SY

Microprocessor (CS-305)

Introduction :-

The microprocessor is a semiconductor device consisting of electronic logic circuits manufactured using either large scale integration (LSI) or very large scale integration (VLSI) technique.

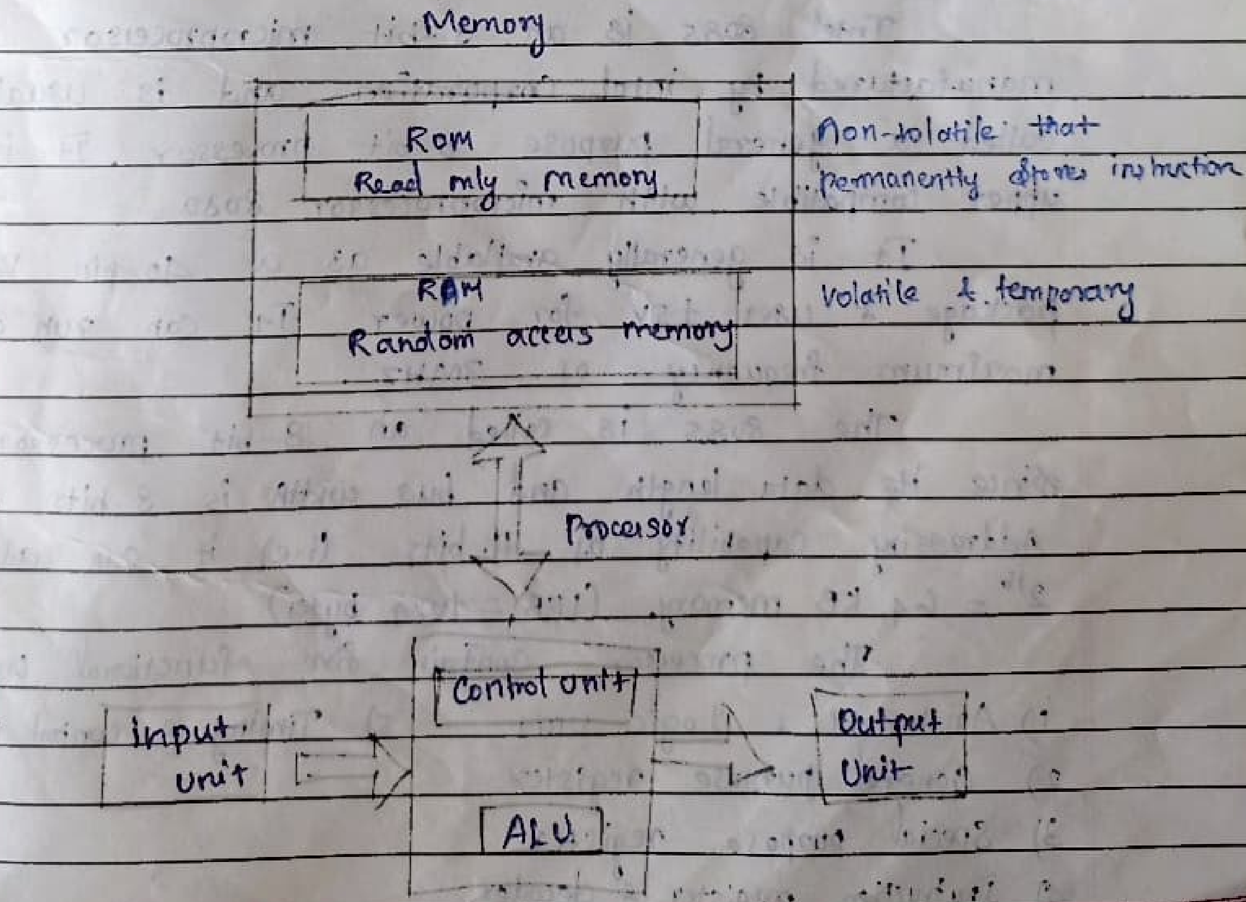
The microprocessor is an electronic chip that functions as the central processing unit (CPU) of a computer. In other words Microprocessor is a heart of any computer system. Today Microprocessor is present in all electronic devices such as washing machine, computer printers, ovens, mobile phones etc.

All microprocessors work using Von-Neumann architecture.

~~Go~~

~~Input Unit~~

A Microprocessor System



A microprocessor system consist of three functional block - CPU unit, input & output units, memory unit & CPU contains register, an arithmetic & Logic unit & a control unit. The control unit translates the instructions and executes the desired task.

A program is a list of instructions for the microprocessor to execute. Before the start of execution the complete program must be stored in the memory.

Let us assume that starting address of the program is 8800 H, while running the program, the microprocessor must be directed to go from 8800 H. once it has executed the instruction in 8800 H. it will go to the next address 8801 H. and so on until it reaches the end of the program.

*) Architecture of 8085 :-

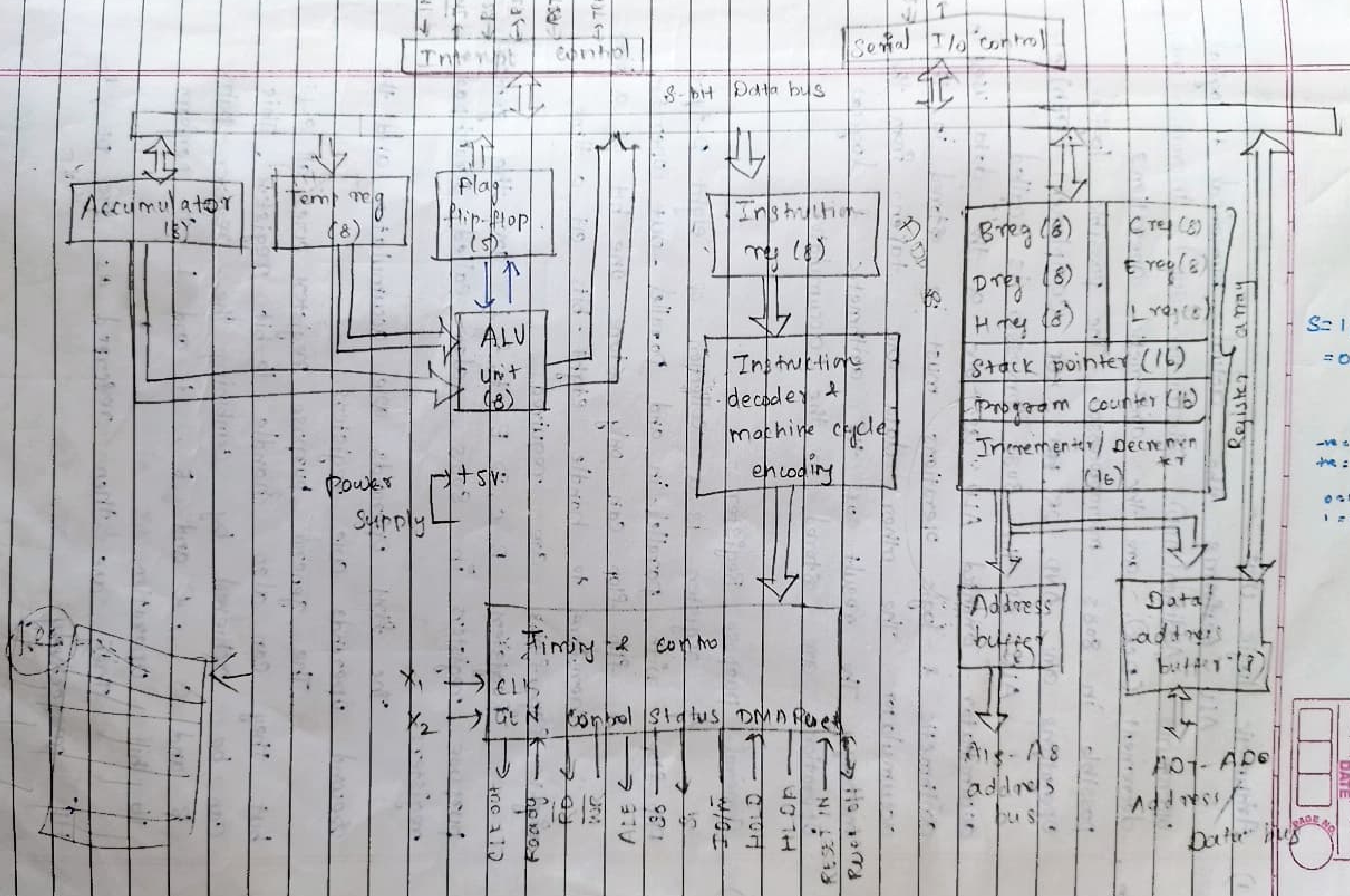
Intel 8085 is an 8-bit microprocessor manufactured by intel corporation and is usually called a general purpose 8-bit processor. It is upper compatible with microprocessor 8080.

It is generally available as a 40-pin IC package & uses +5V for power. It can run at maximum frequency of 3MHZ.

The 8085 is called an 8-bit processor since its data length and bus width is 8-bits. Addressing capability of 16 bits. (i.e) it can address 2^{16} = 64 KB memory (1KB = 1024 bytes)

The processor contain five functional units

1) Arithmetic & Logic unit
2) General purpose register
3) Special purpose register
4) Instruction register & decoder
5) Timing & control unit



S=1(-ve)
=0(+ve)

-ve:1
+ve:0

0:1
1:0

1) Arithmetic logic Unit :-

ALU performs the actual numerical & logical operations. Addition(ADD), Substraction (SUB), increment(INR), Comparison, decrement (DCR) are the arithmetic operations possible in 8085 microprocessor. The possible logical operations are AND, OR, ~~FOR~~ EXOR, Complement (CMA) etc.

ALU of the 8085 processor is called accumulator-oriented ALU as one of the data used in arithmetic & logic operations must be stored in the accumulator. The other data are taken from the memory. The result of the arithmetic & logical operations are stored in the accumulator.

2) General purpose Register :-

A register is a collection of eight D-type flip-flops with parallel-in and parallel-out operation.

A flip-flop can only store one bit at a time

Therefore to handle eight-bits at a time, eight flip-flops are required.

Registers B, C, D, E, H & L are the general purpose registers in 8085. also called scratchpad registers.

The first operands are accumulator and the second operands are registers

The general purpose registers are all 8-bit but they can also handle 16-bit register. This can be achieved by combining the register pairs B and C, D and E, H and L to perform 16-bit operation

They are then referred as BC, DE, & HL respectively

3) Special purpose Registers :-

Special purpose register dedicated to do special functions.

→ The Accumulator

→ Flag register

→ Program counter.

→ Stack pointer.

i) Accumulator :-

It is an 8-bit register. it is a part of the ALU and is the most important register.

It is used to store 8-bit data & to perform arithmetic & logical operations. The output of the ALU also stored in the accumulator.

Any previous data stored in the register will be overwritten as soon as new data is stored.

ii) Flag register :

This is a special 8-bit register. each bit would have a numerical value.

The flag is an 8-bit register used to indicate the status of a recent arithmetic or logical operation. There are five flag bits Zero(Z), carry(CY), Sign(S), Parity (P), Auxillary cary (AC)

S	Z	x	AC	x	P	x	CY
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

D_1, D_3, D_5 are unassigned, so they are marked with an x to show that they are not used & don't care

Flag register bit is said to be "set" when it value is 1, and cleared when the value is 0.

The most commonly used flags are zero, carry sign

3) Program counter :- This register is a memory pointer. It is a 16-bit register that always points to the address of the next instruction to be executed.

This register is used to sequecence the execution of the instructions.

when a byte (machine code) is being fetched, the program counter is incremented by one to point to the next memory locations.

4) Stack pointer :-

It is a 16-bit register. Stack is an array of memory locations organized in (FILO) & (LIFO) fashion.

It is accessed using a 16-bit point register called stack pointer, which holds the address of the memory location of the top of the stack.

Care must be taken by the programmer to ensure that the data stored in the stack is retriered properly, so that the data stored in the stack is not affected.

4) Instruction Register and decoder :-

This is the temporary storage for the current instruction of a program.

Latest instruction is sent to here from memory prior to execution

Decoder then takes instruction and decodes the instruction.

5) Timing & control unit :-

The timing and control unit gets commands from the instruction decoder. and issue signals on the data bus, address bus, control bus.

DATE

PAGE NO.

Data bus:-

The microprocessor performs its functions using wires (or) lines called buses.

For eg:- 8 bit microprocessor uses eight wires to carry data between microprocessor and the memory.

They are bidirectional : data flows in both direction between the 8085 memory & peripheral devices

The 8 lines enable the microprocessor to manipulate 8-bit ranging from 00 to FF (D0 - D7)

Address Bus :-

It is a group of 16-lines. It is unidirectional. bits flow only in one direction from 8085 to the peripheral devices.

The microprocessor uses the address bus to identifying a peripheral or memory locations

Each peripheral (or) memory location is identified by 16 bit address.

The address bus lines are generally identified as (A0-A15). The ~~first~~ higher-order 8 address lines are unidirectional (A8 - A15). The lower order address lines are bidirectional (A0 - A7).

Control Bus :-

The control bus carries control signals that are partialy unidirectional and partialy bidirectional.

eg:- Read/write control signal will indicate whether memory is being written into (or) read from.

ALE, $\overline{RD}$, $\overline{WR}$, $IO/\overline{M}$, S_1 & S_0 status signals

ALE :- Address latch enable — $AD_0 - AD_7$

$\overline{RD}$:- Read signal → The data are being read from the memory device

$\overline{WR}$:- write signal indicates the data are written in to the memory

Io/M :- Signal that distinguish between a memory operation & i/o operation

S_1 and S_0 :-

S_1	S_0	Status
0	0	Halt
0	1	write
1	0	Read
1	1	fetch

Interrupts :-

When the interupt signal is detected by the processor, it suspends the execution of the current program. and executes the program corresponds to the interrupt signal.

Five interrupt signals.

INTR, INTA, RST 5.5, RST 6.5, RST7.5 & Trap

INTR :-

It is a general purpose interupt request signal, It is an active high signal

INTA :-

It is ~~an~~ used to acknowledge ~~an~~ intercipt.

It is an active low signal

RST :- Reset.

when the signal goes low, the program counter is set to 0, the processor is reset.

TRAP :- It is a non-maskable interupt. it cannot be stopped on overhidden by any command.

*) Clasification of Instructions

*) ~~Instruction Set of~~ 8085 :-

Every microprocessor has its own instructions set The instruction set consists of both assembly language & corresponding machine code.

The purpose of instruction set is to based on the development of efficient programmers by the users.

The instruction set is based on the architecture of the processor

The syntax for 8085 instructions may contain one (or more of the following notations

R = 8- bit register (A, B, C, D, E , H & L)

Rs = Source register

Rd = Destination register (A, B, C, D, E , H & L)

Rp = Register pair (BC , DE, HL & SP)

P = Port address

8-bit = 8-bit data

16-bit = 16- bit data

( ) = Contents of

*) Instructions classified based on functionality

1) Data Transfer

2) Aithmetic operations

3) Logical operations

4) Branching operations

5) Machine control operations

1) Data tranfer (copy) operations.

This group of instruction copies data from a location called source register to another destination called destination register.

The content of source register is not modified, Although the term data transfer is used for copy operation.

Mov A, D

Transferring data between one register to another

Copies the content of register D to the A

MVI C, 66H → Storing data byte in a register (or ML

Loads register C with the data 66H

LDA 8800H

Transferring a data between a memory location & a register

Loads the content of ML (8800H) in the accumulator

IN PORT 1

Transferring data between an I/o & an accumulator.

2) Arithmetic Operation:-

Arithmetic operation include addition, Subsraction, increment & decrement.

For eg :-

ADD R	A ← A+R
ADD M	A ← A + (H)(L)
ADI Data	A ← A + data
eg:- ADI, 02H	A ← A+02H
SUB R	A ← A−R
SUB M	A ← A−(H)(L)
SUI Data	A ← A − Data
ADC R	A ← A + (R)+(c)
ADC M	A ← A+(H)(L)+(C)
SBB R	A ← A−(R)−(B)

Increment & Decrement :-

These operations can be used to increment (or) decrement the contents of any register (or) memory locations. Unlike ALU, the increment and decrements operations need not be based on the accumulator.

INR R R=01, R=02

R ← R+1

DCR. R=04, R=03

R ← R-1

Logical operations:-

Logical instructions are also accumulator oriented (i.e) they require one of the operands to be placed in the accumulator.

The other operand can be any register (or) memory locations. The result is stored in the accumulator.

The operation that uses two operands are logical AND, OR & EXOR. The operation that uses single operand (ie the accumulator) is logical complement (or) NOT operation.

The instruction set of 8085 supports rotation of the data stored in accumulator. The data can be rotated left (or) right, through the carry (or) without carry.

Branching operations:-

Branching instruction are of two types

jump instructions

Subroutine instructions

The jump instruction merely transfer the execution from one location in the program to another,

where as sub routine instruction in the main program transfer execution to a new location

The branching can take place conditionally (or) unconditionally.

Machine Control instruction :-

machine control instruction is used control the microprocess execution & functioning.

1) NOP means no operation. when this instruction is executed, nothing is done.

2) HLT used to Halt the execution of the program.

3) Interrupts are disable & enabled using DI & EI. signal

*) Instruction Classification based on length :-

Based on the length of the machine language code, the 8085 instructions can be classified in to following three types.

i) one-byte instruction ii) Two-byte instruction
iii) Three-byte instruction.

Assembly language instructions should be converted into machine code for storage and execution by the processor.

So the length of the machine language code instruction determines the length of the program.

This in turn determines the amount of memory required for the program.

One byte instruction :-

The instruction that require only one byte in machine code is called as one-byte instruction

If no number is given in instruction then it is 1byte instruction

In 1-byte instruction, the opcode and the operand of the instruction represented in one-byte.

eg:-

opcode	operand	Hex code
MOV	A,B	78
ADD	M	86
XRA	A	AF

Even though the instruction add M. adds the content of memory location to the accumulator its machine code requires only one byte

One byte = 8-bit

MOV A,B / ADD C

Two-byte instruction:-

Instruction that requires two bytes in machine code are called as two-byte instructions.

The first byte of the two byte instructions is the opcode, which specifies the operation to be performed.

The second byte is the 8-bit operand which is an address.

Opcode	Operand	MC/Hex code	Byte
SUI	37 H	3E	First byte
MVI	A, 7FH	7F	Second byte.
ADI	0FH	C6	First byte
		0F	second byte
IN	40H	DB	First byte
		40	Second byte

00
↓
FF

If two digit number is given in instruction then it is 2byte instruction.

If four digit no is given in instruction then it is three byte

Three byte instructions:-

Instructions that require three bytes in machine code are called three-byte instructions.

First byte = opcode
2nd & 3rd byte = operand → 16-bit.

0000
↓
FFFF

Opcode	operand	Mc/Hex code	Byte
JMP	9050H	C3	First byte
		50	Second byte
		90	third byte
LDA	8850H	3A	First byte
		50	Second byte
		88	Third byte

*) Addressing modes in Instructions:-

Every instruction in a program has to operate on data.

For eg:- the instruction MOV B, A and MVI A, 82H are used to copy data from a source to a destination. The source & destination is a register also called as operand → 8-bit number.

The various format for specifying operands are called addressing mode.

i) Immediate Addressing ii) Memory direct addressing
iii) Register direct addressing iv) Indirect addressing
v) Implied (or) implicit addressing.

Immediate Addressing:-

Eg:- MVI A, 9AH
ADI 05H.

Immediate transfer of data from source to destination

Memory Direct addressing:-

It move a byte (or) word between a memory location and register.

LDA 850FH

This instruction used to load the content of memory location 850FH in the accumulator.

Register Direct Addressing Mode:-

It transfers a copy of a byte (or) word from source Register to destination register.

Mov Rd, Rs.

Indirect Addressing:-

It transfer a data between a register & a memory location.

Mov R, M.

Implicit Addressing:-

No specific data (or) operand is specified in the instruction

CMA complements the content of accumulator.

*) Instruction Set of 8085:-

The 8085 microprocessor instruction set has 74 operation codes, & 246 instructions.

The 8080A has two additional instructions

- SIM (Set interrupt Mask)
- RIM (Read interrupt mask).

1) Data Transfer instruction:-

It is used to transfer data between two registers in the microprocessor (or) between a peripheral devices & the microprocessor.

MVI instruction is used for storing 8-bit data

LXI instruction is used for storing 16-bit data

Mnemonics	Tasks performed on execution	Addressing mode	Instruction length	Example.
MVI R, 8-bit	Moves the 8-bit data to the register	Immediate	Two bytes	MVI B, 3FH
LXI Rp, 16-bit	Loads the 16-dit data in the register pair	Immediate	Three byte	LXI . B , 5AF3
MOV Rd, Rs	copies the data from the source register to destination register	Register direct	One byte	Mov A, B
LDA 16-bit	Loads the accumulator with the data from the memory location indicated by 16-bit address	Register direct	Three bytes	LDA 905FH
LHLD 16-bit	Loads the H & L registers directly from two consecutive memory locations Indicated by the 16-bit address	Memory direct	Three bytes	LHLD 900AH

STA 16-bit	Stores the Content of accumulator in the memory location Indicated by the 16-bit address	Memory direct	Three bytes	SHLD 9050H
SHLD 16-bit	Stores the contents of HL register in to consecutive memory locations indicated by the 16-bit address.	Memory direct	Three bytes	SHLD 309FH
push Rp	Pushes the contents of the register pair onto a stack	Register direct	one byte	PUSH B.
pop Rp	Pops the top two memory location of the Stack on to a register pair	Register direct	one byte	Pop B.

2) Arithmetic Instructions :-

The arithmetic instructions supported by 8085 are addition, substraction and their variants.

Mnemonics	Tasks performed	Addressing mode	Length	Example
ADI 8-bit	Adds the 8-bit data to the content of accumulator	Immediate	2-bytes	ADI 30H

ACI 8-bit	Adds the 8-bit data & carry flag to the accumulator	Immediate	2-bytes	ACI 4FH
SUI 8-bit	Substract the 8-bit data from the content of accumulator	Immediate	2-byte	SUI 2AH
SBI 8-bit	Substract the 8-bit and the Borrow from the content of the accumulator	Immediate	2 byte	SBI 5CH
ADD R	Add the register with accumulator	Register direct	1 byte	ADD C
ADC R	Add the content of register & Carry with accumulator	Register direct	1 byte	ADC E
SUB R	Substract the content of Register from accumulator	Register direct	1 byte	SUB B
INR R	increment the register by 1	Register direct	1 byte	INR B
DCR R	Decrement the register by 1	Register direct	1 byte	DCR E
DAA	Convert the content of accumulator from binary to BCD	Implicit	1 byte	DAA

i) For arithmetic operations, One of the data must be stored in the accumulator & other given or addressed in the instruction

2) Add with carry instructions are used

3) Subtract with borrow instructions are used

4) Increment & decrement instructions can be operand not only on the accumulator, but also on othe registers & memory locations.

8) Logical instructions :-

The most important logical instructions supported by 8085 are AND, OR, EXOR & NOT.

And=1(1,1)

XOR- 00=0 11=0 01=1 10=1

OR-atleast one 1

ANI 8-bit	The 8-bit data is logically ANDed with the content of accumulator	Immediate	2-byte	ANI 0FH
XRI 8-bit XRA r M	The 8-bit data is logically XOR-ed with the content of accumulator	Immediate	2byte	XRI 01#
ORI 8-bit ORA r M	The 8-bit data is logicaly ORed with accumulator	Immediate	2 byte	ORI 80H
ANA R M ANI data	The content of register logically ANDed with the accumulator	Register direct	1 byte	ANA C
XRA R	The content of register are logically ~~A~~XORed with the content of accumulator	Register direct	1byte	XRA D

4) Branching instructions :-

Branching instruction used to transfer the program execution to a different address

1) Jump instruction - It merely transfer the execution from one location in the program to another

2) Subroutine - Transfer execution to a new location. A group of instructions written separately from the main program to perform function that occur repi-

JMP 16-bit	jump unconditionally	three bytes	JMP 9500
JC 16-bit	Jump if carry is set	3 byte	JC 9500
JNC 16-bit	Jump on no carry	3 bytes	JNC 9500
JP 16-bit	Jump on positive	3 bytes	JP 9500
JM 16-bit	Jump on minus	3 bytes	JM 9500
JZ 16-bit	Jump on Zero	"	JZ 9500
JNZ 16-bit	Jump on no zero	"	JNZ 9500
CALL 16-bit	call unconditionally	"	CALL 9500
CC 16-bit	Call on carry	"	CC 9500
CNC 16bit	call on no carry	"	CNC 9500
CP 16-bit	call on positive	"	CP 9500
RET	Return unconditionally	"	RET

Flag → CY, Zero, parity, sign

5) Machine control instructions :-

NOP	No operation	Implicit	1 byte
HLT	Halts the processor	"	"
DI	Disables interrupt	"	"
EI	Enables interrupt	"	"
RIM	Reads interrupt mask	"	"
SIM	Sets interrupt mask	"	"

*) Introduction to 8051 Microcontrollers:- (1981)

Microprocessor consists only a central processing unit where microcontroller consist of CPU, Memory, I/o in all integrated into one chip.

The 8051 is an 8-bit microcontroller with 8-bit data bus & 16-bit address bus.

The word leaders of microprocessor & microcontrollers are Intel, Motorola, IBM etc.

tly in the main program

In 1981 Intel corporation introduced an 8-bit microcontroller called 8051.

Features of 8051

1) ROM - 4k bytes
2) RAM - 128 bytes
3) Timer - 2
4) I/o pins - 32
5) Serial ports - 2
6) 8-bit data bus
7) 16-bit address bus

It is an 8-bit processor means it can process 8 bit of data at a time. It has total four I/o ports, each 8-bit wide.

PIN diagram of 8051:-

It consist of 8-bit data bus and 16 bit address bus

Component present in 8051 Architecture

1) Bus control - data & address Bus

2) Four general purpose parallel Input /output ports

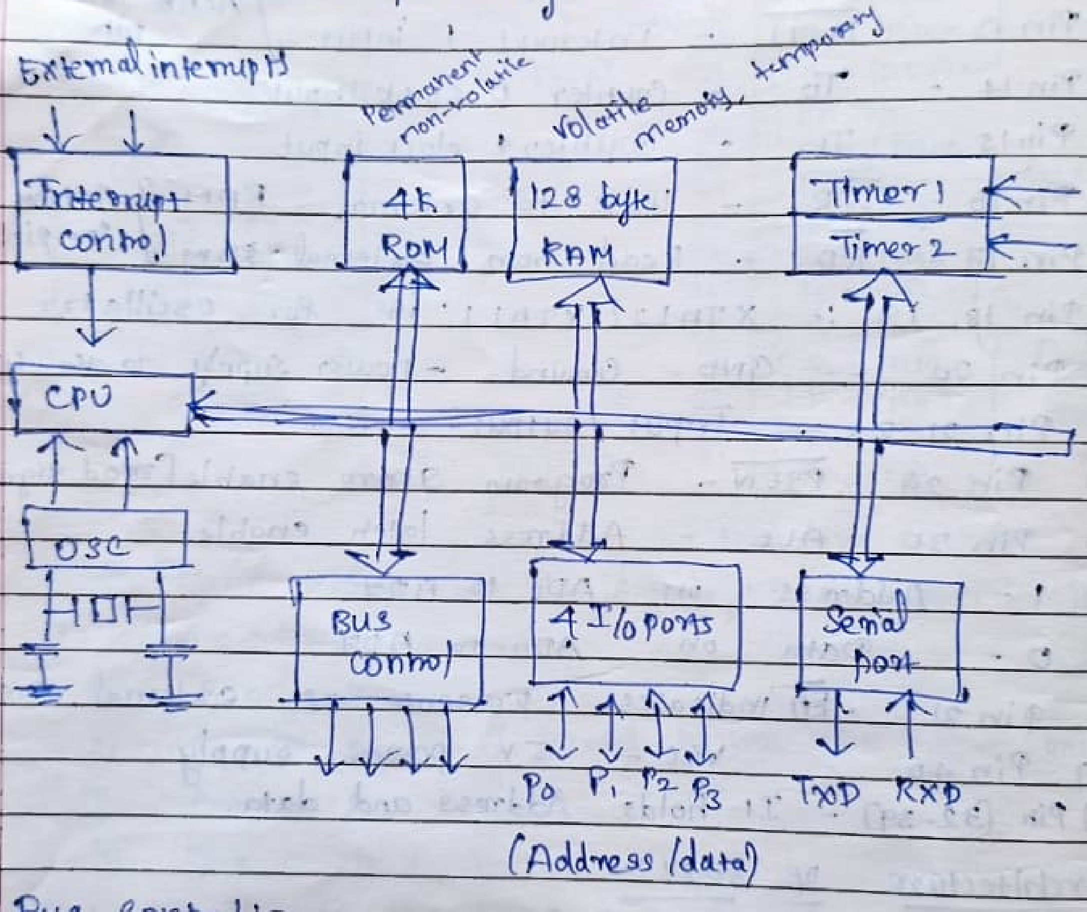
3) Timer & counter

4) Internal RAM & ROM

5) Serial port

6) Interrupt control Logic

7) CPU [central processing unit]



1) Bus control:-

It is a group of wires which uses as a communication channel (or) acts as medinsum of data transfer.

The different bus configuration includes 8, 16 (or) more cables.

Two types of bus used in 8051 microcontroller

1) Address Bus:-

It consist of 16 bit address bus. It is

mainly used for transferring the data from central processing unit to memory

Data Bus:-

8051 micro controller consist of 8-bit data bus It is also used to transfer data between peripheral devices

2) Four general purpose I/o ports:-

Normally microcontroller is used in embedded systems to control operation of machines in the microcontroller.

Therefore to connect it to other machines, devices (or) peripherals we require I/o interfacing ports.

For this purpose microcontroller 8051 has four input output ports to connect it to other peripherals

P_0, P_1, P_2, P_3 are foure different ports.

3) Timer / counter:-

Internal operation can be synchronized using clock circuit.

Two ports are used $XTAL_1$, $XTAL_2$

8051 microcontroller has two 16 bit timers and counters.

The timers are used to measure the intervals to determine the pulse width of pulses

4) Internal RAM / ROM :-

8051 microcontroller has 4K of code memory (or) program memory (i.e) 4KB ROM.

And 128 bytes of data memory of RAM

The data memory of the 8051 is used to store data temporarily for operation is known as RAM memory.

The memory which is used to store the program of the microcontroller is known as code memory (or) ROM memory

5) Serial Port :-

- SBUF - (Serial port data buffer) holds the data
- SCON - (serial control) - Manages the data communication
- PCON - (power control) Manages data transfer rate.

The ports connected to serial port are RXD, TXD.

6) Interrupt control:-

The feature of interrupt is very useful as it helps in case of emergency operation. An interrupt gives us a mechanism to put on hold the ongoing operations, execute a subroutine and then again resume to another type of operations.

The 8051 can be configured in such a way that it temporarily terminates (or) pause the main program at the occurrence of interrupts.

Two ways of giving interrupts to micro controller.

1) By sending Software instruction
2) By sending Hardware signal

5 Sources of interrupts in 8051

→ INT0
→ INT1
→ TF0, TF1, RI/TI

7) CPU - Central processing unit
It monitors (or) Controls all operations that are performed in microcontroller. Its function is to read program written in ROM memory and do the executed task.

Questions:-

1) write short note on classifications of instructions based on length
2) write a short note on following arithmetic instructions related to 8085
i) ADD ii) SUB iii) INR
3) write short note on Data memory of 8051 micro controller
4) write a program of addition of two 8 bit & 16 bit numbers
5) Explain logical instruction for 8085
6) Explain architecture of 8085
7) What are the instruction set of 8085

Completed on
25/10/2021