	CO1

Addressing Modes

The method by which the address of source of data or the address of destination of result is given in the instruction is called **Addressing Modes.**

The term addressing mode refers to the way in which the operand of the instruction is specified.

Addressing modes in 8085 is classified into 5 groups −

1. Immediate addressing mode
2. Register addressing mode
3. Direct addressing mode
4. Indirect addressing mode
5. Implied addressing mode

Cont..

1. Immediate Addressing Mode:

In immediate addressing mode the source operand is always data. If the data is 8-bit, then the instruction will be of 2 bytes, if the data is of 16-bit then the instruction will be of 3 bytes.

Examples:

MVI B,45 (move the data 45H immediately to register B)

LXI H,3050 (load the H-L pair with the operand 3050H immediately)

JMP address (jump to the operand address immediately)

2. Register Addressing Mode:

In register addressing mode, the data to be operated is available inside the register(s) and register(s) is(are) operands. Therefore the operation is performed within various registers of the microprocessor.

Examples:

MOV A, B (move the contents of register B to register A)

ADD B (add contents of registers A and B and store the result in register A)

INR A (increment the contents of register A by one)

Cont..

3. Direct Addressing Mode:

In direct addressing mode, the data to be operated is available inside a memory location and that memory location is directly specified as an operand. The operand is directly available in the instruction itself.

Examples:

LDA 2050 (load the contents of memory location into accumulator A)

LHLD address (load contents of 16-bit memory location into H-L register pair)

IN 35 (read the data from port whose address is 35)

4. Register Indirect Addressing Mode:

In register indirect addressing mode, the data to be operated is available inside a memory location and that memory location is indirectly specified by a register pair.

Examples:

MOV A, M (move the contents of the memory location pointed by the H-L pair to the accumulator)

LDAX B (move contents of B-C register to the accumulator)

Cont..

5. Implied/Implicit Addressing Mode:

In implied/implicit addressing mode the operand is hidden and the data to be operated is available in the instruction itself.

Examples:

CMA (finds and stores the 1's complement of the contents of accumulator A in A)

RRC (rotate accumulator A right by one bit)

RLC (rotate accumulator A left by one bit)

Daily Quiz

- The instruction, Add #45,R1 does _____
  a) Adds the value of 45 to the address of R1 and stores 45 in that address
  b) **Adds 45 to the value of R1 and stores it in R1**
  c) Finds the memory location 45 and adds that content to that of R1
  d) None of the mentioned
- Add #45, when this instruction is executed the following happen/s _____
  a) The processor raises an error and requests for one more operand
  b) **The value stored in memory location 45 is retrieved and one more operand is requested**
  c) The value 45 gets added to the value on the stack and is pushed onto the stack
  d) None of the mentioned
- The addressing mode which makes use of in-direction pointers is _____
  a) **Indirect addressing mode**
  b) Index addressing mode
  c) Relative addressing mode
  d) Offset addressing mode

Daily Quiz

In the following indexed addressing mode instruction, MOV 5(R1), LOC the effective address is _____

a) EA = 5+R1
b) EA = R1
c) EA = [R1]
d) **EA = 5+[R1]**

The addressing mode, where you directly specify the operand value is _____

a) **Immediate**
b) Direct
c) Definite
d) Relative

The addressing mode/s, which uses the PC instead of a general purpose register is _____

a) Indexed with offset
b) Relative
c) **Direct**
d) Both Indexed with offset and direct

Topic Links

https://www.youtube.com/watch?v=1m-jgtGetl4

https://www.youtube.com/watch?v=VK3wnxEbaqI

Daily Quiz

- This signal is used as the system clock for devices connected with the microprocessor.
  A. X1, X2
  B. CLK OUT
  C. CLK IN
  D. IO/M
- Which of the following is true about Control and status signals?
  A. These signals are used to identify the nature of operation.
  B. There are 3 control signal and 3 status signals.
  C. Three status signals are IO/M, S0 & S1.
  D. All of the above
- How many pins of 8085 microprocessor includes?
  A. 39
  B. 40
  C. 20
  D. 25

Daily Quiz

- The 8085A has interrupt pins:-
  a) TRAP, RST7.5
  b) RST6.5, RST5.5
  c) TNTR (pin 10)
  d) All of the above.
- In the 8085A microprocessor, the data size is 8-bit and the address size is 16-bit.
  a) B-C pair
  b) D-E pair
  c) H-L pair
  d) All of the above.
- A microprocessor to execute a program, the CPU has to do the following operations:
  a) Fetch the opcode
  b) Read a memory location for the data.
  c) Perform the required operation
  d) All of the above.

Weekly Assignment

1. Explain general architecture of microprocessor.
2. Sketch the internal architecture of 8085 microprocessor.
3. How many signal groups are in 8085 pin.
4. What is the function of CLK OUT pin of 8085.

Topic Links

https://www.youtube.com/watch?v=STnM8a4hUII

https://www.youtube.com/watch?v=4PemFcazH_A

https://www.youtube.com/watch?v=UWekjor55pc

MCQ

- In static memory, the upper 8-bit bank of an available 16-bit memory chip is called
  - a) upper address memory bank
  - b) even address memory bank
  - c) static upper memory
  - d**) odd address memory bank**
- To obtain 16-bit data bus width, the two 4K*8 chips of RAM and ROM are arranged in
  - a**) parallel**
  - b) serial
  - c) both serial and parallel
  - d) neither serial nor parallel
- To avoid loading during read operation, the device used is
  - a) latch
  - b) flipflop
  - c) buffer
  - **d) tristate buffer**

MCQ

- The CPU sends out a _____ signal to indicate that valid data is available on the data bus:
  A. Read
  B. Write
  C. Both A and B
  D. None of these
- A microprocessor retries instructions from :
  A. Control memory
  B. Cache memory
  C. Main memory
  D. Virtual memory
- EU STAND FOR:
  A. Execution unit
  B. Execute unit
  C. Exchange unit
  D. None of these

MCQ

- The CPU removes the ___ signal to complete the memory write operation:
  - **A. Read**
  - B. Write
  - C. Both A and B
  - D. None of these
- Which RAM is created using MOS transistors:
  - **A. Dynamic RAM**
  - B. Static RAM
  - C. Permanent RAM
  - D. SD RAM
- The RAM which is created using bipolar transistors is called:
  - A. Dynamic RAM
  - **B. Static RAM**
  - C. Permanent RAM
  - D. DDR RAM

MCQ

- Which one of the following is not a vectored interrupt?

a. TRAP
b. INTR
c. RST 7.5
d. RST 3

- In 8085 microprocessor, the RST6 instruction transfer programme execution to following location

a. 0030H
b. 0024H
c. 0048H
d. 0060H

- HLT opcode means

a. load data to accumulator.
b. store result in memory.
c. load accumulator with contents of register.
d. end of program.

MCQ

- In 8085 name/names of the 16 bit registers is/are

a. stack pointer.
b. program counter.
c. both A and B
d. None of the above

- What is SIM?

a. Select interrupt mask.
b. Sorting interrupt mask.
c. Set interrupt mask.
d. None of these.

- The ROM programmed during manufacturing process itself is called

a. MROM
b. PROM
c. EPROM
d. EEPROM

MCQ

- A mask programmed ROM is

a. programmed at the time of fabrication
b. programmed by the user
c. erasable and programmable
d. erasable electrically

- The program counter in a 8085 micro-processor is a 16-bit register, because

a. It counts 16-bits at a time
b. There are 16 address lines
c. It facilitates the user storing 16-bit data temporarily
d. It has to fetch two 8-bit data at a time

- A microprocessor is ALU

a. and control unit on a single chip.
b. and memory on a single chip.
c. register unit and I/O device on a single chip.
d. register unit and control unit on a single chip.

Glossary Questions

Fill in the blanks with correct options:

Options: a) 1951, b) 1960, c) 1940, d) 1971

1. The first digital electronic computer was built in _____.

   (1940)

2. The Texas institute invented IC in _____.

   (1960)

3. Microprocessor was invented in _____.

   (1971)

Old Question Papers (Sessional &University)

- Not Available

Expected Questions

1. With the help of neat diagram explain the architecture of 8085 microprocessor in detail.
2. Explain the sequence of events during the execution of the CALL instruction by 8085 processor with the help of neat timing diagram.
3. Write an assembly language program with comment lines. An 8-bit number is stored in memory location C100H. Count number of ones (i.e. 1) in this byte and store this count in memory location C200H.
4. Draw and explain the timing diagram of memory write cycle with example.
5. Draw and explain the timing diagram of opcode fetch cycle.
6. Explain Functions of PIN configuration of 8085 microprocessor.

Recap of Unit

- 8085 microprocessor internal architecture, PIN Diagram, Timing Diagram, Addressing Modes, Interrupts are discussed.
- Various logic devices are discussed in detail for interfacing with 8085 microprocessor.

Thank You!