

**Microprocessor :-** Central processing (CPU) built on a single integrated circuit (IC) is called a microprocessor. A digital computer with one microprocessor which acts as a CPU is called Microcomputer.

It is a programmable, multipurpose clock-driven, register-based electronic device that reads binary instructions from a storage device called memory, accept binary data as input & process data according to those instruction & provides results as output.

The microprocessor contains millions of tiny components like transistors, registers, & that work together.

**Machine Language :-** Machine language is a low-level language made up of binary numbers or bits that a computer can understand. It is also known as machine code or object code and is extremely tough to comprehend. The only language that the computer understands is machine language. Such as C++, produce our run programmes in machine language before they are run on a computer when a specific task even the smallest program executes.

Machine language is translated to the system process computers are only able to understand binary data as they are digital devices.

**Assembly language :-** Assembly language is a low-level programming language that helps the computer there is an assembler that helps in converting the assembly code into machine code executable.

**Assembler :-**

The Assemblers are used to translate the assembly language into machine language.

## Low Level Languages

The Low level Language is a programming language that provides no abstraction from the hardware & it is represented in 0 or 1 terms which are the Machine Instruction. The languages that come under this category are machine Level Language & assembly language.

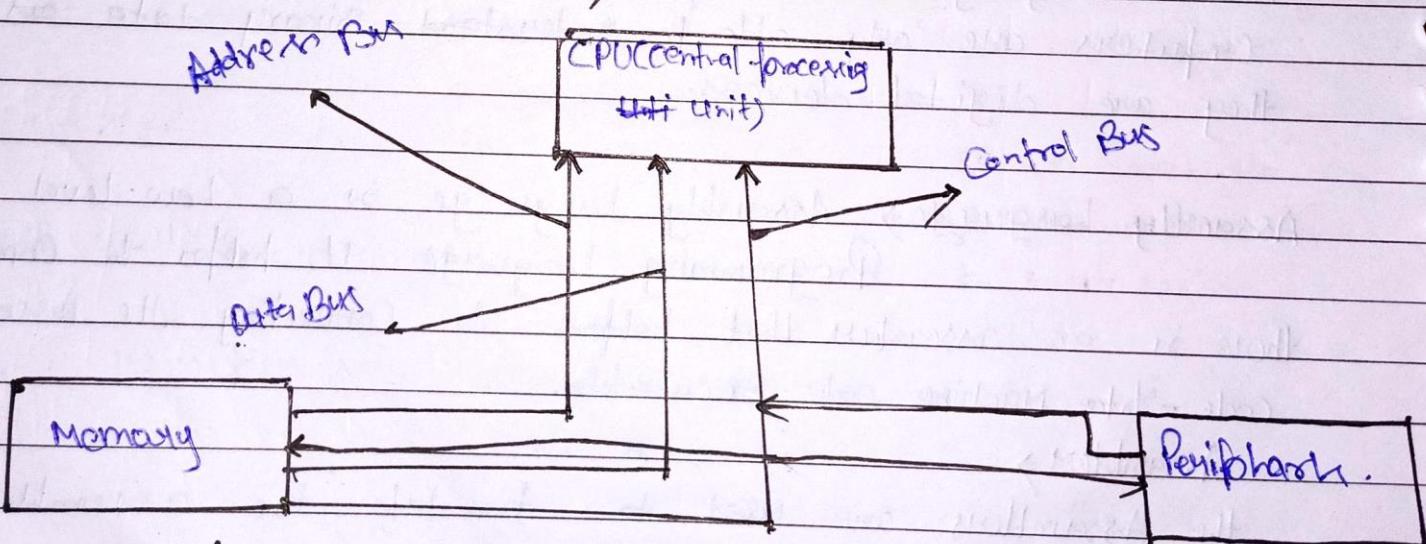
## High Level Languages

The high level language is a Programming Language that allows a programmer to write the programs which are independent of a particular type of Computer. The high-level languages are considered as high-level because they are closer to human languages than machine-level language.

**Interpreters** → An Interpreter translates the entire source code line by line.

**Compilers** → A Compiler translates the entire source code in a single run.

## Basic computer organization



- \* Bus used for connection of CPU, Memory & peripherals
- \* Address Bus gives address to memory.
- \* Control Bus gives control to peripherals & memory.

## External Architecture 8085 $\Rightarrow$

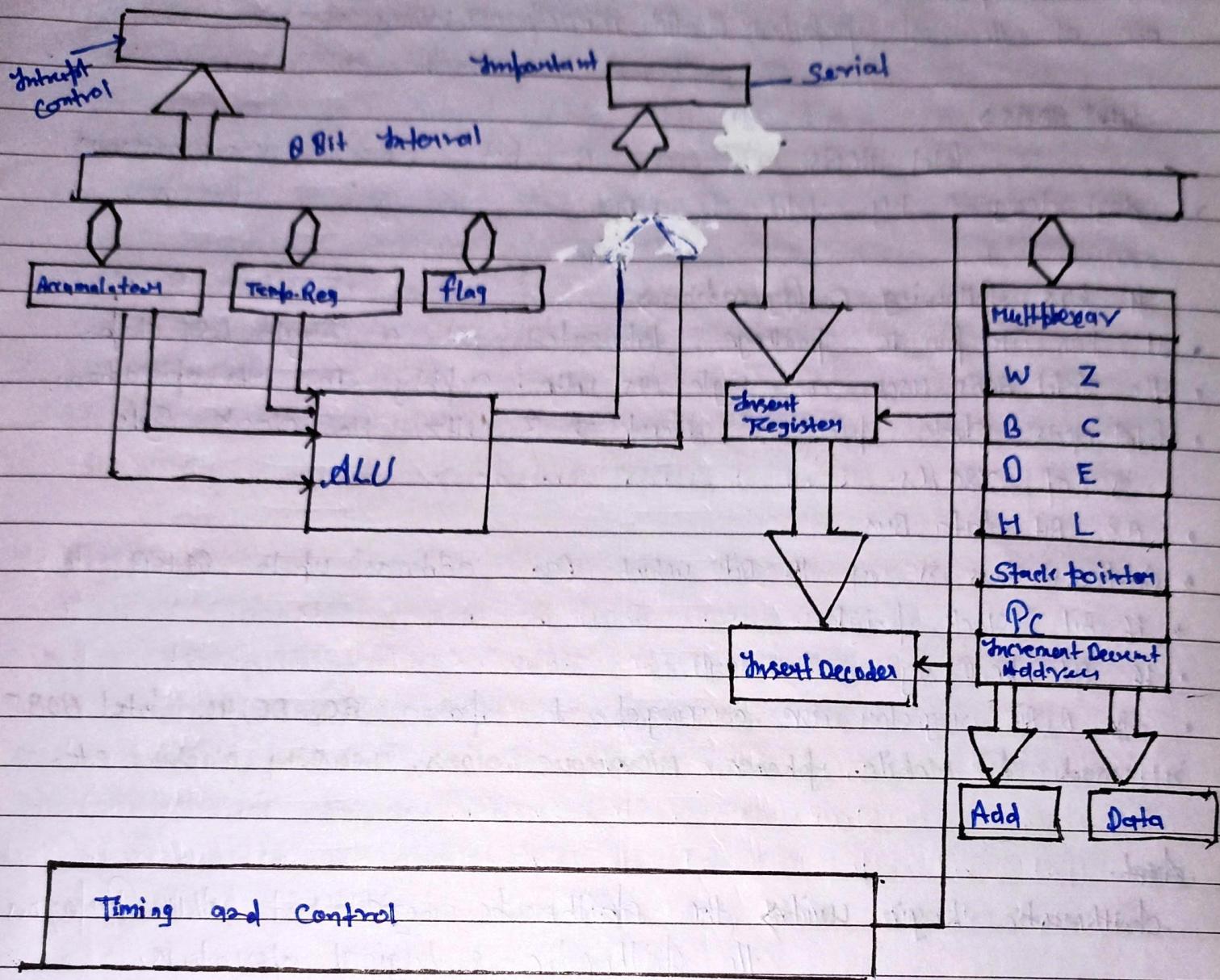
$X_1$	1	40	Vcc
$X_2$	2	39	Hold
Reset out	3	38	HLDN
SD	4	37	CLK Cout
SIO	5	36	Results
TRST	6	35	Ready
RST 7.5	7	34	I/O/m
RST 6.5	8	33	S1
JNTR	9	32	V
JNTR	10	31	RD IP WR
PD <sub>0</sub>	11	30	BS/BSB
AD <sub>1</sub>	12	29	WR
AD <sub>2</sub>	13	28	S0
AD <sub>3</sub>	14	27	A15
AD <sub>4</sub>	15	26	A14
AD <sub>5</sub>	16	25	A13
AD <sub>6</sub>	17	24	A12
AD <sub>7</sub>	18	23	A11
AD <sub>8</sub>	19	22	A10
V <sub>SS</sub>	20	21	A9 A8

The pins of A 8085 microprocessor can be classified into Seven groups

Address Bus  $\Rightarrow$  AD<sub>0</sub>-AD<sub>7</sub>, It carries the Most Significant 8 bits of Memory to address.

Data Bus  $\Rightarrow$  AD<sub>8</sub>-AD<sub>15</sub>, It carries that Least Significant 8 bit address and data bus.

## Architecture of microprocessor (8085)



The microprocessor is the CPU of a computer. It is the heart of the computer. Here, we will describe Intel 8085 as it is one of the most popular 8 bit microprocessor.

### Intel 8085

Intel 8085 is an 8 bit Microprocessor designed by Intel in 1972.

It has following configurations:

- It has 40 pin IC package fabricated on a single LSI chip.
- The Intel 8085 uses a single +5 volt supply for its operation.
- Intel 8085 clock speed is about 3 MHz; the clock cycle is of 320 ns.
- 8 Bit data Bus
- Address Bus is of 16 Bit which can address up to 64 KB
- 16 Bit Stack pointer.
- 16 Bit PC (Program Counter)
- Six 8 Bit register are arranged in pairs: BC, DE, HL. Intel 8085 is used in mobile phones, microwave oven, washing machine etc.

### Arch

Arithmetic Logic Unit, the Arithmetic logic unit also performs the arithmetic & logical operations.

1. Addition
2. Subtraction.
3. Logical AND
4. Logical OR
5. Logical Exclusive OR
6. Complement (Logical Not)
7. Increment (Logical Not) (Add1)
8. Decrement (Subtract1)
9. Left Shift, Rotate left, Rotate right.
10. Clear etc.

## Timing And control units

the timing and control unit is the section of the CPU.

- It is used to generate timing and control signal which are necessary for the execution of instructions.
- It is used to control data flow between CPU & Peripherals (including memory)
- It is used to provide status control and timing signals which are required for the operation of memory and I/O devices.
- It is used to control the entire operations of the microprocessor and peripherals connected to it.

thus we can see that the Control Unit of the CPU acts as the brain of the computer system.

## Registers

Registers are used for temporary storage and manipulation of data and instruction by the microprocessor. Data remain in Registers till they are sent to the I/O devices or memory. Intel 8085 Microprocessor has the following registers.

- \* One <sup>8bit</sup> Accumulator (ACC) i.e. register A.
- \* Six General purpose registers of 8 bit these are BCDEH & L.
- \* One 16 Bit Stack pointer SP.
- \* One 16 Bit Program Counter, PC.
- \* Instruction Register.
- \* Temporary Register.

In addition to the above mentioned registers the 8085 Microprocessor contains a set of five flip-flops which serve as flags (or status flags).

A flag is a flip-flop which indicates some conditions which arises after the execution of an arithmetic or logical instruction.

a.1 Accumulators → the Accumulator is an 8-bit register associated with the Main Memory Register 'A' is an accumulator of the 8085. It is used to hold one of the operands of an arithmetic and logical operation.

b.2 General purpose registers

the 8085 microprocessor contains six 8-bit general purpose registers. They are. B,D,C,E,H & L register.

To hold data of 16 bit a combination of two 8-bit registers can be employed.

The combination of two 8-bit registers is called register pairs in the 8085 i.e. D-E, B-C and H-L.

The H-L pair is used to act as a memory pointer.

c.3 program counters → it is a 16-bit special purpose register.

It is used to hold the address of memory of the next instruction to be executed. If the track of the instruction in a program while they are being executed the microprocessor increments the content of the Next Program Counter during the executing of an instruction so that at the end of the execution of the instruction it points to the Next instruction's address in the program.

d. Stack pointers → it is a 16-bit special function

register used as memory pointer of stack is nothing but a pointer of RAM.

In the stack the contents of only those registers are saved which are needed in the later part of the program.

The stack pointer controls the addressing of the stack.

The stack pointer contains the address of the top element of data stored in the stack.

### c. Instruction Registers

The instruction register holds the opcode (operation code or instruction code) of the instruction which is being decoded and executed.

f. Temporary Registers → It is an 8 Bit register associated with the ALU. It holds data during an arithmetic logical operation. It is used by the microprocessor. It is not accessible to programmer.

Flags → The Intel 8085 Microprocessor contains five flags to serve as status flags. The flip-flops are reset or set according to the conditions which arise during an arithmetic or logical operation.

- Carry flag (C)
- Parity flag (P)
- Auxiliary carry flag (Ac)
- Zero flag (Z)
- Sign flag (S)

If a flip-flop for a particular flag is set then it indicates 1. When it is reset it indicates 0.

Date  
09/Sept/22

Architecture 8085  $\Rightarrow$  8085 is an 8 Bit Microprocessor which was developed in the year around the year 80.

- It is used to work on 5V power supply.
- Accumulator: It stores the content needed for performing arithmetic operation the result is often in ALU is stored in Accumulator.
- ALU: Arithmetic Logic Unit It performs arithmetic and logic operation.

Exps Addition Subtraction etc.

Instruction Register and decoder  $\Rightarrow$  when instruction is fetch from memory it is stored in Instruction Register.

Register and decoder decode the information present in Instruction Register.

Timing and Control unit  $\Rightarrow$  It provides timing and control signals to Microprocessor to perform operation.

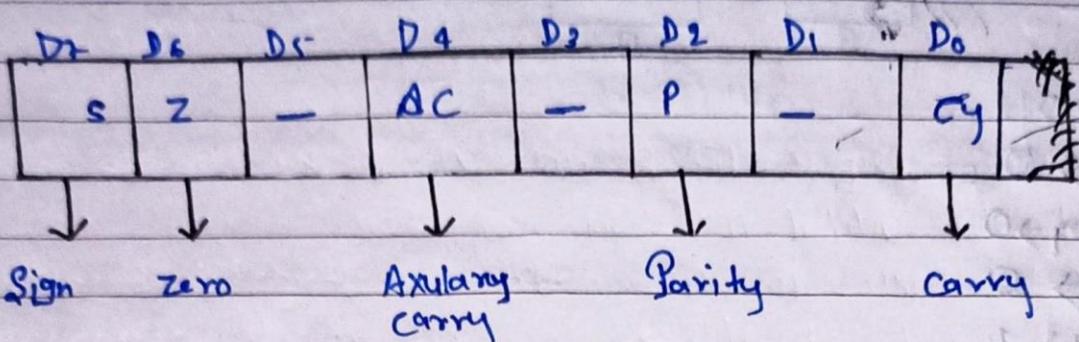
Stack  $\Rightarrow$  It works like stack which is always prefer to work on LIFO follows by push and pop operation.

Program Counter  $\Rightarrow$  It is used to store memory location Program counter. Instruction to be executed.

Serial Input output Control  $\Rightarrow$  It controls the serial data width has to be transferred.

## Components

8085 Flag Register  $\Rightarrow$



$$\begin{array}{r}
 + B C \\
 + D E \\
 \hline
 9 A
 \end{array}
 \quad
 \begin{array}{r}
 10111100 \\
 +11011110 \\
 \hline
 110011010
 \end{array}$$

Depending upon the value of result after any operation flag Register are set on Result.

- (1) Sign flag  $\Rightarrow$  when Most Significant D<sub>7</sub> is one than Sign flag value is one otherwise it is zero.
- (2) Zero flag  $\Rightarrow$  when you are performing any operation and the result is zero then zero flag is one otherwise it is zero.
- (3) Auxiliary carry  $\Rightarrow$  when the carry is generated from lower nibble to higher nibble the Auxiliary carry is one otherwise it is zero.
- (4) Parity flag  $\Rightarrow$  if the result there is even no of one when Parity flag value is one otherwise zero.

Carry flag  $\Rightarrow$  if the result after if there is any carry generated carry flag is value is one otherwise zero.

$$\begin{array}{r}
 33 \\
 + A 6 \\
 \hline
 09
 \end{array}
 \quad
 \begin{array}{ccccccccc}
 D_7 & D_6 & D_5 & D_4 & D_3 & D_2 & D_1 & D_0
 \end{array}$$

Sign  $\Rightarrow$  1  
Zero  $\Rightarrow$  0    Carry = 0.  
Auxiliary = 0  
Parity = 0

Date  
14/5/02

$$\begin{array}{r} + 45 \\ + FS \\ \hline 38 \end{array} \quad \begin{array}{r} 0100\ 0101 \\ + 1111\ 0011 \\ \hline 10011\ \underbrace{1}_{3}\ \underbrace{000}_{8} \end{array}$$

Sign  $\Rightarrow$  0

Zero  $\Rightarrow$  0

Auxiliary  $\Rightarrow$  0

Parity  $\Rightarrow$  0

Carry  $\Rightarrow$  1

$$\begin{array}{r} + AE \\ + 74 \\ \hline 122 \end{array} \quad \begin{array}{r} 10101110 \\ + 01110101 \\ \hline 100\Phi\Phi00011 \\ \quad \quad \quad \underbrace{1}_{1} \quad \underbrace{0}_{2} \end{array}$$

Sign  $\Rightarrow$  0  
Zero  $\Rightarrow$  0  
Auxiliary  $\Rightarrow$  0  
Parity  $\Rightarrow$  0  
Carry  $\Rightarrow$  1

$$\begin{array}{r} + 19 \\ + 01 \\ \hline 1A \end{array} \quad \begin{array}{r} 0001\ 1001 \\ + 0000\ 0001 \\ \hline 0001\ 1010 \end{array}$$

Sign  $\Rightarrow$  0

Zero  $\Rightarrow$  0

Auxiliary  $\Rightarrow$  0

Parity  $\Rightarrow$  0

Carry  $\Rightarrow$  0

$$\begin{array}{r} + FF \\ + 01 \\ \hline 00 \end{array} \quad \begin{array}{r} 1111\ 1111 \\ + 0000\ 0101 \\ \hline 10000\ 0000 \end{array}$$

Sign  $\Rightarrow$  0

Zero  $\Rightarrow$  0

Auxiliary  $\Rightarrow$  1

Parity  $\Rightarrow$  1

Carry  $\Rightarrow$  1

$$\begin{array}{r} + 1 \\ \hline 0 6 \\ \hline 2 0 \end{array}$$

$$\begin{array}{r} 00011010 \\ - 00000110 \\ \hline 00100000 \end{array}$$

2      0

Sign  $\rightarrow 0$

Zero  $\rightarrow 0$

Auxiliary  $\rightarrow 1$

Parity  $\rightarrow 0$

Carry  $\rightarrow 0$

11

Date  
20/2/2019

## (Microprocessor) (Registers)

General purpose  
Registers 8 bit

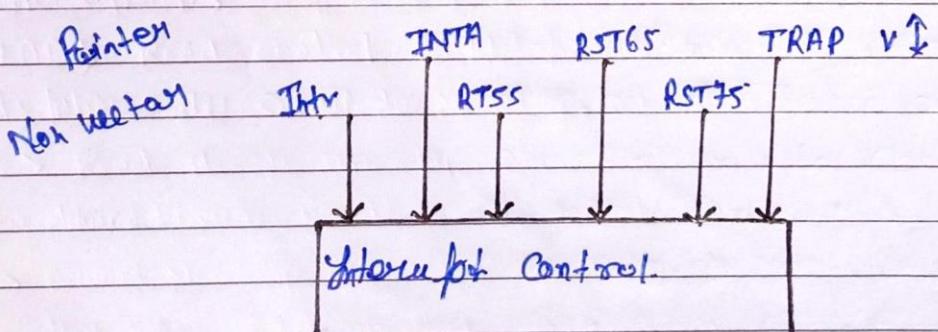
	W (Temp)	Z (Temp)
B	C	
D	E	
H	L	
S	SP	
C	PC	

Increment Decrement  
Address Latch.

W & Z Register  $\Rightarrow$  Both the Registers are to the same temporary data or contain.

General purpose Register  $\Rightarrow$  8085 has 8 bit general purpose Register store 8 bit data here can be combined as BC, DE, HL to perform some 16 bit data.

Special purpose Registers  $\Rightarrow$  It is use to implement or decrement the content of Stack.



Date  
20/Sept/22

Interrupt Control constant of RST75, RST65, RST55 INTR features  
doubt B As vector interrupt can non vector interrupt too if  
RST 6.5, 7.5, 5.5 they all are vector interrupt because they  
have one three memory location.

TRAP - 24H

RST75 - 3CH

" " 65 - 34H

" " 5.5 - 2CH

INTR is a non vector interrupt use ISR INTR interrupt  
knowledge. ISR (Interrupt Service Routine Register).

Service to get request file INTR

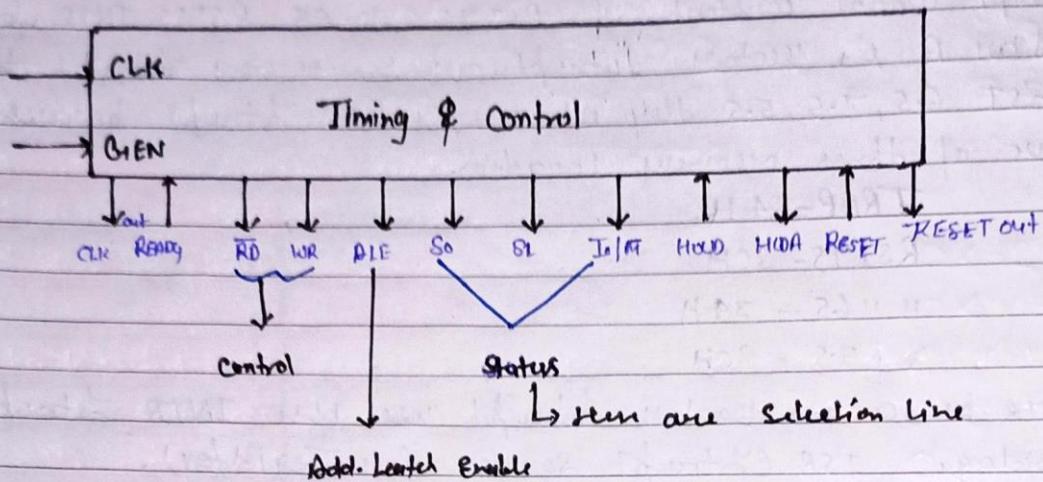
Serial Input output control unit  $\Rightarrow$  SID/SOD  $\Rightarrow$

SID  $\Rightarrow$  Serial input output control in use for data communication  
and transfer.

SOD  $\Rightarrow$  Serial output data line.

Date  
8/5/22

Most important  $\Rightarrow$



Add. Latch Enable (ALE)  $\Rightarrow$  when ALE = 1 (High) Add. Bus Enable 0  
Data bus enable.

Status Signals  $\Rightarrow$  (1) Input output devices if distinctive field is  
for memory I/O devices when high Add.  
Bus I/O Low if for Add Memory.

So, etc  $\Rightarrow$  here static signals distinguish various types of operation  
ALE = 1 (high)

So	SI	operations
0	0	HALT
0	1	WRITE
1	0	READ
1	1	FETCH

Control Signals  $\Rightarrow$

READ  $\Rightarrow$  Read Control Read operation.

It defines the selected memory or I/O device is for Read.

WRITE  $\Rightarrow$  Read it is used by microprocessor to check whether device is ready to transfer data.

DMA  $\Rightarrow$  Direct Memory Access it is designed to transfer data directly to the memory without participation of CPU processor.

HOLD  $\Rightarrow$  It means another device is requesting for the use of Add. and data bus.

HLDA  $\Rightarrow$  It is user's job to check for Hold signal

Clock  $\Rightarrow$  It synchronizes the peripheral device

Rent  $\Rightarrow$  Rent the Program Counter.

Rent out  $\Rightarrow$  It indicates CPU is Renting

Date  
23/4/22

## Most Important

Addressing Modes. In 8085 Microprocessor  $\Rightarrow$

- (1) Immediate addressing mode.
- (2) Register addressing mode.
- (3) Direct addressing mode.
- (4) Indirect addressing mode.
- (5) Implied / Implicit addressing mode.

(1)  $\xrightarrow{\text{opcode}}$  Instruction execution by CPU is known as opcode.

(2)  $\xrightarrow{\text{operands}}$  Data or Memory location used to execute the instruction is known as operand.

Addressing Modes  $\Rightarrow$  Addressing Modes means transfer of data from one Register to another Register or Memory to same Register, there are different type of Addressing Modes.

(i) Immediate Add. Mode. (ii) Register Add. Mode. (iii) Direct Add. Mode  
in Indirect Add. Mode. (iv) Implicit Add. Mode.

1. Immediate addressing mode  $\Rightarrow$  In Immediate Addressing Mode data is present in the instruction operand comes immediately after the opcode.

Example  $\Rightarrow$  MVI45 (Move the data 45H immediately to register B).

LXI H 3050 (Load the H-L Pair with the operand 3050H Immediately)

2. Register Addressing mode  $\Rightarrow$  This Addressing Mode specifies data whose content is between Register or pair.

Exps (i) MVI B HC, (ii) ~~MVI BC~~ MVI BC.

3) Direct Addressing Mode  $\Rightarrow$  In direct the operand Add. is been directly copy or Move because be known the operand Add. ,

Exps LDA 2050, LHLD, IN35.

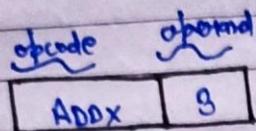
ADD X	IO
-------	----

$\curvearrowleft$   $\curvearrowleft$   
operand      opcode

Adder

Date  
23/5/2022

Indirect Addressing Mode  $\Rightarrow$  there is an intermediate value which is specified in operand field.



(5) Implied / Implicit Addressing Mode  $\Rightarrow$  it does not require any operand the data specified by opcode.

Ex: HLT, STOP.

Most Important

Assembly :-

Programming language are basically categorized into three types.

- (1) Assembly L.L.
- (2) High L.L.

Machine is More language because the parameters used to describe them consist more of the coded language which was less difficult for humans to read and remember.

Assembly languages

Assembly language is low level language or shows the representation of the ~~all~~<sup>all</sup> method on 0, 01 or 1 from. It same type means look like Machine Language.

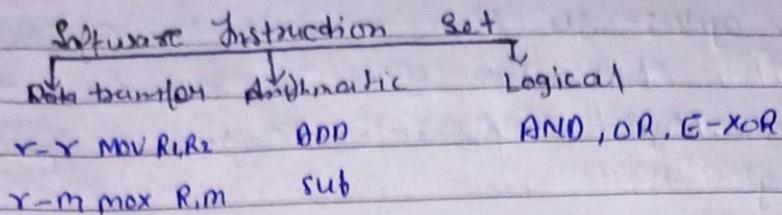
High level language:-

Since was low level language for high program high level language.

High level language ~~designed~~ by ~~desi~~.  
The first high ~~level~~ level programming language designed for computers was Plankalkül, created by Konrad Zuse.

Date  
02/Nov/12

## [Second Unit]



Software Instruction Set command given to the Computer to perform certain task is known as Software Instruction set. Here there are basically three types of Software Instruction set.

- (1) Data Transfer Instruction
- (2) Arithmetic Instruction set.
- (3) Logical Instruction set.

(1) Data transfer Instruction set:- Data transfer Instruction we can copy the data either between register to register or Register to memory.

Ex:-  $r-r \text{MOV } R_1, R_2$ .  
 $r-m \text{MOV } R, M$

(2) Arithmetic:- these instruction of 8085 performs Arithmetic operation such as Addition Subtraction BCD Addition etc.

Ex:- ADD, SUB, .. 20H.

(3) Logical:- these instruction which performs various logical operation are known as Logical Instruction set.

Ex:- AND, OR, E-XOR

AND, 20H A 30H

OR A.

Ex:-

MVIA, 12H

MVID, 23H

AND, D.

$$\begin{array}{r} 0001\ 0010 \\ 0010\ 0011 \\ \hline 0000\ 0010 \end{array}$$

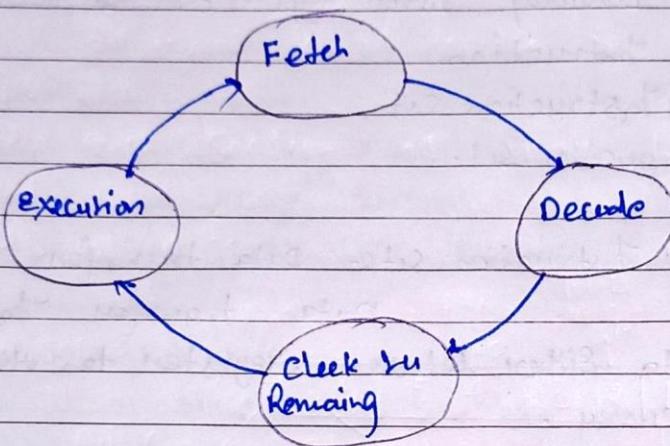
Date  
04/11/22

Date  
03/11/22

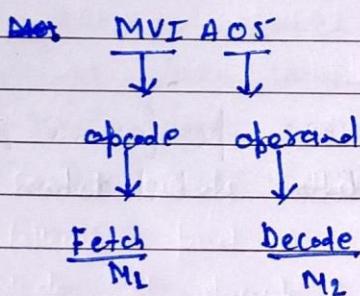
Instruction cycle :- The time required to complete an instruction given is known as Instruction cycle.

Instruction cycle consists of following process.

- (1) Fetching
- (2) Decoding
- (3) Check the Remaining or left data
- (4) Execution.

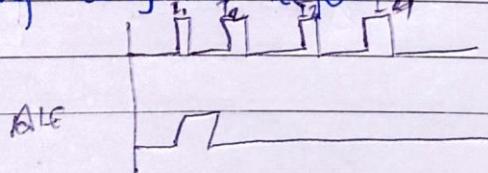


Ex :-



Machine Cycle :- It is the time required to complete one operation. Machine cycle consists of its state. Mean the operation has to be completed operation. If state is nothing But the clock fails.

Timing diagram - To keep the instruction in machine cycle in sync with the microprocessor we make the timing diagram for different control bus states. Signals timing diagram Ref:



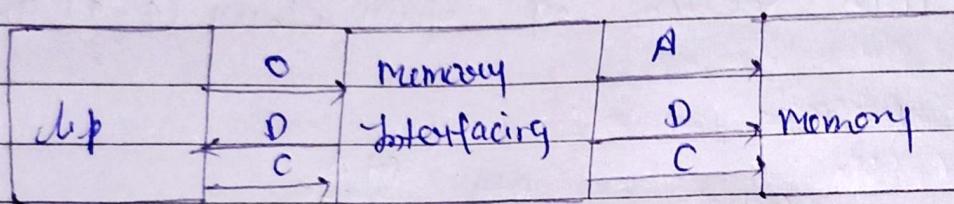
Date  
09/11/22

## ~~Computer System Architecture~~

Memory Interfacing & Memory interfacing Since Computer has to frequently used Instruction code and the data for the execution of Program memory interfacing is been done.

There are several steps for memory interfacing.

- (1) Selection of chip IIC
- (2) Selection of Register and Signals.
- (3) Appropriate Execution logic and Buffering is done.



D → Address Bus

D → Data Bus

C → Control Bus

Memory interfacing is done via to Bus

① Input output

② Memory map I/O

Input output

① work as an I/O device

② I/O map execute to Bus  
out instruction

③ I/O microprocessor and in  
microcontroller

Memory map I/O

(1) Memory device

(2) Execute all instruction

(3) complex and Expression

(4) memory map used for  
microcontroller.

Date  
15/11/12

### [Third Unit 8255]

QD 8255: 8255 is a programmable peripheral interface device.

Sun manufacturer also attached PIA.

(Programmable Interface) is a multi-pole device.

Depends upon program how it is operated.

It consists of three port Port A, Port B, Port C.

Port A is 8 Bit Port

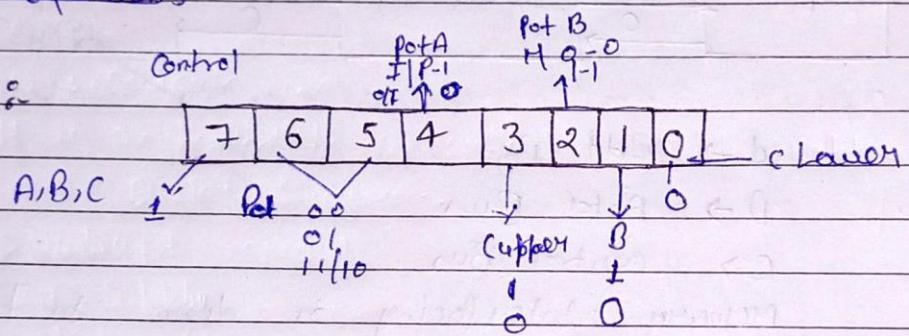
Port B is 8 Bit

Port C is 8 Bit but C is divided in two lower into  
e upper.

Hence now each section has four or four (4-4) bit

#### Control words of 8255

Control word of 8255:



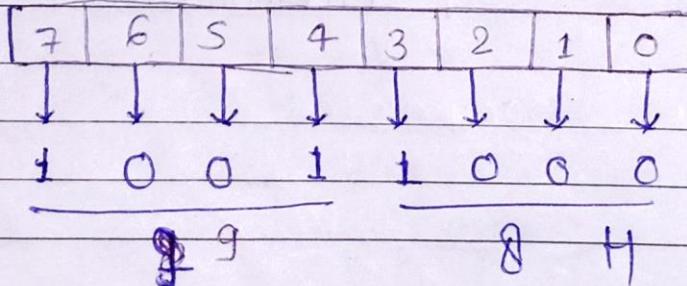
to make control words when the port of Intel 8255 are defined as follows port A as an input port mode of port A - mode 0

Port B as an output port

Mode of port B - mode 0

Port Cupper as an I/O port

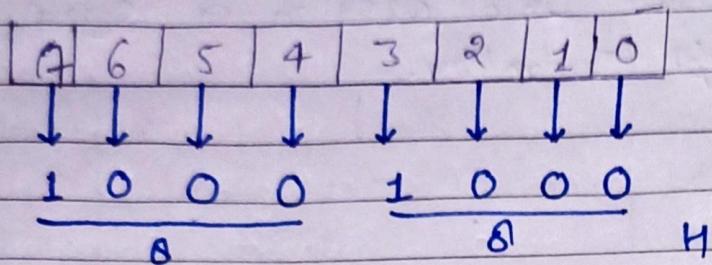
Port Claway as an output port



Date  
16/11/22

Most important:-

- (Q:- Form from control word for the following configuration of the port out 8255 for mode 0 operation  
Port A output Port B output Port C lower output  
Port C upper input.



8279:- Programmable key display interface :-

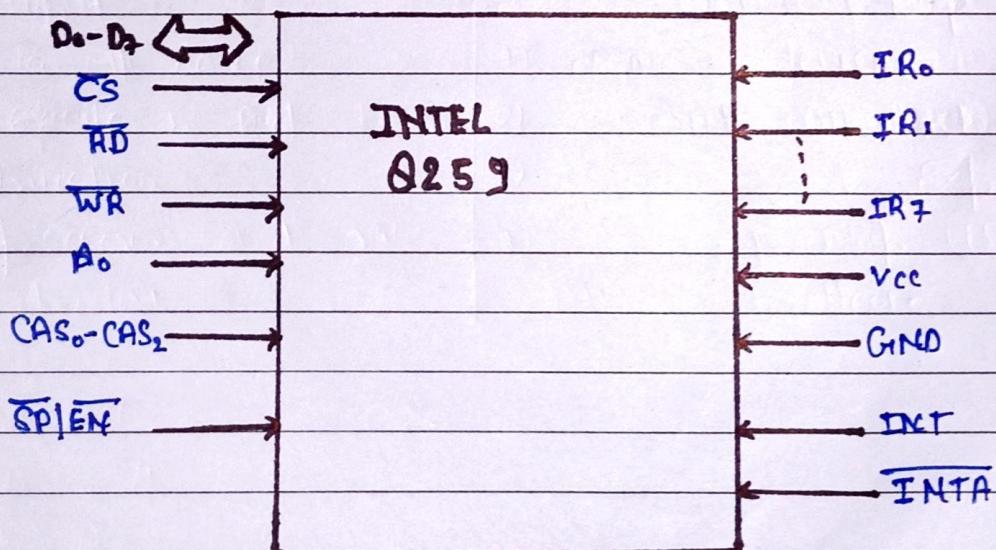
The Intel 8279 is a programmable keyboard interface device data input and display either integral part of Microprocessor Base System. 8279 has two section namely -

(1) Keyboard section and display section.

function of the keyboard section is to interface with the keyboard which is used as an input device for the microprocessor.

(2) the purpose of display section is to realize from the border of scanning the keyboard by referencing the display.

8259:-



Date  
16/11/22

8259 It is programmable Input Controller.

It is programmable Input Controller. It is used when several I/O device transfer data using interrupt and they are to be connected to some input of the microprocessor. When No of I/O devices are less than No of interrupt level in the microprocessor such controllers are not suitable. Intel 8259 is a signal source programmable interrupt controller.

The detail of pins:

D<sub>0</sub>-D<sub>7</sub> Data Bus

CS → Control Signals

RD → Read data

WR → write or Read

A<sub>0</sub> → Address.

CAS → CAS → Cascade line

SPIEN → It is repeated to cascade line. Sleep enable Buffer program.

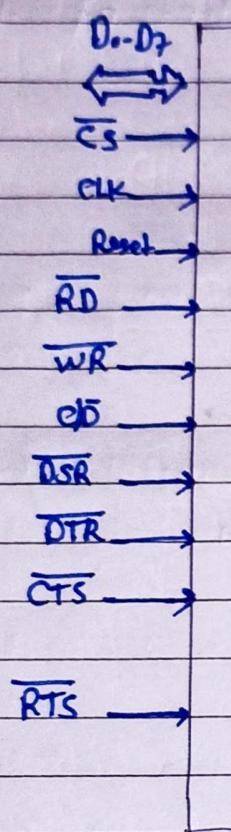
IRQ → INT → Interrupt request

Vcc → Power Supply

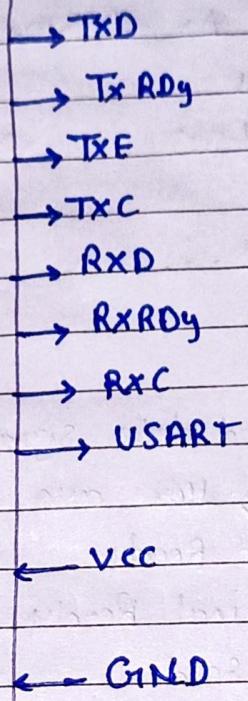
GND → Ground level.

↔ D-9

ATMEL  
P239



Q251



Date  
01/11/22

8251: is a programmable communication interface. It is also known as USA RT (Universal 9 Source Receiver transmitter). It accept data in parallel format and convert them into serial data for transmission - the description of pins in 8251 are as follows.

D<sub>0</sub>-D<sub>7</sub> data Bus

CS → chip select

CLK → clock

Reset RD and WR

RD → Read data

WR → write data

C/D → control and data signals

DSR, DTR, CTS, RTS → These are modem control signals

DSR → Data Set Ready.

DTR → Data terminal Ready.

CTS → clear to send.

RTS → Request to send.

TxD → Transmission data

T<sub>x</sub>Ready → Tx RDY → Transmitter ready

TxE → Transmitter

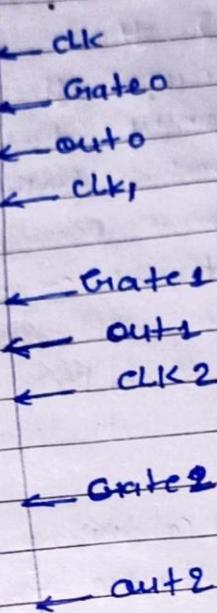
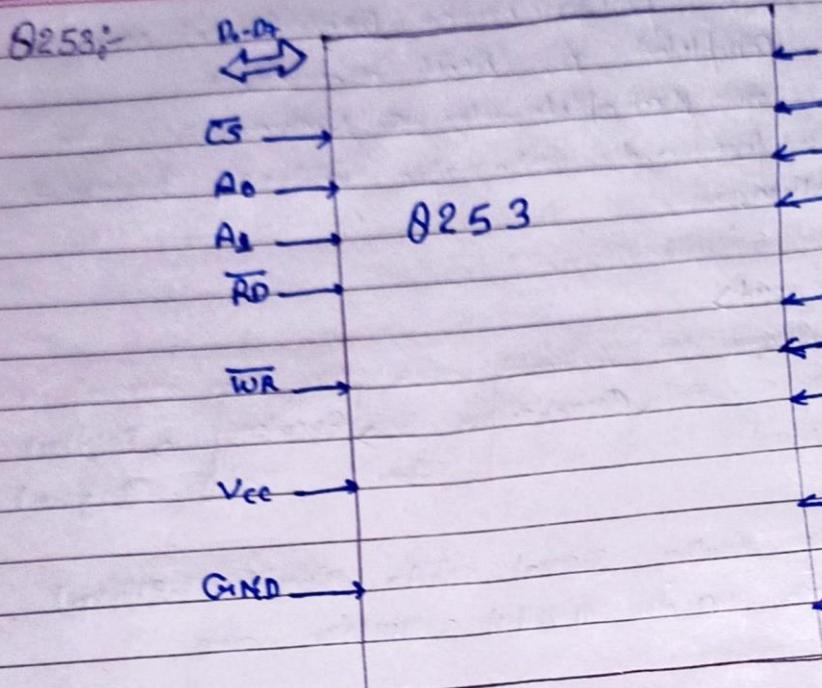
TxC → Transmitter clock

RxD → Receiver data

RxRDY → Receiver Ready

VCC → power supply

GND → ground



8253 Programmable Counter Timer: It is used for real time application for timing and counting function such as BCD.

8253 is comparatively 8086 and 8085.

pin description of 8253 are as follows

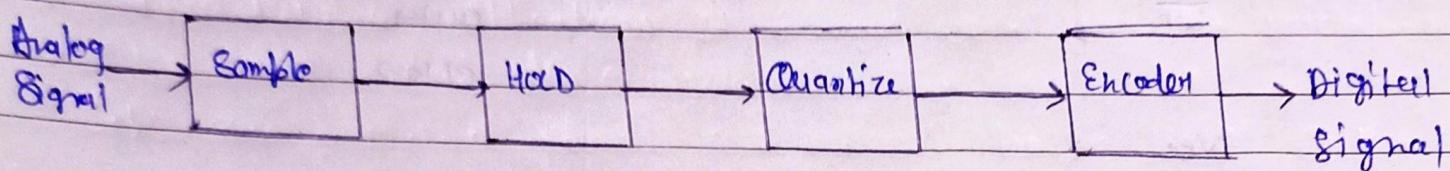
- D<sub>0</sub>-D<sub>7</sub> → Data Bus
- CS → Chip select
- A<sub>0</sub> → Address Bus
- A<sub>1</sub> → Address Bus
- RD, WR → Read and write
- Vcc → Power Supply
- GND → Ground

CLK<sub>012R</sub>: are clock for counter 012 gate 012 a  
gate terminal for Counter 012  
output 012

Date  
27/11/22

8237:- 8237 is a high performance DMA (Direct memory access) yet it is completed as 8086 and 8088 microprocessor yet it is incomplete by software. Yet has memory transfer and automatic address increment and element feature.

Analog to digital converts



Analog means converting Analog signal into Digital signal the step involve in analog conversion are

- (1) Sampling
- (2) HOLD
- (3) Quantize
- (4) Encoder

(1) Sampling  $\Rightarrow$  Sample Convert a particular frequency which we have to work.

(2) HOLD  $\Rightarrow$  Hold the function HOLD is to hold the value until the next sample are until the next sample obtain.

(3) Quantize  $\Rightarrow$  Quantize converting the Analog signals into discrete one which every technique is been used.

Making the a coming signal into smaller feature for error free communication.

Encoder:- It function encoder to convert connect the Analog signal into Digital signal.

Date  
28/11/22

### Digital to Analog:

Since the computer understand the digital language we need of digital to Analog Converter Array.

The digital to Analog Converter Computer the value on the weighted sum method in which position of each bottom been calculated and further added cumulatively.

So, that a human readable language obtain is been obtain.