CS & IT ENGINEERING

COMPUTER ORGANIZATION AND ARCHITECTURE

Basics of COA

Lecture No.- 03

By- Vishvadeep Gothi sir

Recap of Previous Lecture

- Topic: CPU Registers
- Topic: Types of Architecture
- Topic: Program Counter
- Topic: Instruction Register
- Topic: Stack Pointer

Topics to be Covered

- Topic: CPU Registers
- Topic: Memory Addressing
- Topic: Memory Access
- Topic: Architecture Type (Based on Size of Input)

Topic : Address Register or MAR

(AR)

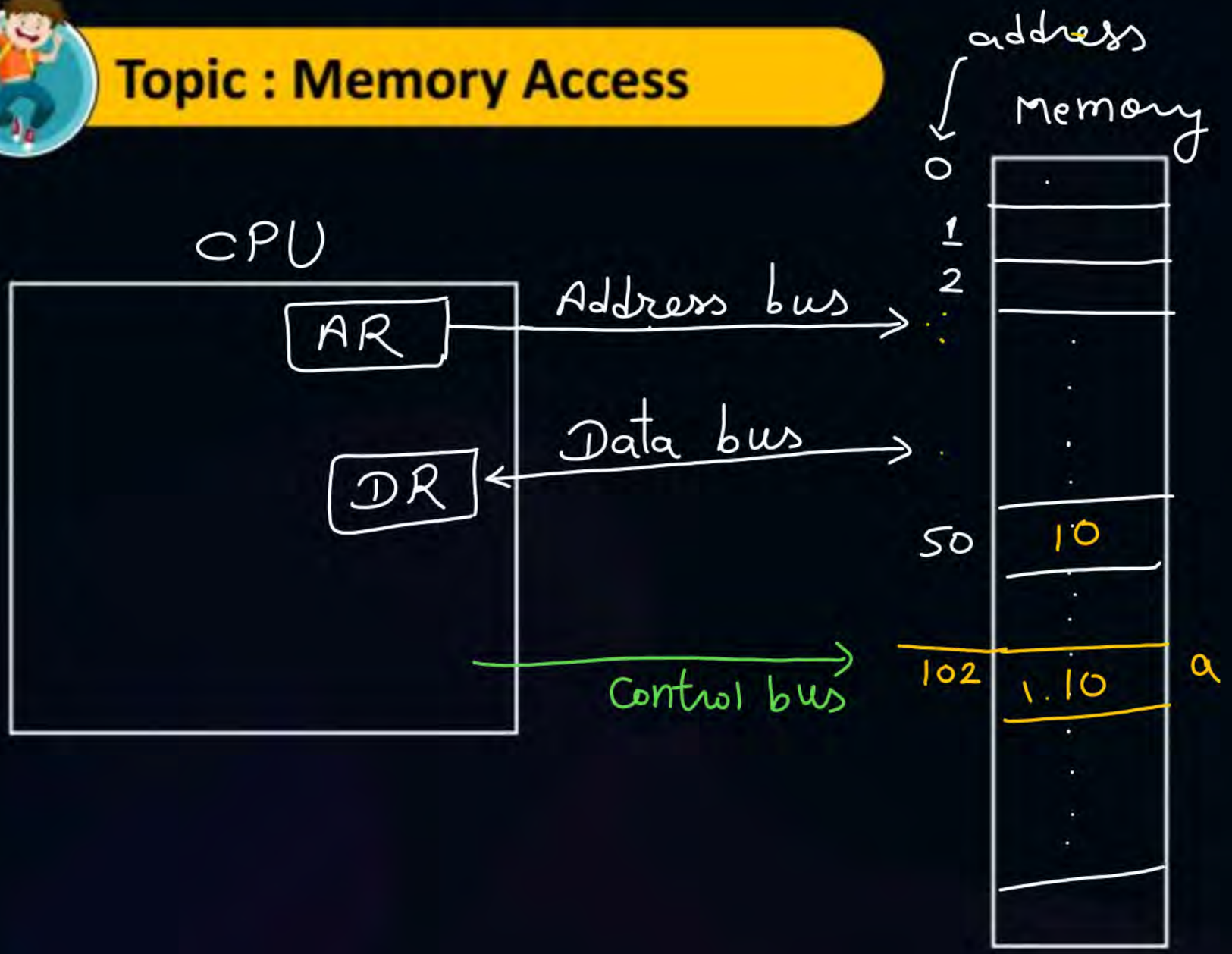
- Used to send address to memory

Topic : Data Register or MDR

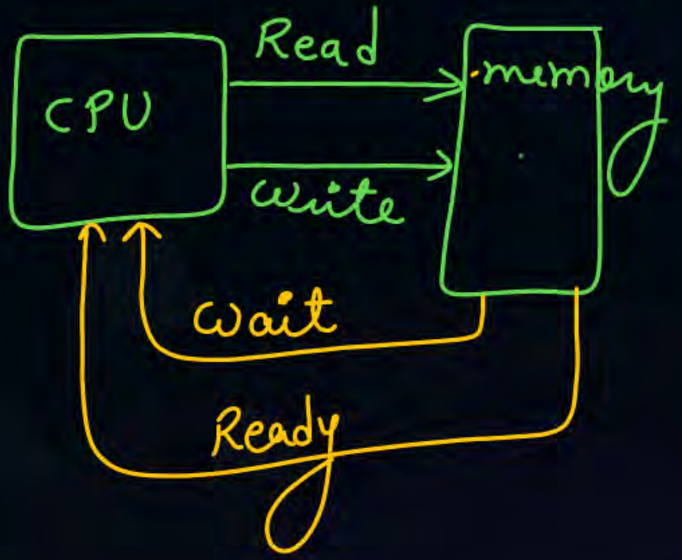
(DR) or MBR

- Used to send data to memory (memory write)
- And to receive data from memory (memory read)

Topic : Memory Access



Control signals b/w CPU & memory



Memory operation :-

1. Read :- Read content from memory and send it to CPU
2. write :- write content in memory which is sent by CPU.

Read:-

1. CPU sends address to memory through address bus
2. CPU sends active Read control signal to memory
3. Memory reads content on given address and sends that content to CPU through data bus.

write :-

1. CPU sends address to memory through address bus
2. CPU sends data to memory " data "
3. CPU sends active write control signal.
4. Memory writes given content to given address.

Control signals from memory to CPU :-

1. **wait :-** when memory is busy in performing read or write then this signal is active to show to CPU that memory can not take any new operation.

2. **Ready :-** when operation in memory is over then memory makes wait signal inactive and ready signal active, to show to CPU that memory is free for another operation.

#Q. CPU can perform 1 operation every 10 ns (nanoseconds)

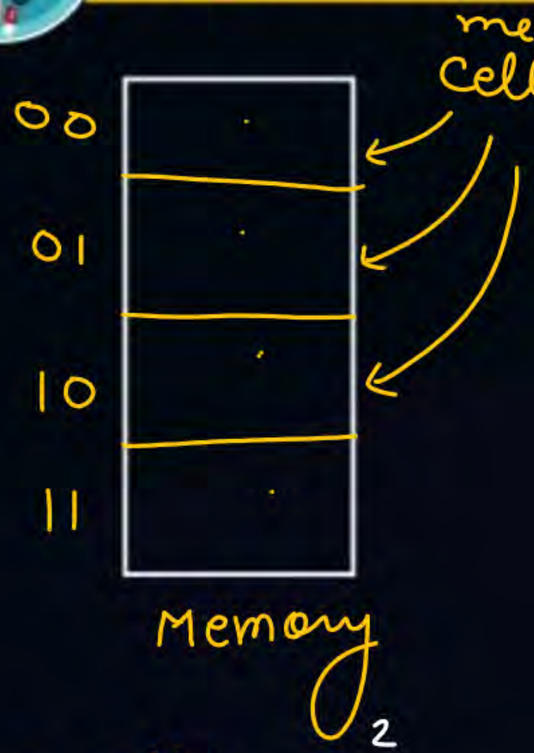
memory —— 11 —— 11 —— 100 ns —11——

No. of memory operations CPU can perform in 500 ns ?

Ans :-

$$\frac{500}{100} = 5 \text{ Ans.}$$

Topic : Memory Addressing



4 cells $= 2^2$

address $\Rightarrow$ 2-bits

8 cells $= 2^3$

$\Rightarrow$ 3-bits

16 cells $= 2^4$

$\Rightarrow$ 4-bits

Topic : Memory Addressing

if no. of cells = n $= 2^a$

address size = $\log_2 n$ bits = a bits

---

address size = x bits

no. of cells = 2^x

#Q No. of cells in a memory $= 2^8$

add. size for memory = ____ bits ?

Ans = 8 bits

#Q No. of cells in memory = 512 $= 2^9$

add. size ____ bits ?

Ans = 9 bits

#Q. No. of address bits to access a memory = 12

No. of cells in memory = ____ ?

Ans :- $2^{12} = 4096$

$K = 2^{10}$

$M = 2^{20}$

$G = 2^{30}$

$2^{12} = 2^{10} * 2^{2} = 4K$

#Q. No. of cells in memory = 32M (32 Mega) $= 2^{5} * 2^{20} = 2^{25}$

add. size = ____ bits

Ans = 25 bits

Each cell has a capacity for data to be stored.

Capacity => 1 byte = 8-bits ⇐ byte addressable memory

or

2 bytes = 16-bits

or

4 bytes = 32-bits

or

8 bytes = 64 bits

or

16 bytes = 128 bits

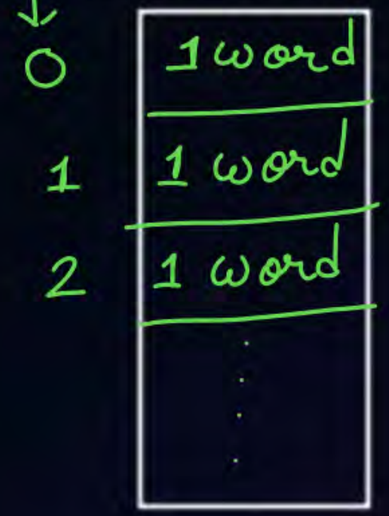
} word addressable memory

Topic : Memory Types

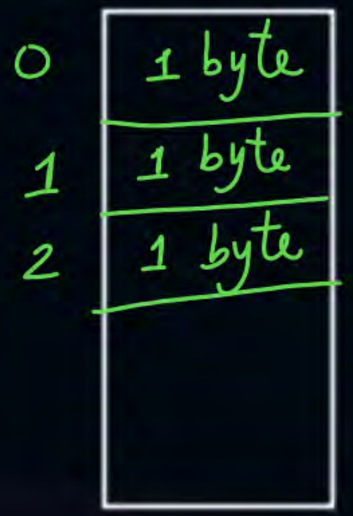
1. Byte Addressable ⇒ each cell of memory has 1 byte capacity
2. Word Addressable ⇒ —11 —— 11—— 1 word 11

addresses

Word Addressable



Byte Addressable



[NAT]

#Q. A CPU has 4 bytes instructions. A program (Instructions I_1 to I_{200}) starts at address 200 (in decimal). Find the address of following instructions:

1. I_1
2. I_5
3. I_{120}

[NAT]

#Q. A CPU has 4 bytes instructions. A program (Instructions I_1 to I_{200}) starts at address 500 (in decimal). What should be the PC value when instruction I_6 will be executing in CPU?

[NAT]

#Q. A CPU has 4 bytes instructions. A program (Instructions I_1 to I_{200}) starts at address 500 (in decimal). What should be the PC value when instruction *i* will be executing in CPU?

Topic : Architecture Type (Based on Size of Input)

Topic : Micro Operation

- The operations executed on values stored in registers
- Symbolic Notation to describe the micro-ops: Register Transfer Language (RTL)

Topic : Micro Operation

- Register Transfer:
- Comma:
- Memory Transfer:

2 mins Summary

t.me/vishvadeepsir

- **Topic** CPU Registers → AR, DR
- **Topic** Memory Addressing
- **Topic** Memory Access ✓
- **Topic** Architecture Type (Based on Size of Input)

Happy Learning

THANK - YOU