Unit-2

Basic Architecture

Date. _____
Page No. _____

* Introduction to 8085 microprocessor with its Architecture (or Block diagram).

The 8085 microprocessor is 8-bit general purpose microprocessor capable of addressing 16K of memory. The device has 40 pins which required +5V single phase power supply and can operate with 3MHz single phase clock.

The 8085 is an enhanced/improved version of it's previous model 8080. Its instruction is upward compatible with that of 8080. It was most popular microprocessor for commercial purpose of 8-bit.

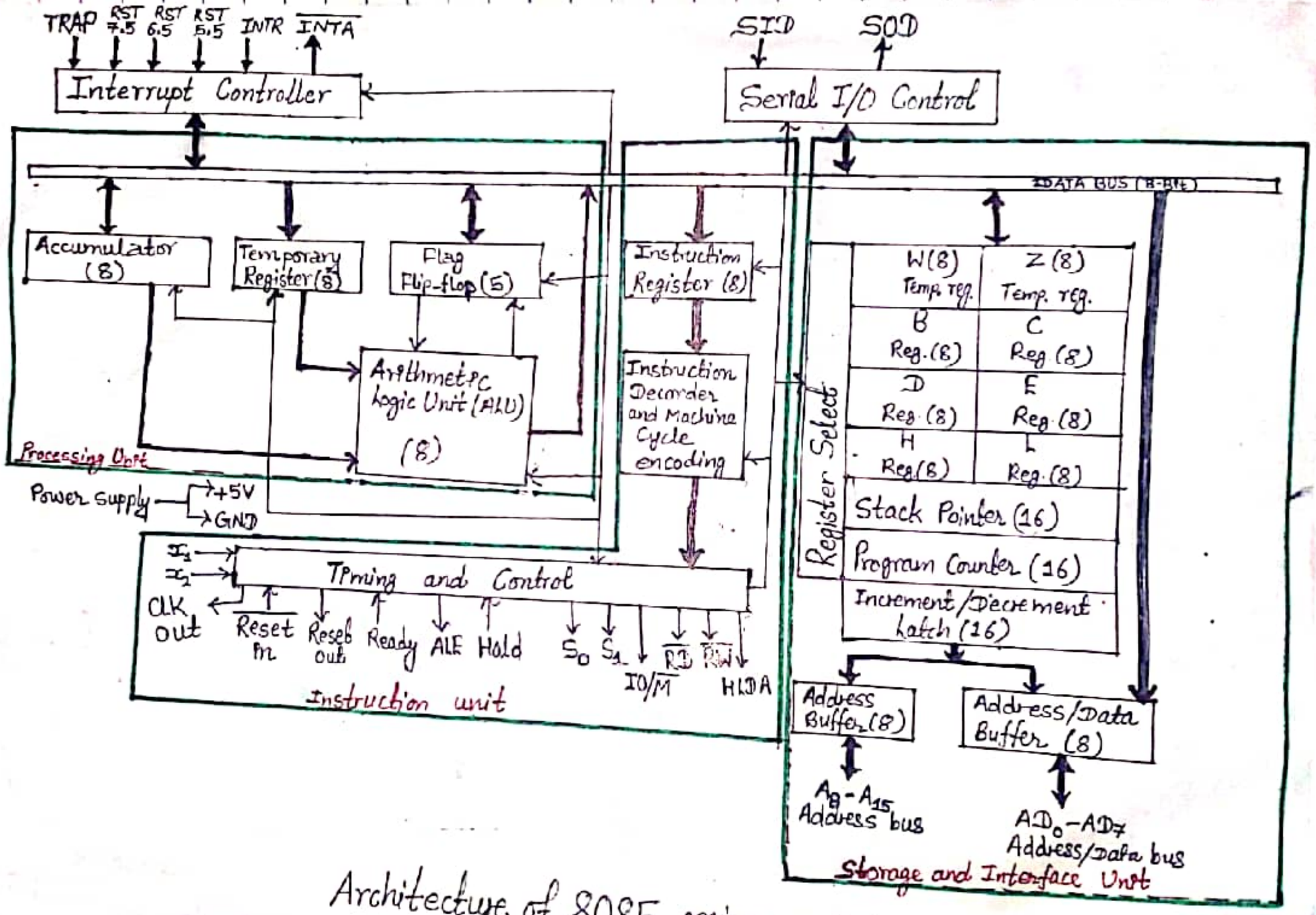
This microprocessor contain 3 buses Address, Data and Control. Address bus is of 16 Bit i.e. 2^{16} = 64kb of data can be recognized by at a single cycle in single operation by address bus. Data bus carries data required from recognized bus and control bus will help for the read/write operation. The size of data bus is 8-bit, therefore 8085 microprocessor is also called 8-bit microprocessor. All the signals in 8085 microprocessor can be classified into six groups.

PIN configuration of 8085 µP

i) Address bus.
ii) Data bus
iii) Control and status signal.
iv) Power supply and frequency signal.
v) Externally initiated signals.
vi) Serial I/O ports.

The block diagram of 8085 is as follows:-

Scanned with CamScanner



Architecture of 8085 microprocessor.

Scanned with CamScanner

Date.
Page No.

Read this after pin diagram

⊛ Address, Data & Control Buses of 8085 microprocessor:

Address bus & data bus

The microprocessor has 16 signal lines (pins) that are used as address bus which one are grouped into two segment A8–A15 and AD_0–AD_7. The eight signal lines A8–A15 are unidirectional and used for most significant bits, called the higher order address and the signals AD_0–AD_7 are bidirectional and used for dual purpose, ~~While~~ address as well as data. While executing an instruction these bits are used as lower address bus and during lower part of cycle these lines are used as data bus. This is known as multiplexing of bus.

Control and status signals

The group of signals includes two control signals $\overline{RD}$ and $\overline{WR}$, three status signals IO/$\overline{M}$, S0 & S1 to identify the nature of operation and one special signal ALE (Address Latch Enable) to indicate the begining of operation. ALE is positive going pulse generated everytime the microprocessor begins an operation, which indicate that the bits AD_0 – AD_7 are address bits.

$\overline{RD}$ $\overline{WR}$ IO/$\overline{M}$ & S_0 S_1

$\overline{RD}$ control signal (active low) indicates that the selected IO for memory device is to be read and data are available on data bus. $\overline{WR}$ control signal (active low) indicates that data on the bus are to be written into memory or I/O Location. IO/$\overline{M}$ status signal is used to differentiate between IO and memory location. When it is high it indicates IO operation & when it is

These are also generations of control signals

Date.
Page No.

low it indicates memory operation. S0 and S1 status signal are similar to $IO/\overline{M}$ which can identify various operations but they are rarely used in small program.

⊛ Internal data operations and Registers:

The ALU performs the computing functions which are mathematical and logical and which includes accumulator, temporary register and flag register which are used to hold data during arithmetic logic operation. The result of operation is shared in accumulator and flags are used to indicate the status of accumulator and other register after the completion of operation. These flip-flops are set or reset according to the data condition of the result in accumulator. Five flags are used in 8085 microprocessor which are described below:-

i) Sign flag (S) → It indicates whether the result of operation is positive or negative. If the result is positive the flag will reset (i.e. S=0) and if the result is negative the flag will set (i.e. S=1).

ii) Zero flag (Z) → It indicates whether the result of operation is zero or not. If the result of current operation is zero the flag will be set (i.e. Z=1) otherwise the flag will be reset (i.e. Z=0).

Scanned with CamScanner

Date.
Page No.

iii) Auxiliary carry (AC) → In operation when the carry is generated by bit D_3 and passes onto bit D_4 the AC flag will be set (i.e. AC=1) otherwise AC flag will be reset.

iv) Parity flag (P) → This flag indicates whether the current result is even parity or odd parity. If there is even parity parity flag (P) will be set and reset for odd parity.

v) Carry flag (V) → This flag indicates whether carry is generated or borrow is generated during addition or subtraction operation. If carry or borrow is generated the flag will be set otherwise reset.

Register:

8085 microprocessor has 8 bit registers B, C, D, E, H, L, A (Accumulator) and F (flags) and 16-bit registers PC (Program Counter), SP (Stack Pointer), IR (Instruction Register). These registers are classified into two groups general purpose register and specific purpose register.

B, C, D, E, H, L are 8-bit general purpose registers and can be used either singly or as paired 16 bit registers as BC, DE and HL. The content of these register is called higher order bytes and lower order bytes. The register pair HL function as data pointer. Accumulator and Flag are 8-bit special purpose registers whereas PC, SP and IR are 16-bit special purpose registers.

Scanned with CamScanner

$\mu p \rightarrow$ microprocessor

Date.
Page No.

(*) Externally initiated operations of 8085 μp:

8085 microprocessor support some externally initiated operations, which are also known as peripheral operations. Different external input/output devices or signals can initiate these type of operations. Following are some externally initiated operations.

i) RESET → This reset key is used to clear the program counter and update with 0000H memory location. When this RESET pin is actived by any external key, then all the internal operations are suspended for that time. After that the execution of the program can begin at the zero memory address.

ii) Interrupt → 8085 microprocessor chip have some pins for interrupt like ~~trap~~ TRAP, RST 5.5, RST 6.5 and RST 7.5. The microprocessor can be interrupted from the normal instructions and asked to perform some other emergency operations, which are also known as Service routine. The microprocessor resumes its operation after the completion of Service routine.

iii) READY → The 8085 microprocessor has a pin called READY. If the signal at this pin is in low state then the microprocessor enters into the wait state. The Input/Output devices are connected to microprocessor are of different speed, which is need to be synchronized with the speed of microprocessor.

Scanned with CamScanner

Date.
Page No.

iv) HOLD → When HOLD pin is actived by an external signal, the microprocessor relinquishes control buses and allows the external peripheral to use them. For example, the HOLD signal is used in Direct memory access (DMA) data transfer. In this DMA, the external Input / Output device are directly communicating with the memory without interfering the processor every time.

(*) Addressing Modes : (V. Imp)

The 8085 microprocessor have 5 types of addressing modes which are described below with examples.

i) 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:-

a) MVI B 45 (i.e. move the data 45H immediately to register B).

b) LXI H 3050 (i.e. load the H-L pair with the operand 3050H immediately).

c) JMP address (i.e. jump to the operand address immediately).

ii) Register Addressing Mode → In register addressing mode, the data to be operated is available inside the registers and registers are operands. Therefore the operation is performed within various registers of the microprocessor.

Scanned with CamScanner

Examples:

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

ⓑ ADD A,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).

iii) 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)

ⓑ IN 35 (read the data from port whose address is 01).

iv) 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 pair register to the accumulator).

v) Implied/Implicit/Inherent Addressing Mode → In this addressing mode the operand is hidden and the data to be operated is available in the instruction itself.

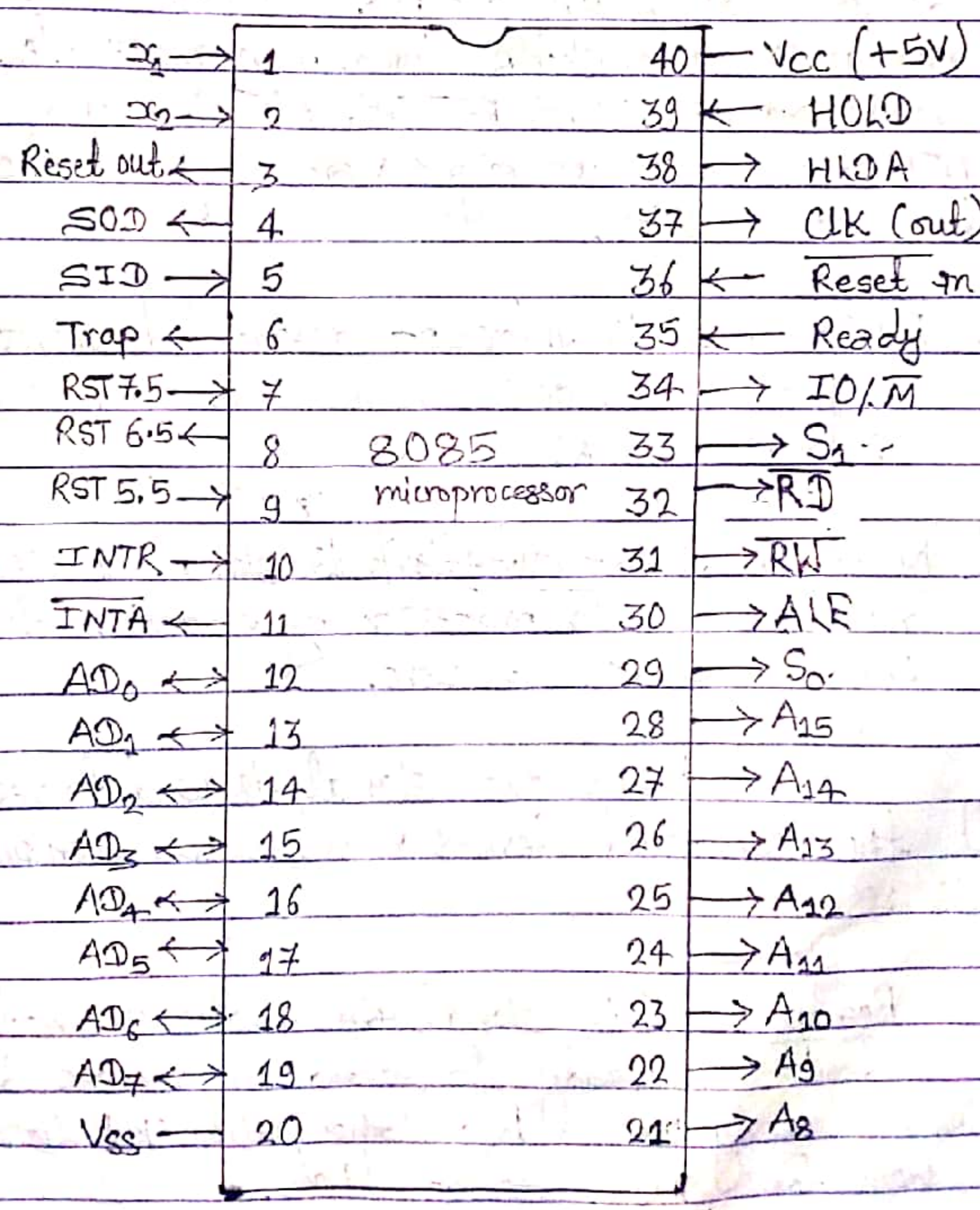
Scanned with CamScanner

Date.
Page No.

Examples:

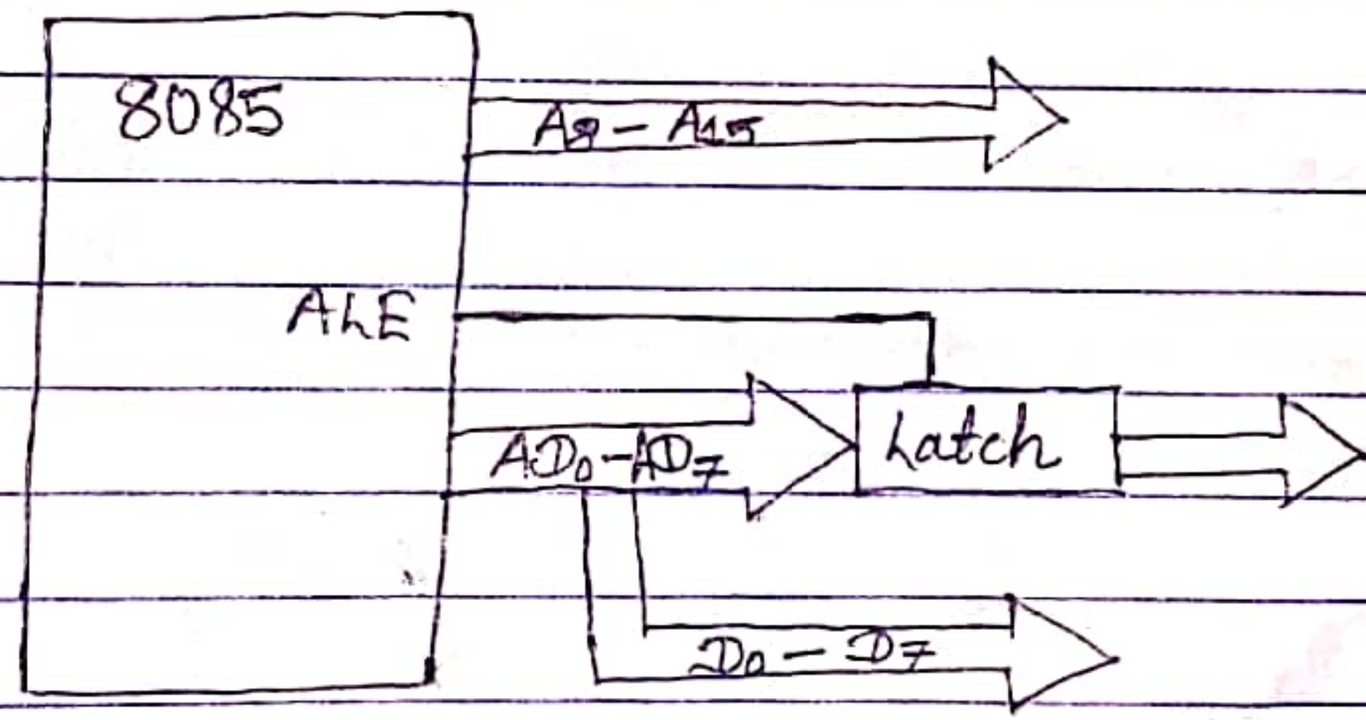
ⓐ EI (Enable interrupt)
ⓑ STC (Set the carry flag)
ⓒ NOP (No operation)
ⓓ CMA (finds and stores the 1's complement of the contents of accumulator in A).

⊛ Pin diagram of 8085 microprocessor with functions:



Scanned with CamScanner

Demultiplexing Address Data bus:-



Given, the ALE operates as a pulse during T_1 we will be able to latch the address then when ALE goes low the address is saved and the $AD_7 - AD_0$ lines can be used for their purpose as the bi-directional data ~~bus~~ lines.

Scanned with CamScanner

Date.
Page No.

Demultiplexing the Bus $AD_7 - AD_0$

→ The high order address is placed on the address bus and hold for 3 clk periods.

→ The low order address is lost after the first clk period, this address needs to be hold however we need to use latch.

→ The address $AD_7 - AD_0$ is connected as inputs to the latch 74LS373 (name of latch).

→ The ALE signal is connected to the enable (G) pin of the ~~levels~~ latch and the OC - output control of the latch is grounded.

Scanned with CamScanner

BIU → Bus interface unit
EU → Execution unit

Date:
Page No.

ES → Extra seg
CS → Code seg
SS → Stack seg
DS → Data seg
IP → Instruction pointer

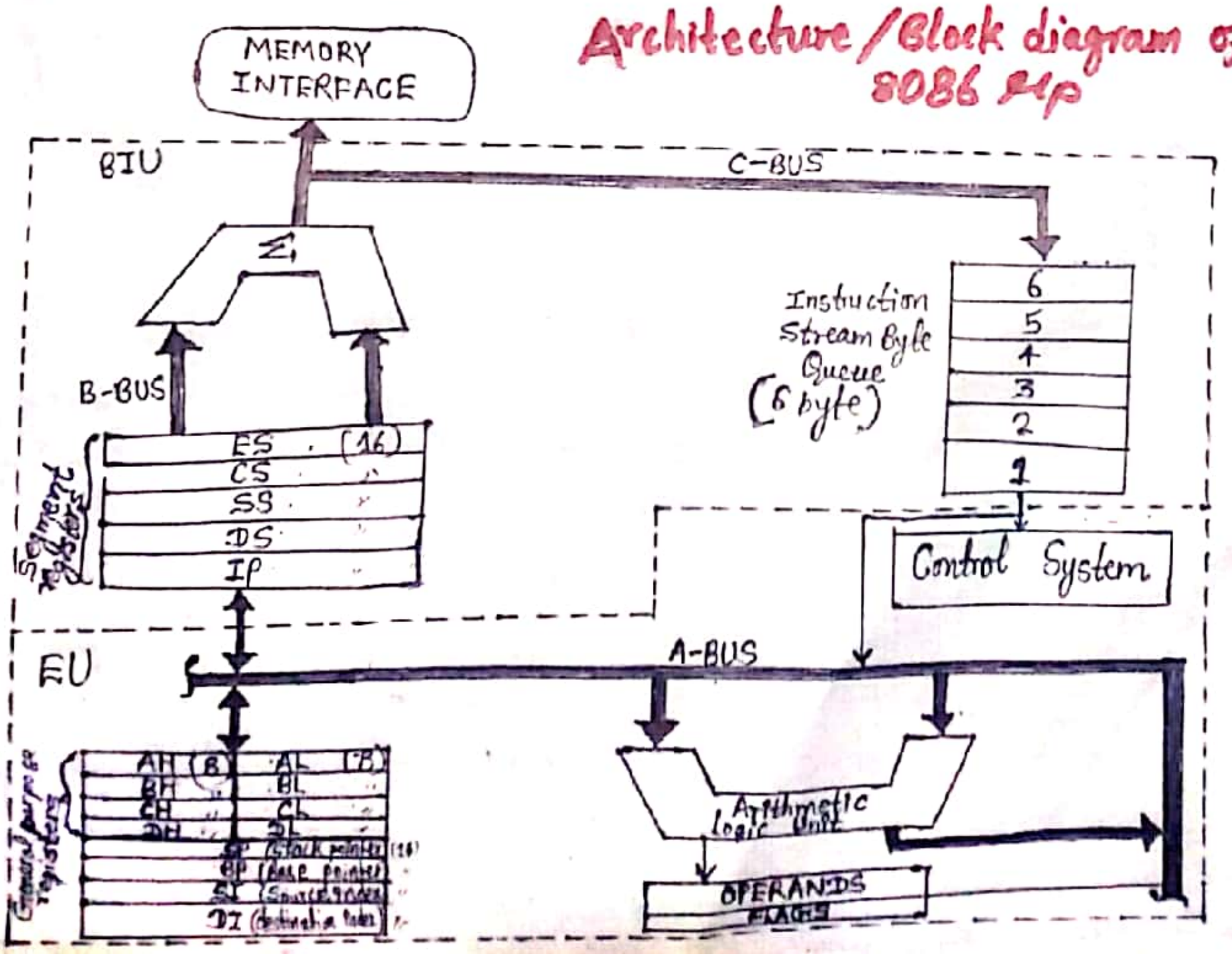
8086 Microprocessor

8086 is 16-bit microprocessor which has arithmetic logic unit capable of processing 16-bit data. It has its internal registers and most of the instruction having size 16-bit. The 8086 microprocessor has 20-bit addresses so it can address any one of 2^{20} memory locations. 8086 microprocessor is divided into two independent functional parts to speed up the processing namely BIU (Bus Interface Unit) and EU (Execution Unit).

BIU unit is responsible for all transfer a data and address on buses for execution unit as well as it fetches out instruction from memory, read/write data from/to ports & memory.

While execution unit is responsible for decoding & executing instruction & tells BIU where to fetch instruction and data from.

Architecture / Block diagram of 8086 µp



ES, CS, SS or DS

Physical address (PA) = Starting Segment address × 10H + Offset
where H represents hexadecimal

(*) Segment registers / Memory Segmentation:-

Segmentation is the process in which the main memory of the computer is divided into different segments and each segment has its own base address. It is basically used to enhance the speed of execution of the computer system, so that processor is able to fetch and execute the data from the memory easily and fast.

Need for segmentation – The Bus Interface Unit (BIU) four 16-bit special purpose registers (mentioned below) called as segment Registers.

i) Code segment register (CS) → It is used for addressing memory location in the code segment of the memory, where the executable program is stored.

ii) Data segment register (DS) → It points to the data segment of memory where the data is stored.

iii) Extra Segment register (ES) → It also refers to a segment in the memory which is another data segment in the memory.

iv) Stack Segment register (SS) → It is used for addressing stack segment of the memory. The stack segment is that segment of memory which is used to store stack data.

Scanned with CamScanner

Date.
Page No.

⊛ Instruction pointer (IP) → All program instructions located in the memory are pointed by using 20-bits which is the logical address that is the content of CS & IP. 16-bit content of CS will be shifted 4-bits to the left & then adding the 16-bit content of IP. Thus all instruction of program are relative content of IP.

⊛ Bus Interface Unit & Execution Unit:

Bus Interface Unit (BIU):

BIU takes care of all data and addresses transfers on the buses for the EU like sending addresses, fetching instructions from the memory as well as writing data to the ports and the memory. ~~as well as writing data to~~ EU has no direction connection with system buses so this is possible with the BIU. EU and BIU are connected with the Internal Bus. It has following functional parts.

i) Instruction queue → When EU is executing an instruction, bus will be idle at that time. BIU gets upto 6-bytes of instruction to be executed and place them in queue. In each time of execution of new instruction to be executed instead of sending address to system memory it just pick up the fetched instruction byte from queue. This improves the overall execution speed.

Note → fetching the next instruction while the current ~~position~~ instruction executes is called pipelining.

ii) Segment registers → Already described before.

Scanned with CamScanner

Date.
Page No.

Execution Unit:-

Execution Unit gives instructions to BIU stating from where to fetch the data and then decode and execute those instructions. Its function is to control operations on data using the instruction decorder & ALU. EU has no direct connection with system buses. It performs operations over data through BIU.

Following are the functional parts of 8086 microprocessor.

ⓐ **Instruction decorder & ALU** → Decorder in EU translate instruction phased from memory into series of action which the EU carries out. 16-bit ALU performs different operations like AND, OR, XOR, increments, decrements etc.

ⓑ **Flag register** → It has 16-bit register that behaves like a flip-flop, i.e, it changes its status according to the result stored in the accumulator. It has 9 flags and divided into two groups — Conditional flags (6) and control flags (3).

i) **Conditional flags** → It represent the result of the last arithmetic or logical instruction executed. Following are the conditional flags:

Date.
Page No.

Carry flag → This flag indicates an overflow condition for arithmetic operations.

Auxiliary flag → When the operation is performed at ALU, it results in a carry/borrow from lower nibble (i.e. $D_0 - D_3$) to upper nibble (i.e. $D_4 - D_7$), then this flag is set. i.e, carry given by D_3 bit to D_4 is AF flag. The processor uses this flag to perform binary to BCD conversion.

Parity flag → This flag is used to indicate the parity of the result. i.e, when the lower order 8-bits of the result contains even number of 1's, then the Parity Flag is set. For odd number of 1's, the Parity Flag is reset.

Zero flag → This flag is set to 1 when the result of arithmetic or logical operation is zero else it is set to 0.

Sign flag → This flag is set to 1 when the result of operation is negative, else set to 0. This flag holds the sign of the result.

Overflow flag → This flag represents the result when the system capacity is exceeded.

Scanned with CamScanner

Date.
Page No.

ii) Control flags → Control flags control the operations of the execution unit. Following is the list of control flags:

Trap flag → It is used for single step control and allows the user to execute one instruction at a time for debugging. If it is set, then the program can be run in single step mode.

Interrupt flag → It is ~~used~~ a an interrupt enable/disable flag. It is set to 0 for interrupt disabled condition.

Direction flag → It is used in string operation. As the name suggests when it is set then string bytes are accessed from the higher memory address to the lower memory address and vice-versa.

(c) General Purpose register:

There are 8 general purpose registers i.e, AH, AL, BH, BL, CH, CL, DH and DL. These registers can be used individually to store 8-bit data and can be used in pairs to store 16-bit data. The valid register pairs are AH and AL, BH and BL, CH and CL, and DH and DL. It is refered to the AX, BX, CX and DX respectively.

AX register → It is also known as accumulator register. It is used to store operands for arithmetic operations.

Scanned with CamScanner

Date.
Page No.

BX register → It is used as a base register. It is used to store the starting base address of the memory area within the data segment.

CX register → It is refered to as counter. It is used in loop instruction to store the loop counter.

DX register → This register is used to hold I/O port address for I/O instruction.

(d) Stack pointer register:-

It is a 16-bit register, which holds the address from the start of the segment to the memory location, where a word was most recently stored on the stack.

(e) Pipelining : X (from assignment copy for better).

Pipelining is an implementation technique where multiple instructions are overlapped in execution. The computer pipeline is divided in stages. Each stage completes a part of an instruction in parallel. Pipelining does not decrease the time for individual instruction execution. Instead, it increases instruction throughout.

Pipelining is the process of fetching the next instruction when the present instruction is being executed. It has become possible due to the use of 6-byte queue. The advantages of pipelining are: it eliminates the waiting time of EU and speeds up the ~~the~~ processing and the execution unit always reads the next instruction byte from the queue in BIU which is faster than sending out an address to the memory and waiting for the next instruction byte to come.

Scanned with CamScanner

Date.
Page No.

Differences between 8085 and 8086 µp :

The differences between 8085 and 8086 microprocessor are as follows:-

	8085 microprocessor	8086 microprocessor
Size →	i) 8085 is 8-bit microprocessor.	i) 8086 is 16-bit microprocessor
	ii) 8085 microprocessor contain address bus of 16-bit	ii) 8086 microprocessor contain address bus of 20-bit.
	iii) It does not have instruction queue.	iii) It have instruction queue.
	iv) It does not support pipelined architecture.	iv) It supports pipelined architecture.
I/O →	v) It can address 2^8 = 256 I/O's.	v) It can address 2^{16} = 65,536 I/O's.
Cost →	vi) 8085 microprocessor is of lower cost.	vi) 8086 microprocessor is of higher cost

Scanned with CamScanner