

Micro Topics:

- 1. Microprocessor history and evolution.
- 2. How processor works? using diagram.
- 3. Three BUSES (Address, Data, Control bus)
- 4. Difference between micro controller and microprocessor
- 5. Internal architecture of 8086 + Pipelining
- 6. Flag registers.
- 7. Physical address calculation. (Page 57)
- 8. Addressing modes (Page 79 + Photo)
- 9. Hardware specification of 8086 (MAX & MIN) (Page - 306 also)
- 10. Demux
- 11. Latching
- 12. Buffer
- 13. BUS timing diagram. (Read / Write)

~~15.~~ Memory interfacing (Page 329)

~~16.~~ Volatile, Non-volatile memory (def + diff)

~~17.~~ SRAM, DRAM (def + diff) + RAM - ROM (same)

~~18.~~ Address decoding (Page 340 to 342)

- Simple NAND gate decoder

- 74LS138 decoder (P-341 + 343)

~~19.~~ Odd and Even memory bank (Page 357)

~~20.~~ I/O Interfacing:

① IN, OUT, Fixed, variable address (P-378)

② 4 types of I/O

③ PPI \rightarrow IC 82C55 (Page 395)

④ I/O internal block diagram

⑤ Mode register (3 modes - page 397)

⑥ Command byte A and B. (page 398)

20. Interrupts: (Page 452) (~~Type~~)

① Interrupt types (Type 0 - Type 5)

② Interrupt vector (Page - 453)

21. Hardware interrupt (Page - 459 → 12.2)

22. Operation of a real mode interrupt (P 456 + 455)

23. Programmable Interrupt controller (d259A)

config. → mod. Status Register → Class note

24. Chapter 13 :

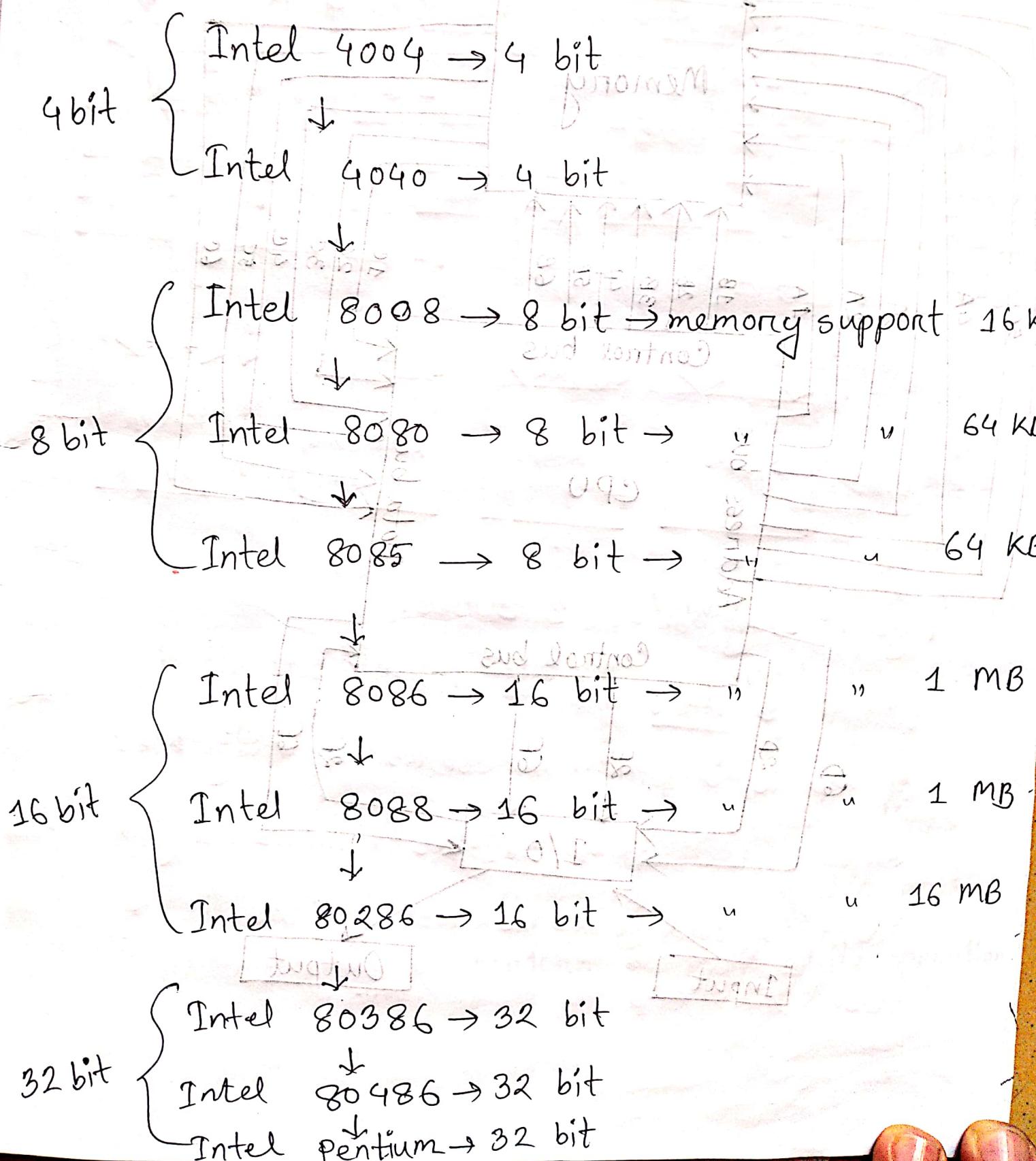
① Basic DMA operation (Page 490 → 13.1)

② 8237 DMA controller.

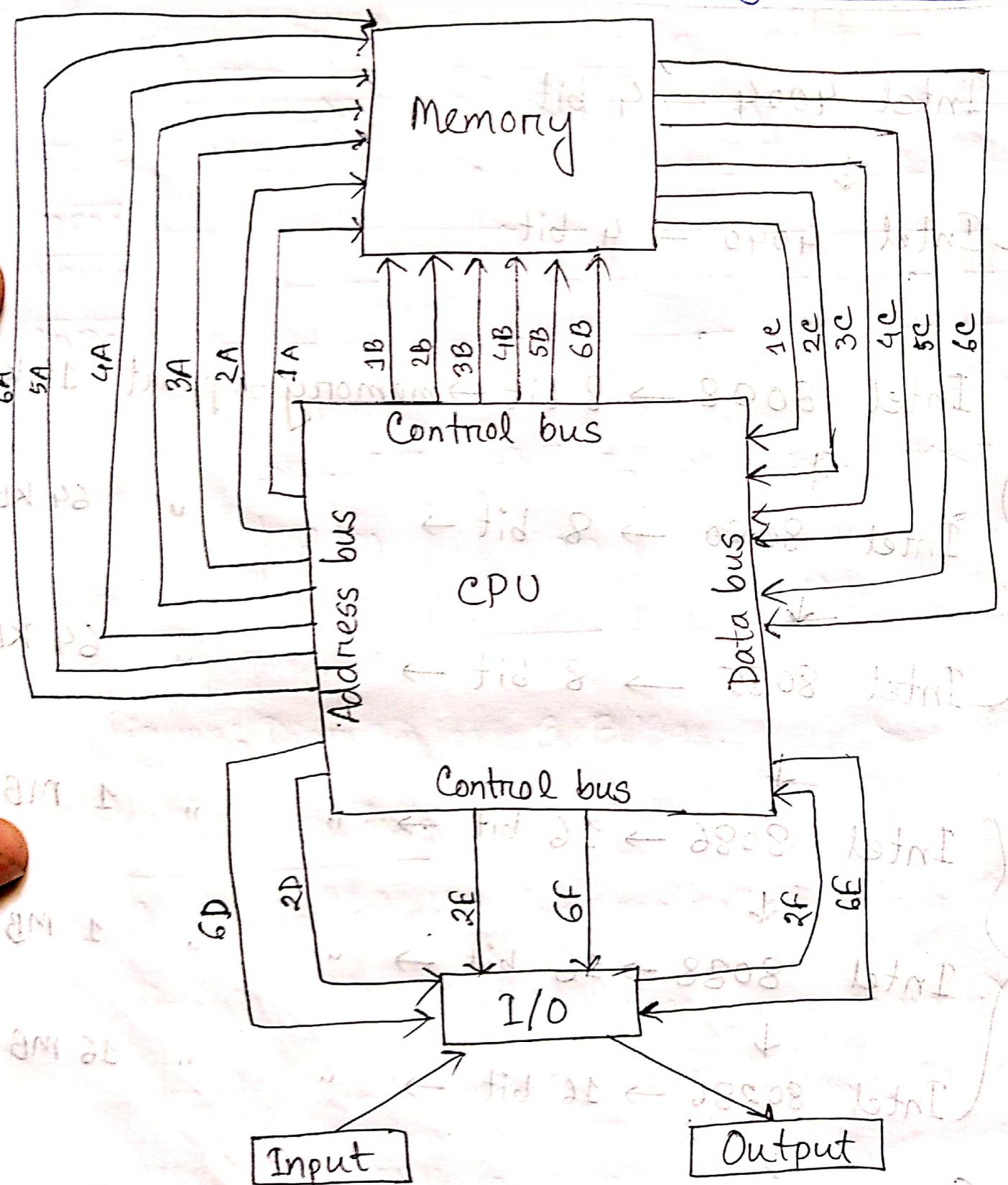
25. Assembly Language

Reset > Hold > NMI > Interrupt

History and evolution of microprocessor



How processor works (using diagram):



Three BUSES:

BUS: A bus is a common group of wires

that interconnect components in a computer system.

3 BUSES are -

① Address BUS - it requests a memo

location from the memory or an I/O location from the I/O devices.

② Data BUS - it transfers information

between CPU and its memory and I/O address

space.

③ Control BUS - it contains lines

that select the memory and/or I/O and cause them to perform a read/write operation.

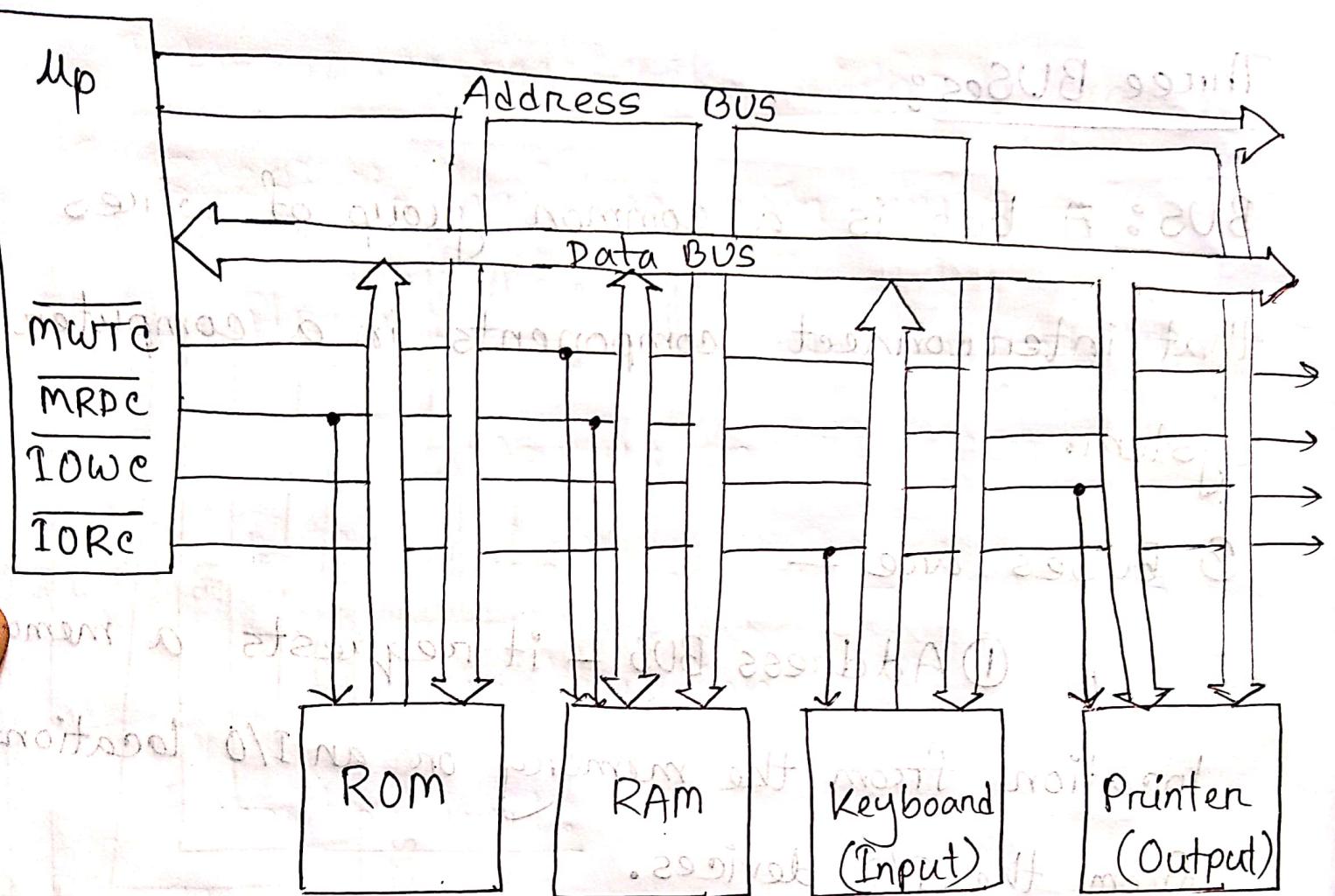


Figure: Diagram of 3 Buses.

What is Mp:

A Mp is a program controlled integrated circuit (IC) which fetches (from memory), decodes and encodes instructions. It is used as CPU in computer.

Difference between Microprocessor and Microcontroller

Microprocessor	Microcontroller
I Heart of computer system.	I Heart of embedded system.
II Just a processor, memories and I/O are connected externally.	II Memories and I/O are embedded with processor.
III Doesn't have power saving feature.	III Have power saving mode.
IV Large in size.	IV Small in size.
V High cost.	V Low cost.
VI Less number of registers.	VI More number of registers.
VII Based on Neumann Architecture.	VII Based on Harvard architecture.

Internal architecture of 8086

8086 Up has two main units —

① BIU - BUS Interface Unit - memory

instruction নিয়ে আজি।

② EU - Execution Unit - executes

instructions.

Execution unit:

It has 4 parts.

① Control circuit

② Instruction decoder

③ ALU - Arithmetic Logic Unit

④ Flag register

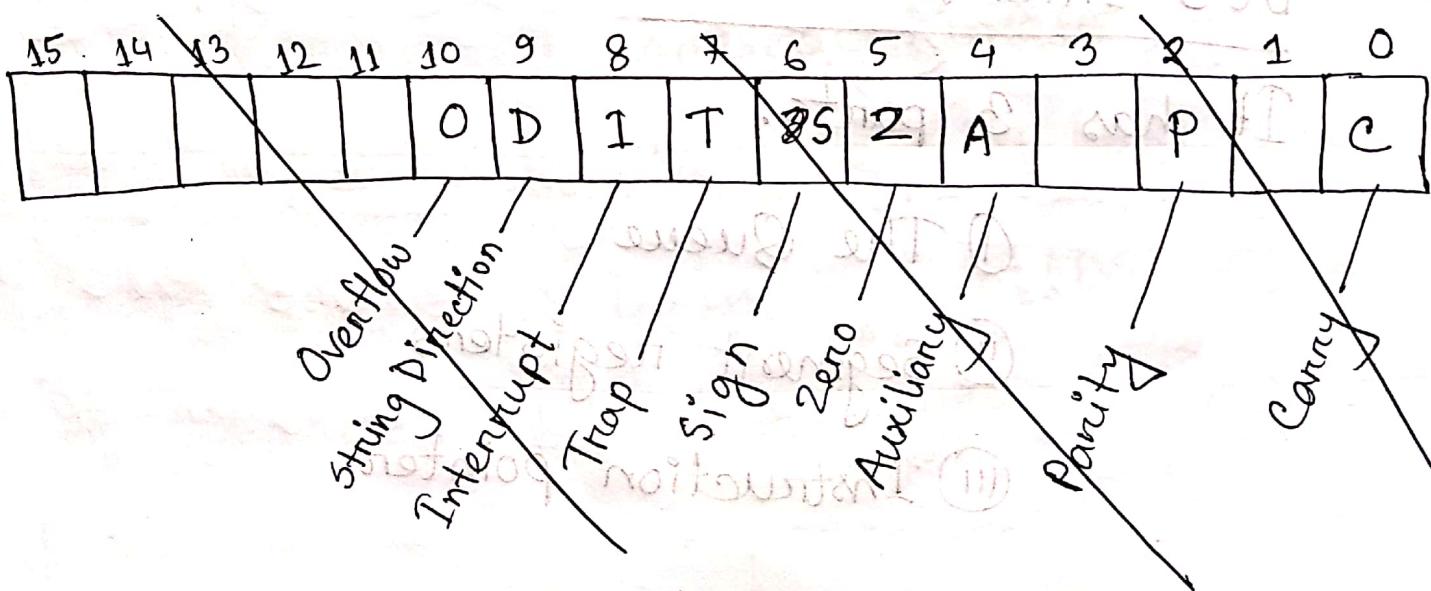
Control circuit - directs internal operations.

■ Instruction decoder — translates instructions fetched from memory into a series of actions which the EU carries out.

■ ALU — 16 bit arithmetic logic unit can add, subtract, AND, OR, XOR, increment, decrement, complement or shift binary numbers.

■ Flag registers — 16 bit register.

A flag is a flip-flop. Among them, 9 flags are active.



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				O	D	I	T	S	Z	A		P		C	

O - overflow

A - Auxiliary

D - String direction

P - Parity

I - Interrupt

C - Carry

T - Trap

S - Sign

BUS Interface Unit:

It has 3 parts.

① The Queue

② Segment registers

③ Instruction pointer

The Queue — While the EU is decoding an instruction or executing an instruction which doesn't require use of buses, the BIU fetches up to six instruction bytes for the following instructions. The BIU stores these prefetched instructions in a first-in-first-out register called the queue.

Segment Registers — Four segment registers are used. These are —

(a) Code Segment (CS)

(b) Stack Segment (SS)

(c) Data Segment (DS)

(d) Extra Segment (ES)

— Code segment is a section of memory that holds the code used by the program.

— Data segment is a section of memory that contains most data used by a program. Data are accessed in the data segment by an offset address.

— Stack segment defines the area of memory used for the stack.

(22) — Extra segment is an additional

(23) — data segment used by some of the string

(24) — instructions to hold destination data.

④ Instruction Pointer - It holds the 16-bit address or offset of the next code byte within ~~this~~^{the} code segment which holds the upper 16-bits of the starting address.

General purpose registers:

These registers are mainly part of EU.

EU has eight (08) general purpose registers.

These are -

AH AL \rightarrow AX (Accumulator)

BH BL \rightarrow BX (Base Index)

CH CL \rightarrow CX (Count)

DH DL \rightarrow DX (Data)

These can be used individually for temporary

storage of 8-bit data. Here AL register

is also called the accumulator.

Advantage of using internal registers:

The advantage of using internal registers for the temporary storage of data is

that, since the data is already in EU,

it can be accessed much more quickly

than it could be accessed in external

memory.

Pipelining: Pipelining is the process

where the next instruction is fetching

while the current instruction is execu-

ting.

Physical address calculation

To get the physical address, a zero (0) is needed to the right of the segment base address (this zero is called hardwired zero) and then add this with the offset.

Example:

CS	3	4	8	A	handwired zero ↓ 0.	(Segment Address)
IP (+)	4	2	1	4		(Offset Address)
	3	8	A	B	4	Physical address.

SS	5	0	0	0	handwired zero ↓ 0.	points
SP (+)	F	F	E	0		points
	5	F	F	E	0	Physical address (top of the stack)

Default segment and offset registers:

Segment	Offset	Special purpose
CS	IP (instruction pointer)	Instruction address
SS	SP (Stack pointer) BP (Base)	Stack address
DS	BX, DI, SI etc	Data address
ES	DI for string instructions	String Destination Address

Addressing modes:

④ Register:

MOV AX, BX

BX এবং data AX এ ~~একটি~~ copy করে।

④ Immediate:

MOV AX, 473BH

473BH এই data AX এ copy করে।

④ Direct:

MOV AX, [1234H]

Offset address

Data segment \times 10H + offset করে address

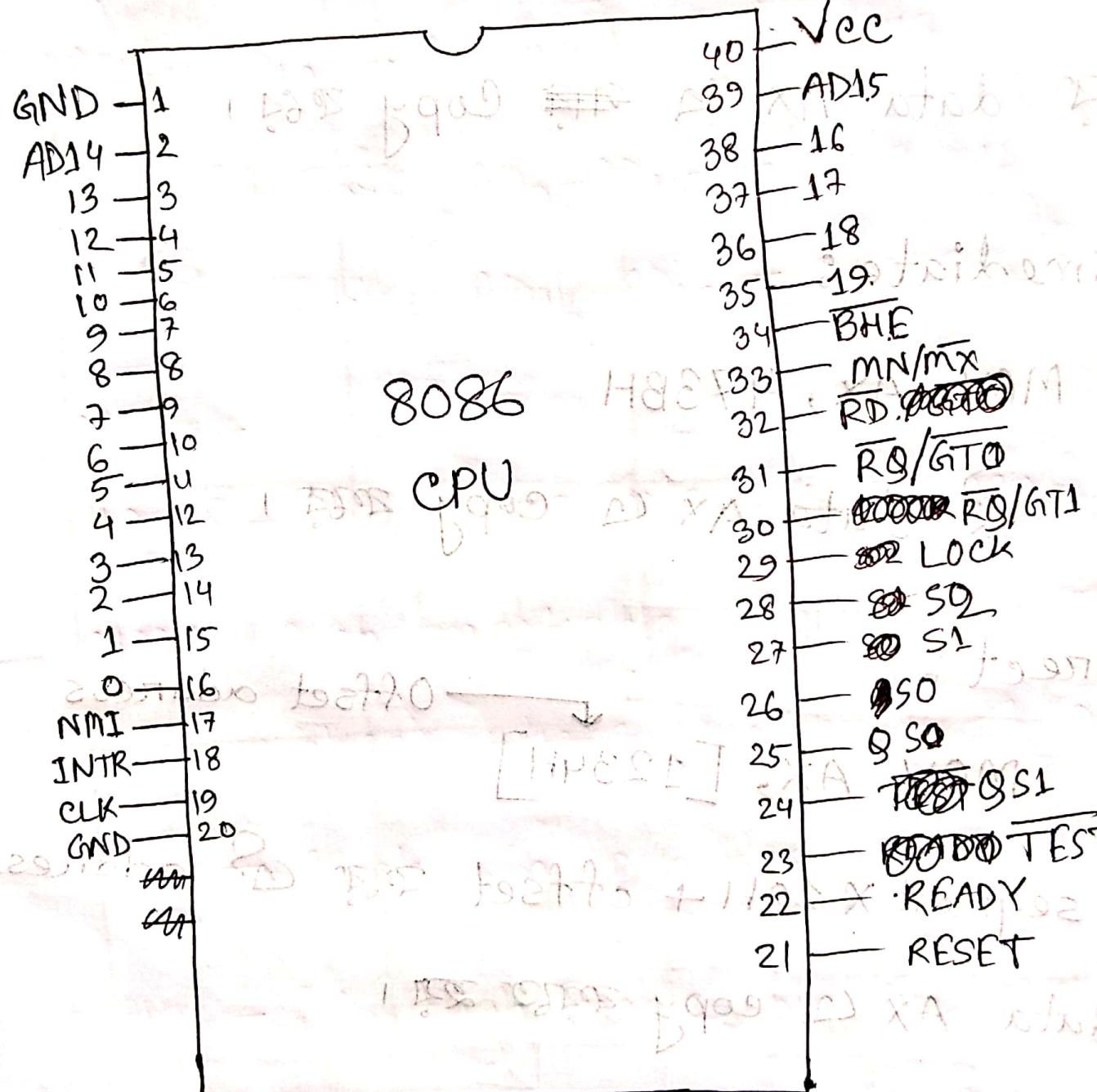
এই data AX এ copy ক্ষমতা রয়ে।

④ Register indirect:

MOV CX, [BL]

BL sized data এ address করে data CX এ copy রয়ে।

Hardware specification of 8086



	Pin	Description	Pin	Description
GND	1	0V GND	40	Vcc +9V
AD14	2		39	15/100
13	3	and stop - 21DA	38	16/S3
12	4	and stop - 21DA	37	17/S4
11	5	and enable - 21DA	36	18/S5
10	6	and enable - 21DA	35	19/S6
9	7		34	BHE/S7
8	8	8086 CPU instruction	33	MN/MX
7	9		32	RD
6	10	CPU	31	\overline{RQ}/GTO
5	11	bus lock	30	$\overline{RQ}/GT1$
4	12		29	LOCK
3	13	enable signal generation	28	S2 HOLD
2	14		27	S1
1	15	ref bus enable	26	$\overline{S0}$ DEN
0	16	and stop instruction	25	QSO
NMI	17	enable interrupt	24	QS1 INTA
INTR	18	int. request	23	TEST
CLK	19		22	READY
GND	20	go to instruction	21	RESET

GND

V_{cc} — Power supply to the MP (5.0 V)

GND — Return for the power supply.

AD₀ — AD₁₅ — Address/data bus

এবং এটি AD₀—AD₁₅ — data bus

AD₁₆—AD₁₉ — address bus

NMI

INTR — Hardware interrupt when IF = 1.

NMI — Non-maskable interrupt but it doesn't

check ~~whether~~ whether IF = 1 or not.

CLK

Clock pin provides basic timing signal

BHE

BUS High Enable pin is used to enable

ATH

the most significant data bus bits.

MN/MX

Minimum or maximum pin indicates

the condition of MP.

RD — Read signal.

TEST — Test pin is an input that is tested by the WAIT instruction.

READY — it is controlled to insert wait states into the timing of M_p.

RESET — To reset the M_p.

Max

• RQ/GT1 and RQ/GTO — request or grant pins, request direct memory access (DMA).

LOCK — to lock peripherals off the system.

S0-S2 — Status bit indicates current bus cycle.

QS₁-QS₀ — Queue bits shows the internal instruction queue.

M_{IN}

HOLD - Hold input request a direct memory access (DMA)

HLDA - Hold acknowledge indicates the CPU has entered in hold state.

WR - Write line

M/I_O - this pin selects M or I_O

DT/R - Data transmit/Receive

DEN - Data Bus Enable activates external data bus buffers.

ALE - Address Latch Enable

INTA - Interrupt acknowledge is a response to INTR pin.

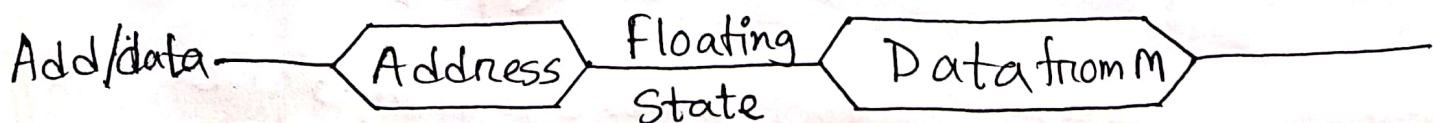
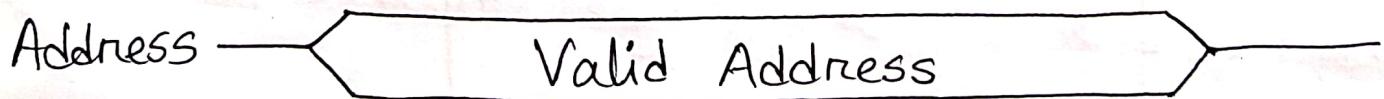
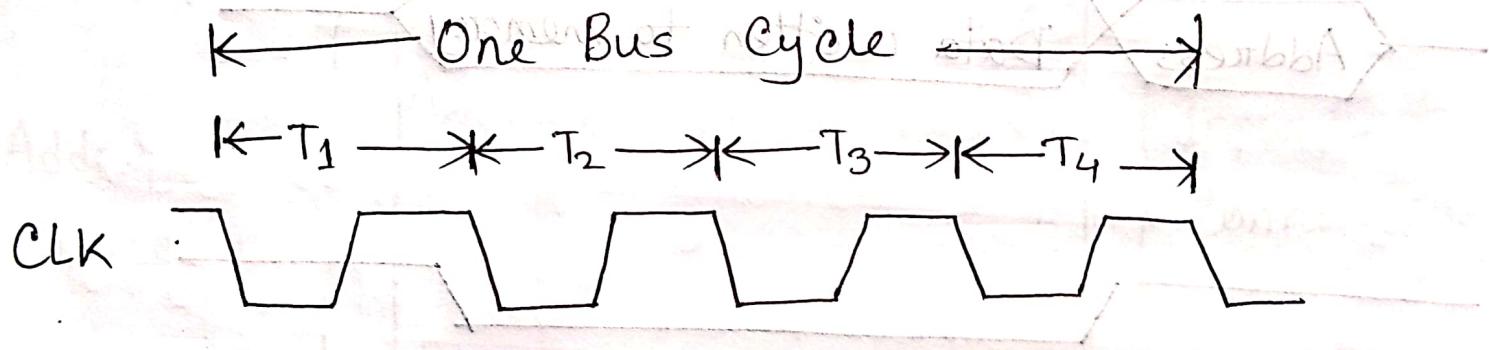
BUS Timing Diagram

Two types —

① Read bus cycle

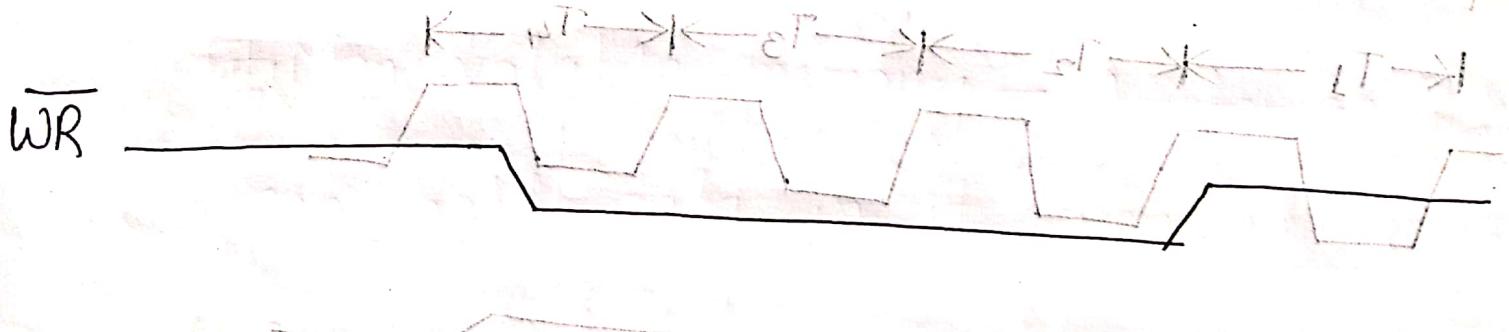
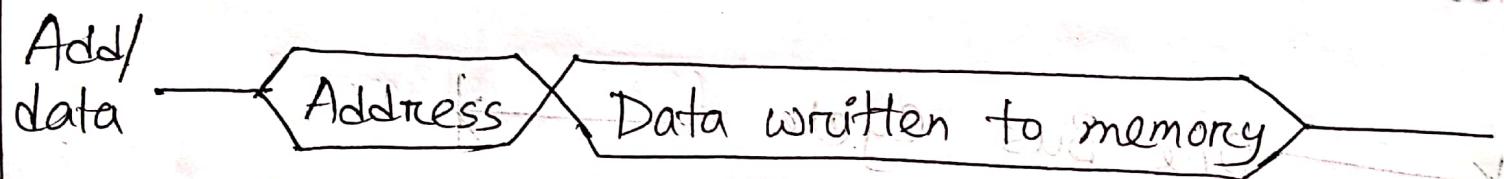
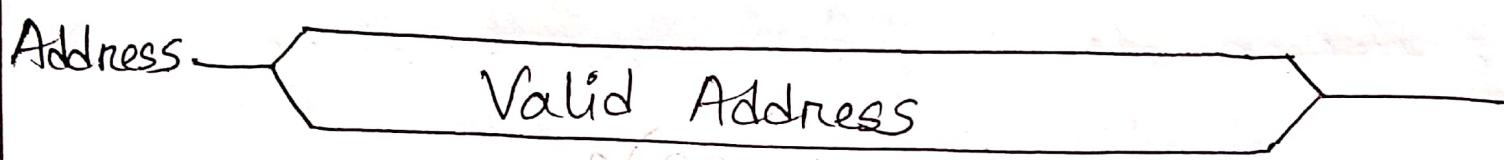
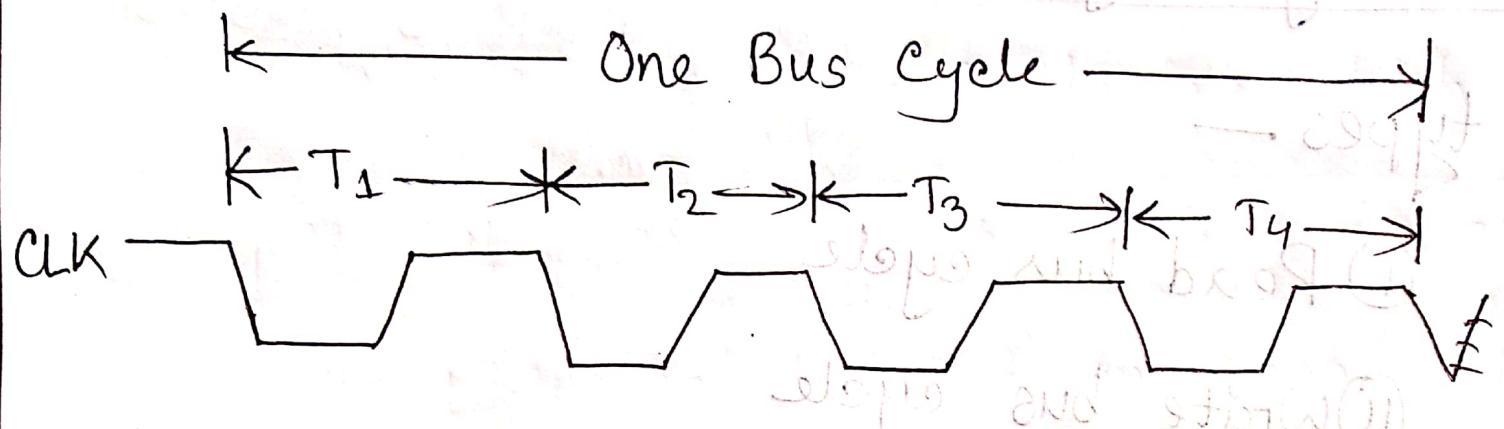
② Write bus cycle

■ Read Bus Cycle (8086/8088) :



\overline{RD}

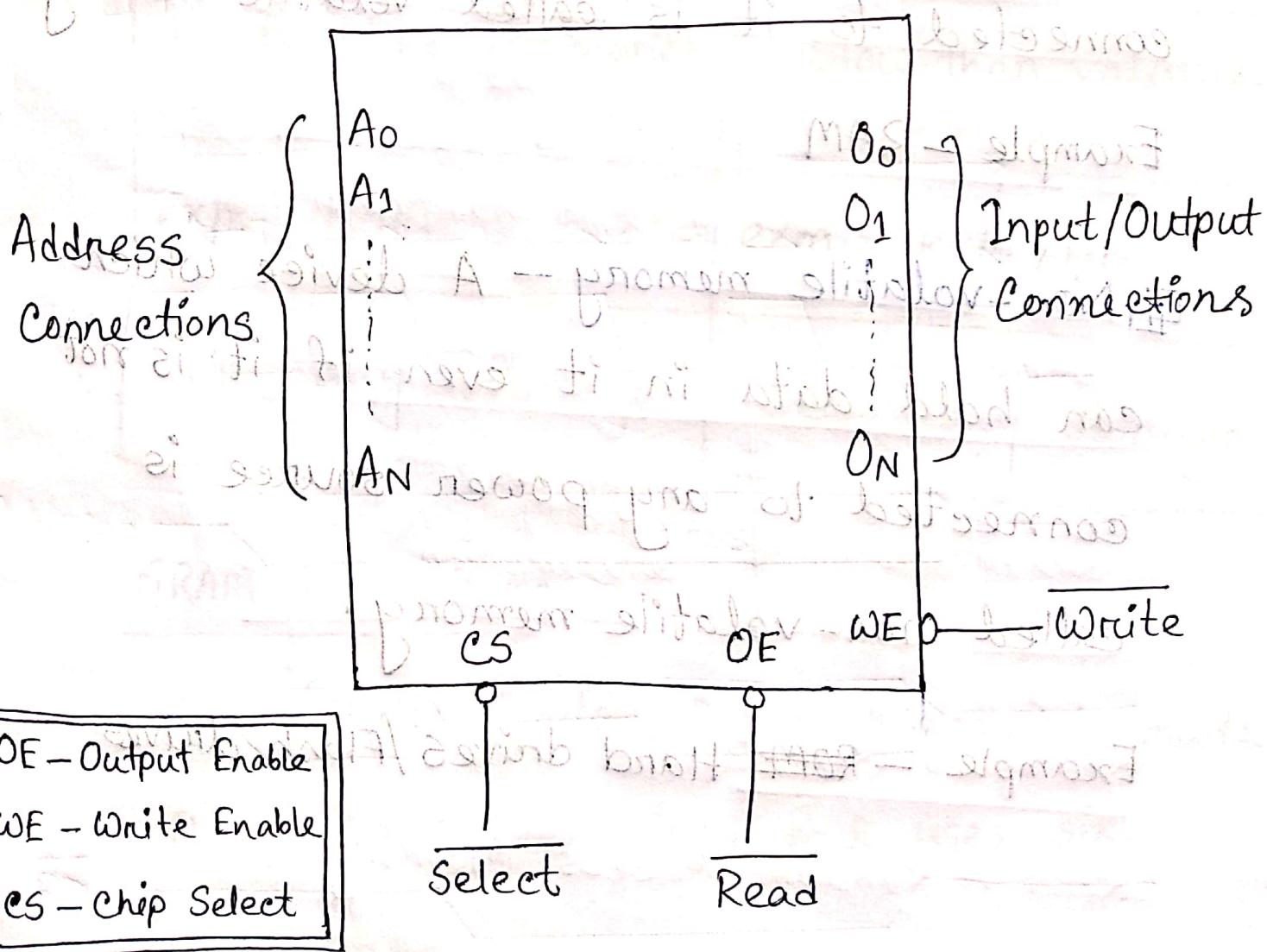
Write Bus Cycle (8086/8088)



Memory Interfacing

A pseudo-memory component illustrating

the address, data and control connections
is shown below -



* OE and WE are control connections.

Volatile vs Non-volatile memory

By Volatile memory - A device which holds the data as long as it has power supply connected to it and can't hold the memory when there is no power supply connected to it is called volatile memory.

Example - RAM

By Non-volatile memory - A device which can hold data in it even if it is not connected to any power source is called non-volatile memory.

Example - ~~ROM~~ Hard drives / Flash drives

Q1 Difference between volatile and non-volatile:

Volatile	Non-volatile
Definition	Definition
Temporary	Not temporary
Fast performance	Slow than volatile
ex: RAM	ex: ROM / HDD
Primary storage	Secondary storage

SRAM vs DRAM:

SRAM (Static Random Access Memory) is made

up of CMOS technology and uses six transistors.

Row address bus — RAM

~~Jitter over basic library~~ DRAM (Dynamic Random Access Memory) is made of capacitors and few transistors.

SRAM	DRAM
Faster	Slower
Small	Large
Expensive	Cheap
used in cache memory	used in main memory
Complex	Simple
Low power consumption	High power consumption

~~RAM vs ROM~~ RAM vs ROM

RAM - Random Access Memory

ROM - Read Only memory

RAM	ROM
Volatile	Non-volatile
High speed	Slower than RAM
CPU can access data stored in it	CPU directly can't access data stored in it
Large size	Small size
High capacity	Less capacity
Expensive	Way cheaper than RAM