8086

* How define processor? → 8 bit / 16 bit / 32 bit

Sol ① in terms of bits.

② How many bits does the reg. of a processor have?

16-bit pro. (8086)

→ has 16 data lines

→ all internal registers are 16 bit long.

32-bit processor.

→ All int reg. are 32 bit long.

→ Has 32 bit data bus

① CS → holds inst codes of a progra

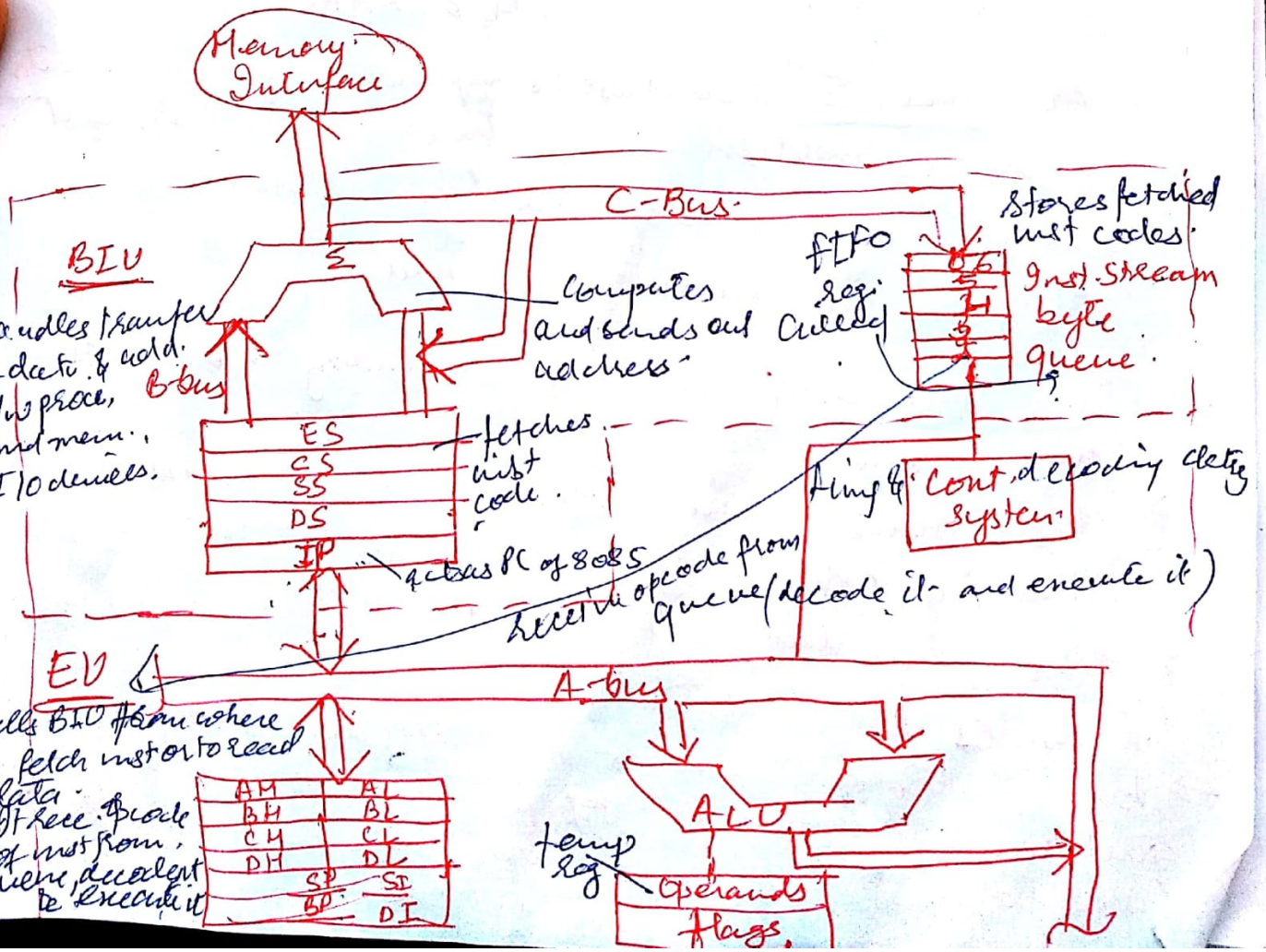
② DS → hold data, variable and constants

③ SS: addresses and data of subroutines also hold contents of reg or ML's given in Push inst.

ES: holds the destination addresses of some data of certain string inst.

8086 has 16 bit data bus and a 20 bit add. bus

It can address 2^{20} = 1MB unique memory.



It has two units: EU & BIU.

EU → executes inst. for the processor.

→ It consists of ALU, which performs the A&L operations required of any program.

→ Consists of GPRS, that can be addressed either as 8-bit or 16-bit reg. → DX use in operand and destination inst.

→ 8-bit → AL — Acc. low
AH — Acc high
16-bit → AX
} for all others same.

→ BX – use as offset storage to form phy. add.
→ CX – CX use as default counter in string and loop inst.

BIU → It has segment registers used to address memory space (either ROM or RAM or I/O).

→ Also has [Σ] address compute engine converts the logical address that is held by the segment registes into a phy. address (that is exposed into outside world).

→ Inst. queue (Internal memory holds the inst queue.

→ These units will access to the outside world either by add or data bus or I/O ports.

→ BIU handles all the data and address on the busses for the EU.

→ bus operations includes inst. fetching, reading and writing operands for memory and calculating the addresses of the memory operands

→ Inst are transferred to the inst queue. this is used to hold 6 consecutive inst.

→ Why required? Because EU of processor is much faster than peripheral devices (like memory ROM or RAM). Because nowadays processor becomes faster).

→ "The inst queue permits to store in advance specific instructions that would be executed in the future." This whole concept is called "pipelining". "pipelining is a technique wherein the bus interface unit pre-fetches instructions for the EU to use. In case of 8086 upto six inst. are pre-fetched."

→ It helps EU to perform at optimum speed. and only when jump inst has to execute pipe needs to be refreshed other it will be filled in normal sequence of inst.

Now segment registers: [ES, CS, SS, DS], IP.

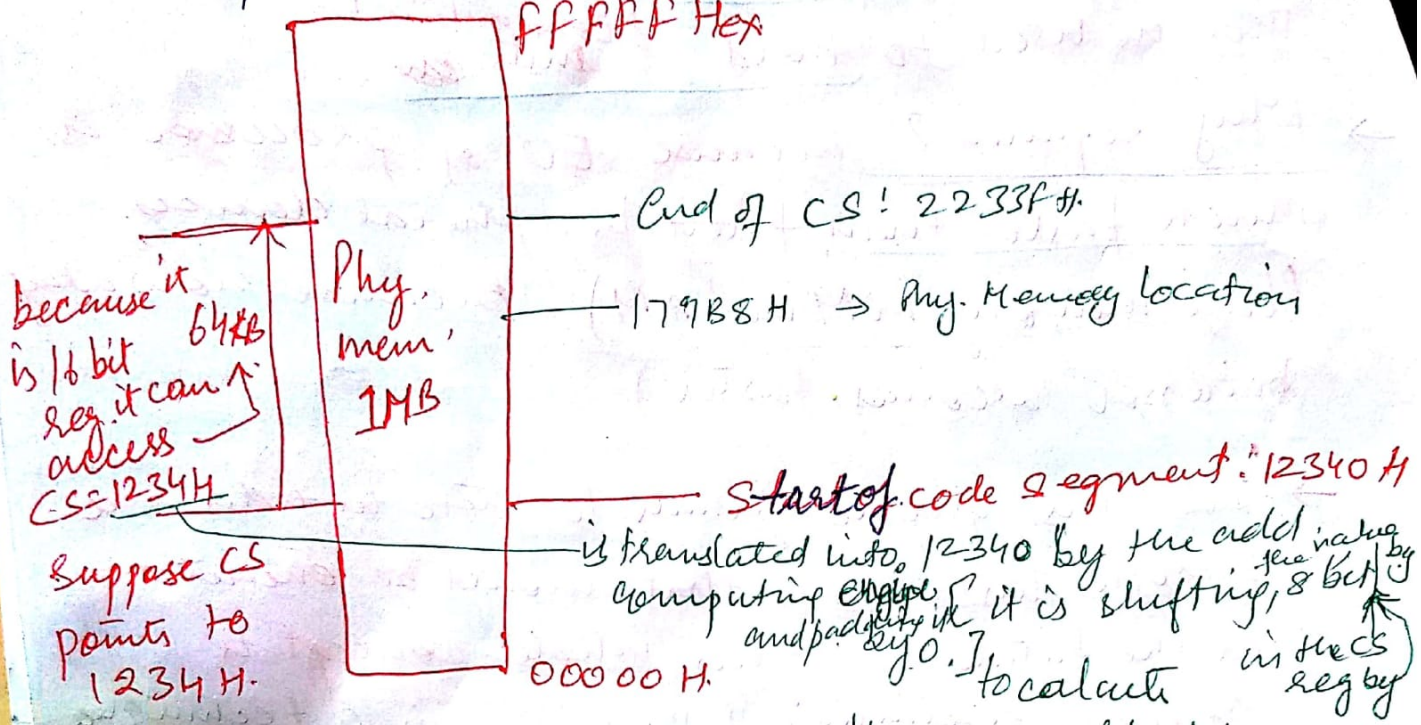
Each SR is 16-bit long and has 2^{16} or 64KB address.

→ They are used to perform specific operations

→ different kinds of SR's are used to calculate the addresses within the 1MB memory space.

CS	code seg. reg. use for addressing mem. location
SS	stack seg. reg. use addressing stack segment of memory.
DS	data seg Reg use data seg. of memory.
ES	extra seg. reg. also use data of memory.

During addressing in a memory bank location add. calcu
in two parts : seg add + offset add.



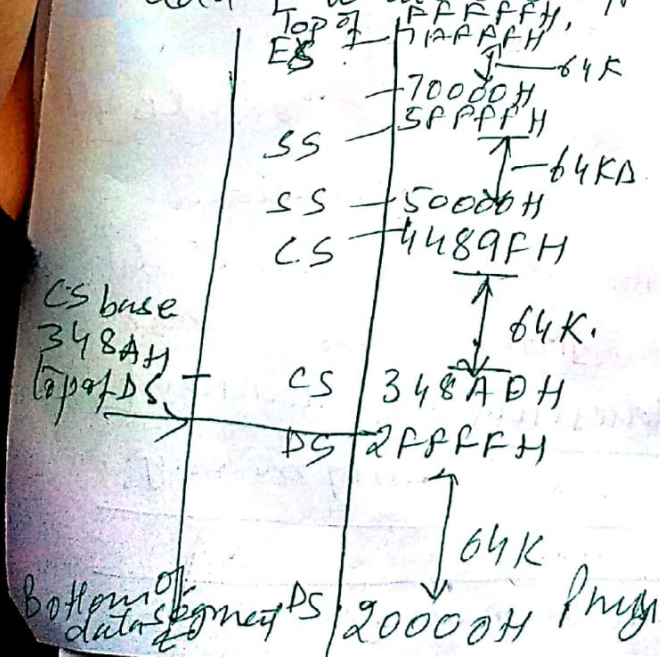
to calculate the phy. add of the start of CS.

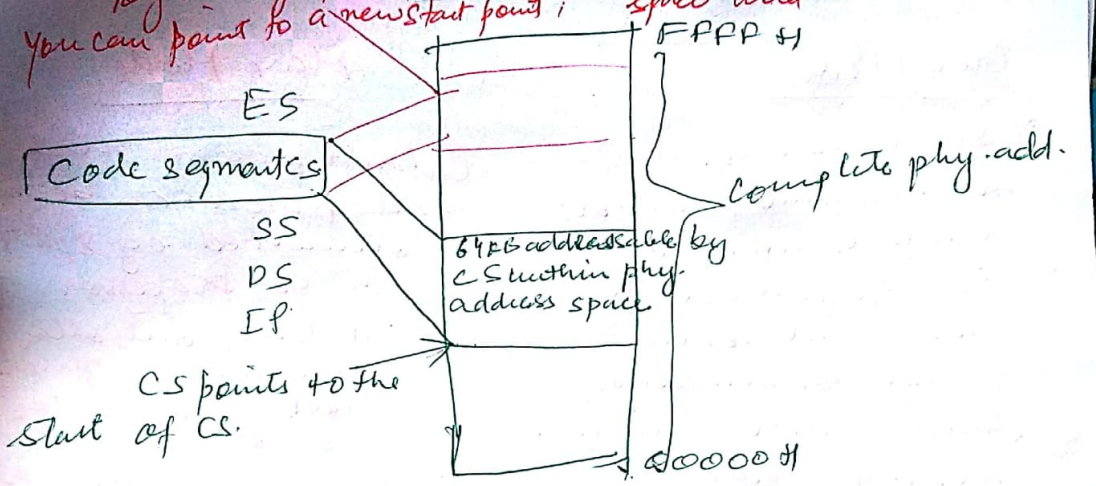
The IP = 5678 H

CS : 12340
IP : 05678

Phy. add. : 179B8

Thus, two 16 bit reg. are used to calculate one 20 bit physical add which is o/p to A_0 to A_{19} (20 bit long).





* EU consists of GP Register's.
- AH, AL → AX - Accumulator
- BH, BL, BX
- CH, CL, CX
- DH, DL, DX

All arth and logical computation
Memory pointers in DS and other addressing modes
Counter reg used in loop inst
Used in divide & multiplication operations.

Pointers and Index reg contain offset within a particular segment
- SP, BP
- SI
- DI
- IP

contain offset
use for store offset of source data in data segment
use for store offset of destination data in DS.

→ Index reg: SI and DI are used primarily in string operations.

→ Pointer reg: BP → points to the location within the SS where the data is.
SP → points to a specific mem. location within the SS.

→ Flag reg: shows cond. of changes due to execution of instructions.

X	X	X	X	O	D	I	T	S	Z	X	AC	X	P	X	CY

Modes of operation

Mini Mode

1. Single processor mode. no additional processor can be connected
2. 8086 responsible for generating all control signals for mem. and I/O.
3. used to design sys. used in simple app.
4. pin 30 = 1 to enable min mode.

Max Mode

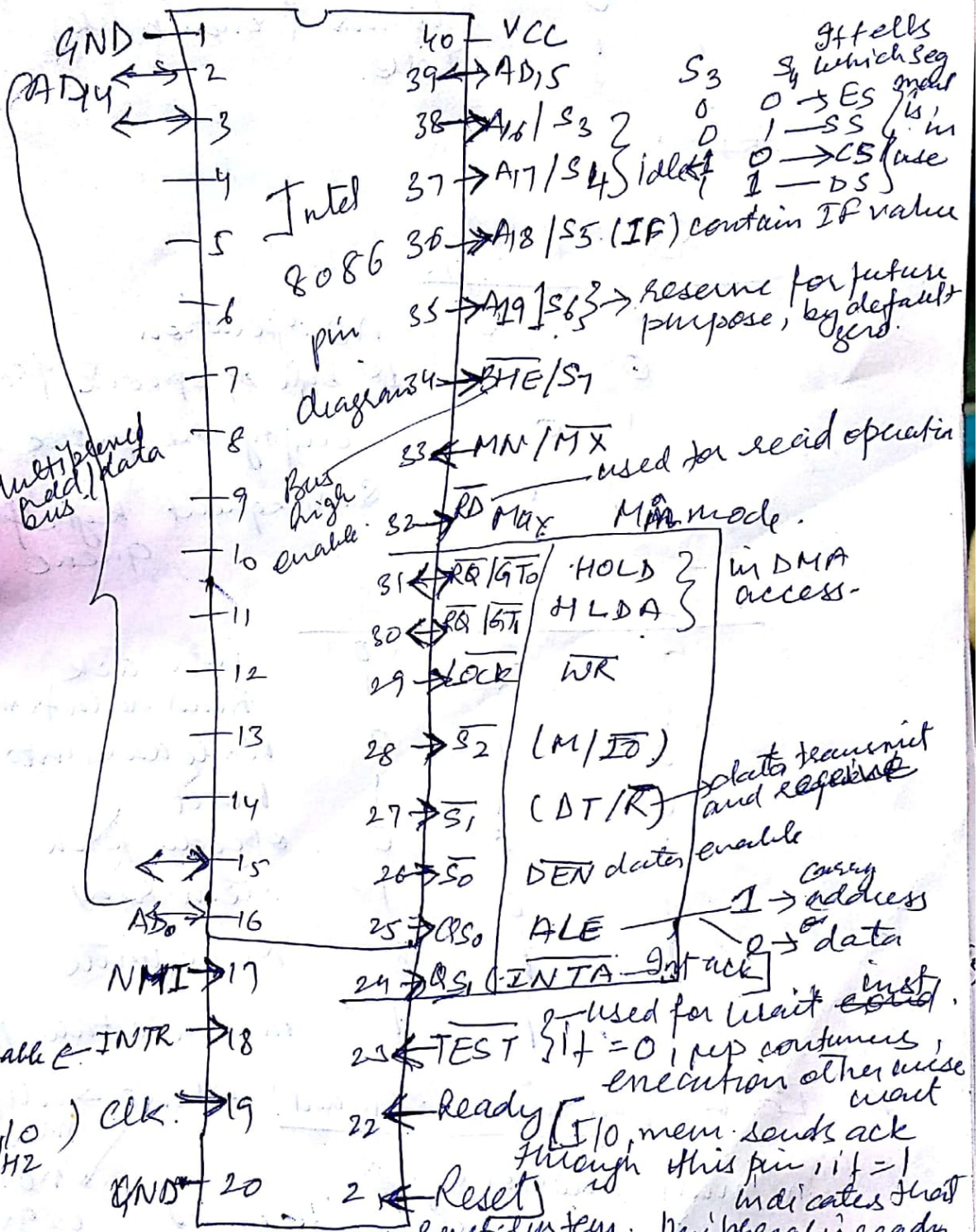
1. Multiprocessor mode. add-process can be connected eg 8087 with co-processor.
2. Ext bus controller responsible for generating all control signals for mem. and I/O.
3. used for complex and large applications.
4. Pin 30 = 0 enable max mode.

8086 flag reg: 0 -> if result is > than destination reg.

- S. if result -ve, flag is set, denoted by MSB.
- Z. if prev inst is zero this flag is set.
- P. if lower byte of result contain even no. of 1's this flag is set.
- CY if carry produce in MSB this flag is set.
- T. if process in single step execution mode, this flag is set
- I if Maskable inter detect by CPU, this flag is set.
- D this flag is 0 string is process in auto increment mode. and 1 in auto decrement mode.

* PIN DIAGRAM (1978) first 16-bit µp.

Add bus → 20 bit
Data bus → 16



→ if 1 µp connected with I/O, memory connected along with Sys. bus then min. mode.

→ 3. MN/MX → GND → MAX
MN/MX → Vcc → Min

$\overline{BHE}$	S_7	(It is used to enable data onto the half of data bus $D_8 - D_{15}$)
0	0	Whole
0	1	Upper (odd bank)
1	0	Lower (even bank)
1	1	Idle

* ~~RD~~

QS_0	QS_1	
0	0	no operation
0	1	1st byte of opcode from queue
1	0	empty the queue
1	1	Subsequent byte from queue.

$\overline{S_2}$	$\overline{S_1}$	$\overline{S_0}$	
0	0	0	Intr ack
0	0	1	Read data from I/O port
0	1	0	Write data into I/O port
0	1	1	Halt
1	0	0	opcode fetch
1	0	1	Mem. Read
1	1	0	Mem. write
1	1	1	Passive state / None

* $\overline{Lock}$: active low signal. when 0 → all intrpts are masked
→ no hold request is granted.

In multiprocessor sys. All other processors are informed by this signal that they should not ask the CPU for relinquishing the bus control.

* $\overline{RQ}/\overline{GT_1}$, $\overline{RQ}/\overline{GT_0}$ (bidirectional)

Local bus Priority control signal.

→ Other processors ask the CPU through these lines to release the local bus.

→ $\overline{RQ}/\overline{GT_0}$ has higher priority than $\overline{RQ}/\overline{GT_1}$

→ In max. mode of operation signals $\overline{WR}$, ALE, $\overline{DEN}$, $DT/\overline{R}$ are not available directly from processor.

→ These signals are available from the controller 8288 (bus controller).