

Microprocessors & Programming

PAGE NO. 1

DATE _____

Chapter - 1) Basics of microprocessors	- 8
2) 16-bit microprocessors	- 24
3) Instruction set of 8086 microprocessors	- 20
4) The art of Assembly Language Programming	- 8
5) 8086 Assembly Language Programming	- 2
6) Procedure and macro in Assembly Language Programming.	- 16

Chapter - 1) Basics of Microprocessors

Evolution of MicroProcessors and

its types:

The developments leading to the intel architecture (IA) can be traced back through the 8085 and 8080 8-bit microprocessors to 4004 4-bit microprocessors. 4-bit microprocessor 4004 is the first microprocessor designed by Intel in 1969.

First actual processor in IA family is 8086 quickly followed by more cost effective version for smaller systems 8088.

Object code program created for this processors starting in 1978 will still execute on the latest members of the IA family. The 8086 has 16-bit registers and a 16-bit external databus with 20bit addressing giving a one megabyte address space. The 8088 is identical except for a smaller external databus of 8 bits.

These processors introduced IA segmentations but only in real mode. 16 bit registers can act as pointers to address into segments of upto 64KB in size. The four segment registers hold the effective 20 bit base address of the currently active segment.

Intel 80286 processor introduced the protected mode. This new mode uses segment register contents as selector or pointers into descriptor tables.

Intel 386 processor introduced 32-bit registers into the architecture.

Intel 486 processor added more parallel execution capability.

Intel Pentium: This processor added a second execution pipeline to achieve superscalar performance. Two pipelines are known as u and v, together can execute two instructions per clock.

The intel pentium pro processors introduced dynamic execution. It has three ways super scalar architecture which means it can execute 3 instructions per clock.

Salient features of 8085 microprocessor:

- 1) 8-bit microprocessor i.e. 80854P can read or write or perform arithmetic and logical operations on 8-bit data at a time.
- 2) Single chip N-mos device implemented with 6200 transistors
- 3) Requires a single +5Vdc supply
- 4) Provides on-chip clock generator so 80854P does not require external clock generator, but require external tuned circuit like R-C, L-C or crystal.
- 5) Requires two phases, 50% duty cycle TTL-clock.
- 6) On-chip bus controller.
- 7) Provides 74 instructions with 5 addressing modes.
- 8) Maximum clock frequency is 8 MHz and minimum clock frequency is 500 kHz
- 9) Provides 16 address lines so 80854P can access $2^{16} = 64 of memory$
- 10) Provides 5 level hardware interrupts and 8 software interrupts
- 11) Can generate 8-bit I/O address, so $2^8 = 256$ input and 256 output ports can be accessed.
- 12) Provides 2 serial I/O lines, so serial peripheral can be interfaced directly with 80854P

PIN diagram and description

X1 [1]	Vcc
X2 [2]	HOLD
RESET OUT [3]	MLDA
SOD [4]	CLK(GOUT)
SID [5]	RESET IN
TRAP [6]	READY
RST 7-5 [7]	IO/M
RST 6-5 [8]	S1
RST 5-5 [9]	RD
INTA [10]	WR
INTA [11]	ALE
AD0 [12]	SD
AD1 [13]	A15
AD2 [14]	A14
AD3 [15]	A13
AD4 [16]	A12
AD5 [17]	A11
AD6 [18]	A10
AD7 [19]	A9
GND [20]	A8

a) AD0 to AD7 : Multiplexed Address / Databus: These are 8-bit bidirectional tri-state pins to carry address as well as data in time shared mode. The lower order 8-bit address signals are multiplexed with data bus.

In earlier clock cycle of machine cycle, these lines carry lower order 8-bit address of 16-bit address and in latter clock cycle, these lines carry 8-bit data.

b) A8 to A15 : Address bus - These are 8-bit output, tri-state signals used to carry higher order address signals of 16-bit address. These are non-multiplexed unidirectional tri-state address lines.

c) ALE : Address Latch Enable - This is an output signal used to inform that the content on AD0 to AD7 lines is lower order 8-bit address of 16-bit address. When ALE goes high.

When ALE is low, it indicates the content on AD0 to AD7 lines is 8-bit data. ALE signal is used to separate address signals from the data signals i.e. AD0 to AD7 and D0 to D7 from AD0 - AD7.

d) IO/M : Input Output / Memory - This is an output status signal used to give the status of a operation performed with memory or I/O by microprocessor. When IO/M is low, the microprocessor is performing operation related to memory and when IO/M is high, the microprocessor is performing operation related to I/O device.

e) RD : Read - This is an active low output control signal used to read data from memory or I/O device generated by the microprocessor.

f) WR : Write - This is an active low output control signal used to write data to memory or I/O device generated by the microprocessor.

g) S1 & S0 (status signals) - These are output status signals used to give the status of operations performed by the microprocessor. These signals specify four different conditions as given in table.

8085 Machine Cycle Status

IO/M	S1	S0	Status	Control signals used
0	1	1	Op-code fetch	RD = Low
0	0	1	Memory read	RD = Low
0	0	0	Memory write	WR = Low
1	1	0	I/O Read	RD = Low
1	0	1	I/O Write	WR = Low
1	1	1	Interrupt service	INTA = Low
Z	0	0	Hold	RD and WR = High impedance
Z	X	X	Hold acknowledge	RD and WR = High impedance
Z	X	X	Reset	INTA = High

h) READY - This is an active high input control signal and used by the microprocessor to check whether a peripheral is ready or not for data transfers. If READY pin is high, then the microprocessor completes the operation and proceeds for next operation. When READY pin is low, the microprocessor will wait until it goes high. This is signal used to synchronize slower peripheral with faster processor.

i) TRAP - This is an active high edge and level triggered, non-maskable highest priority hardware interrupt. When TRAP occurs, then microprocessor starts execution from @024H automatically.

j) INT_R and INT_A: Interrupt request and interrupt acknowledge:

INT_R is an active high, level triggered general purpose hardware interrupt. When INT_R occurs, the microprocessor generates interrupt acknowledge signal INT_A. Using INT_A, a call instruction along with vector address can be inserted to jump to the interrupt service routine (ISR).

k) RST 7.5, RST 6.5: Restart interrupts —

These are active high maskable hardware interrupts. RST 7.5 is edge triggered interrupt, RST 6.5 and RST 5.5 are level triggered interrupts. When RST 7.5, RST 6.5 and RST 5.5 occurs, then the microprocessor transfers program control to vector address 003CH, 0034H and 002CH respectively.

l) HOLD and HLDA:

HOLD is an active high input signal used by the other master controller to request microprocessor for gaining the control of address, data and control buses. When microprocessor receives HOLD request signal, then microprocessor completes current machine cycle i.e. operation and release bus control for other master in the system. Microprocessor also generates HLDA signal to acknowledge requesting device after receiving hold signal.

m) RESET IN : RESET INPUT:

This is an active low input reset signal used to reset microprocessor. When this signal is received by the microprocessor, microprocessor clears Program counter i.e. 0000 and makes address, data and control lines tri-stated and start execution from memory location 0000H onwards. After reset, the status of internal registers and flag registers are unpredictable.

n) RESET OUT: Reset output :

This is active high output signal generated by the microprocessor after receiving RESET signal and used as a system to RESET other devices in the system.

o) S_IO and S_OD : Serial Input data & Serial output data:

S_IO and S_OD are an active high input and output serial port pins respectively. S_IO pin is used to accept 8-bit data under the software control.

When RIM instruction is executed, the S_IO pin data is loaded at D₇ position of the accumulator. When SIM instruction is executed, the S_OD pin is set or reset depending on the status of D₇ and D₆ bits of the accumulator.

p) X₁ and X₂ :

These are the clock input pins, connected to the crystal, LC or RC circuit. The crystal frequency is divided by 2 and used as operating frequency.

q) CLK OUT :

This is an output clock signal and used as a system clock. The internal operating frequency is available on this pin and used to operate other device in the system.

Difference between maskable and non-maskable interrupt:

In maskable interrupt, the interrupt signal is ignored if the interrupt enable bit in SR is 0.

In non-maskable interrupt, the interrupt signal is always accepted.

Non-maskable interrupt is generated by timer, keyboard etc.

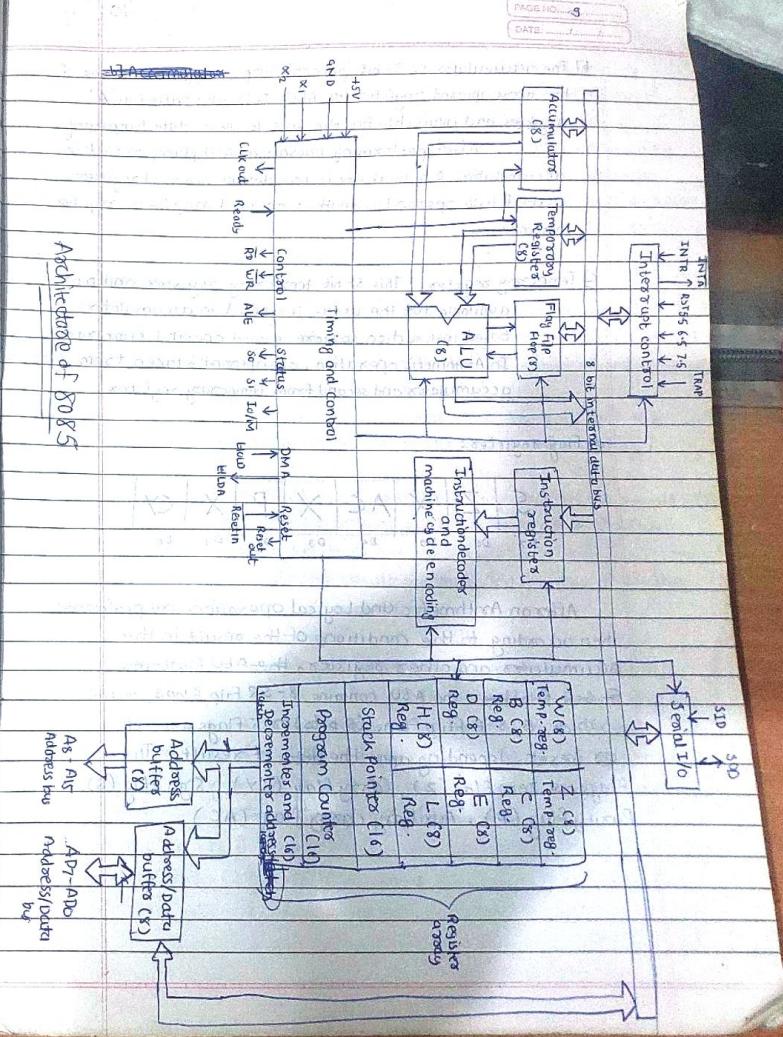
- PAGE NO. 8
- NMI**
(Non-maskable interrupt)
- MI** (Maskable interrupt)
- 1) Cannot be avoided by microprocessor.
 - 2) Always edge as well as level triggered.
 - 3) Response time is fast.
 - 4) No instruction is available for masking.
- ex:- TRAP of 8085
- ex:- INTR, RST 7.5, 6.5, 5.5 of 8085

Architecture of 8085 [block diagram]

Architecture of 8085 is as shown in figure. This architecture is divided into different groups as given below:

- A] Arithmetic & Logical Unit
- B] Register group
- C] Interrupt control group
- D] Serial I/O group
- E] Instruction Register, decode, timing and control group

A] **Arithmetic & Logical unit:** This group consist of ALU, Accumulator, Temporary registers and flag register. The 8-bit ALU performs arithmetic operations such as addition, subtraction and logical operations such as ANDing, ORing and XORing etc. on 8-bit data at a time. It takes operand from the 8-bit temporary register and 8-bit accumulator. After performing operation, the result is placed in accumulator only.



b) The accumulator is 8-bit general purpose register and one of the two operand registers for ALU. It is also called as 'A' register and available for the user to store data temporarily. After performing operation, ALU places result in accumulator. Accumulator is one default operand register out of two operands, another operand may be in register or memory.

c) Temporary registers: This 8-bit temporary register and not available for the user. It is used in accumulator based instruction to store second operand temporarily. In Arithmetic operation, one operand is taken from accumulator and second from temporary register.

d) Flag registers:

S	Z	X	AC	X	P	X	CY
D7	D6	D5	D4	D3	D2	D1	D0

After Arithmetic and Logical operations are performed then according to the conditions of the result in the accumulator and other registers, the ALU contains 5 S-R flip-flops. The ALU contains 8-SR flip-flops, out of which 5 S-R flip-flops (5 bits) or 5 flags are set or reset depending upon the status of result. These flags are zero flag (Z), carry flag (CY), sign flag (S), parity flag (P), auxiliary carry flag (AC).

Flag register is an 8-bit register and not available for the user. It provides status of the result of different operations performed by ALU. The flag register is connected to ALU. This register will get affected only when microprocessor performs Arithmetic or Logical Operations. Output of the flip flop is not available outside the microprocessor. The state of the flip flop can only be transferred to the stack and then can be loaded in general purpose registers in the stack.

* CY: Carry Flag :- If an operation performed in ALU generates a carry from D7th bit to next stage, the CY flag is set to 1 which acts as 9th bit of result of addition and borrow flag for the subtraction. If there is no carry, or borrow out of the most significant bit D7 of the result, then CY flag is reset.

* P: Parity Flag :- This bit is used to indicate parity of the result. If the result contains even number of 1's, then flag is set to 1. If the result contains odd number of 1's, then this flag is reset to zero.

* AC: Auxiliary carry flag :- If an operation performed in the ALU generates carry from D0 to D4 position. Then AC flag is set to 1, else reset to 0. That is carry generated from lower nibble to higher nibble. This is not a general purpose flag. It is used only by microprocessors perform binary to BCD conversion.

* Z: Zero flag : If the result of operation performed by ALU is zero, then Z flag is set to 1 else reset to 0. When this flag is set, then result i.e. content of A register is 0.

* S: Sign flag : In sign magnitude number, the sign of the number is indicated by most significant bit D₇ of 8-bit number. The sign flag is a copy of D₇ bit to indicate that result of operation performed by the ALU is negative so when result is negative, this flag will be set to 1 else reset to 0 for positive result.

Registers group : This group consists of three types of registers.

A] Temporary registers

B] General Purpose registers

C] Special purpose registers

A] Temporary registers : The 8085 microprocessor provides two 8-bit temporary registers i.e. W and Z and not available for the users. They are used by the microprocessor itself for internal operation such as to store operand, immediate operand or address of the memory.

B] General purpose registers : The 8085 provides 6 general purpose registers of 8 bit each i.e. B, C, D, E, H, L. These registers are available for the user and can be used to store 8-bit data or can be used to form registers pair such as BC, DE, HL to store 16-bit data.

C] Special purpose registers : The 8085 has 3 special purpose registers i.e. program counter (PC), Stack Pointer (SP), and Increment/Decrement Latch.

① Program counter : It is a 16-bit register used for sequencing of the execution of program. The register always points to the memory location from which the next instruction is to be fetched and executed. When the microprocessor performs the operation of fetching instruction, the PC content is automatically incremented by 1 to point next instruction. So program counter keeps track for execution of the program.

On reset, the program counter is loaded with address 0000H and starts execution of the program from the first memory address 0000H onwards.

② Stack pointer : It is a 16-bit register used to address stack location. Stack is user defined reserved portion of memory where data can be stored temporarily. SP always points current top of stack.

③ Increment & Decrement Latch : It is a 16-bit register to increment or decrement the contents of program counter and stack pointer registers.

Interrupt Control Group : This block accepts different interrupt request input such as TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR. When interrupt request is present, it informs to control logic to take action on each signal. In response to TRAP, RST 7.5, RST 6.5 and RST 5.5, program control is transferred to corresponding vector address. But for INTR, it generates INTA to acknowledge the same and expects that the external device should send opcode of CALL instruction along with vector address, so that the program control can be transferred and action can be taken for the interrupt request.

Maskable interrupts are:

- i) INTR
- ii) RST 7.5
- iii) RST 6.5
- iv) RST 5.5

Non-maskable interrupts are:

- v) TRAP

Serial I/O control group: The serial data transfer is implemented in 8085 by using SID and SOD pins. The data can be accepted from SID pin or the data can be transferred on SOD pin under software control by serial I/O control group block. To perform serial data transfer, two instructions are provided in instruction set of 8085 i.e. RIM and SIM.

Instruction register, decoder and control group:

Instruction register (IR): It is 8-bit register used to store current instruction of a program temporarily i.e. latest instruction is sent in IR from memory prior to execution and not available for the user. When microprocessor read opcode from the memory, it is loaded in the instruction register and then transferred to decoders for decoding.

Instruction decoder (ID) & Machine cycle decoding:

It accepts an opcode of the instruction from the instruction register, decode it and give information to the control logic. The information include what operation is to be performed, who is going to perform, how many operand bytes the instruction has.

Timing and Control Unit: It accepts the information from the instruction decoder and generates different control signals. This block accepts clock input to perform sequencing and synchronising operation by generating different status and control signals.

8085 system bus: Computer system uses a number of buses i.e. group of wires, which transmits binary numbers, 1 bit per wire. Any microprocessor transfers data between memory and other I/O devices using three buses i.e. address bus, data bus and control bus.

Address bus: 1 wire for each bit i.e. 16 bits = 16 wires.

The address bus consists of 16 wires, therefore its width is 16 bits. A 16-bit binary number allows 2^{16} different numbers or 64K different numbers i.e.

0000000000000000 to 1111111111111111 \rightarrow 0000H to 1111H

To communicate with memory, the microprocessor sends address to the address bus. Address bus is unidirectional i.e. numbers only sent from microprocessor to memory (Address signals), not others.

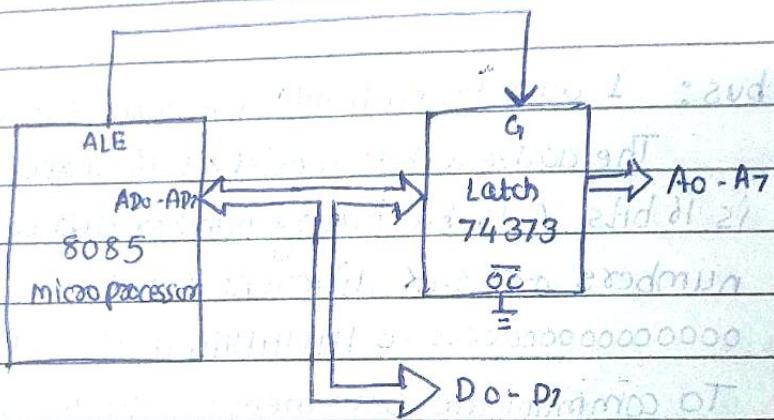
Data bus: Data bus carries data in the binary form between CPU and other external units such as memory or peripherals.

Data bus typically consists of 8 wires in 8085 and it is bidirectional. So $2^8 = 256$ possible combinations of binary digit are possible i.e.

The range of digits is 00H to FFH. Data bus is used to transmit binary data in the form of bits i.e. opcode and range of instruction, operand and results of arithmetic and logical operations between memory and the microprocessor.

Control bus: Control bus has various lines which have specific functions for co-ordinating and controlling microprocessor operations i.e. read, write, line control bus carries control signals partly unidirectional, partly bidirectional and control signals are things like READ or WRITE. From control signals memory READ, memory WRITE OR I/O READ, I/O WRITE is known to the memory as well as I/O device.

De-Multiplexing of A₀ to A₇



In 8085 microprocessors, the higher order address lines i.e. A₈ to A₁₅ are directly available but the lower order address lines are multiplexed with the data bus in time sharing. Hence the De-Multiplexing of address / data bus is required i.e. separation of address & data bus. De-Multiplexing of data/address bus can be implemented by using tri-state octal latch 74LS373 and this latch can be controlled by using ALE signal of 8085 as shown in figure.

When ALE goes high, the address signals will be latched in the octal latch 74LS373 and output of the latch will be provided on A₀ to A₇. When ALE goes low, the latch will be disabled and A_{0o} to A_{D7} can be used as data bus D₀ - D₇.