CS & IT ENGINEERING

Operating Systems

Memory Management

Lecture No. 1

By- Dr. Khaleel Khan Sir

TOPICS TO BE COVERED

- Abstract View of Memory
- Loading vs Linking
- Address Binding

Q.

Consider the following snapshot of a system running n processes. Process i is holding x_i instances of a resource R, for $1 \le i \le n$. Currently, all instances of R are occupied. Further, for all i, process i has placed a request for an additional y_i instances while holding the x_i instances it already has. There are exactly two processes p and q such that $y_p = y_q = 0$. Which one of the following can serve as a necessary condition to guarantee that the system is not approaching a deadlock?

A. $\min(x_p, x_q) < \max_{k \neq p, q} y_k$

B. $x_p + x_q \geq \min_{k \neq p, q} y_k$ ✓

C. $\max(x_p, x_q) > 1$

D. $\min(x_p, x_q) > 1$

if the request (least) of the Process is Satisfiable n-Processes ($P_1 \ldots P_n$)

R

Pid	Alloc R	Req R	Avail R
P_1	x_1	y_1	0
P_2	x_2	y_2	
P_3	x_3	y_3	
⋮	⋮	⋮	
P_p	x_p	~~0~~	
P_q	x_q	~~0~~	
⋮	⋮	⋮	
P_n	x_n	y_n	

① 't' : $(x_p + x_q) \geq \text{Min}(y_i)$, $i \neq (p, q)$

② : $(x_p + x_q) < \text{Min}(y_k)$, $k \neq (p, q)$

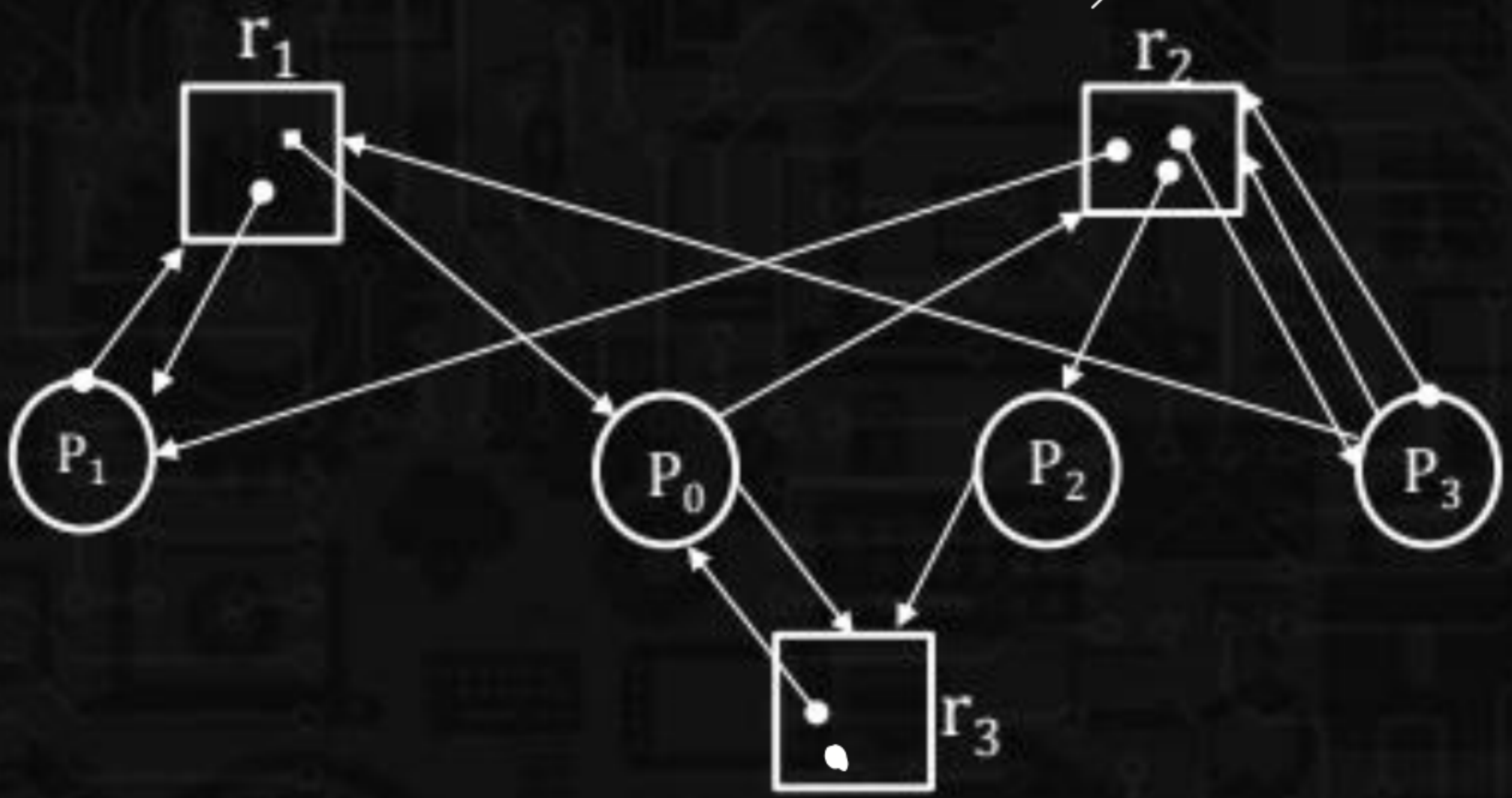
∴ deadlock

③ $(x_p + x_q) > \text{Max}(y_i)$

"Safe"

(detection) (Which Algo)?

Q. Consider the Following Resource Allocation Graph. Find if the System is in (Deadlock State)



(R.A.G $\Rightarrow$ matrix form)

	Alloc r_1 r_2 r_3	Req r_1 r_2 r_3	Avail r_1 r_2 r_3
P_0	1 0 1	0 1 1	0 0 1
P_1	1 1 0	1 0 0	⟨0 1 1⟩
P_2	0 1 0	0 0 1	⟨1 1 2⟩
P_3	0 1 0	1 2 0	⟨2 2 2⟩

⟨P_2; P_0; P_1; P_3⟩

(SAFE)

$(1, 2, 3)$: deadlock can happen

Dining Philosophers

$n = 3$ processes $(P_1 \ldots P_3)$

$R = 4$

$P_i \rightarrow 2(R)$

a) Max (R) to cause deadlock

b) min (R) for deadlock freedom

No deadlock

$P_1 - 2$
$P_2 - 1$
$P_3 - 0$

Deadlock

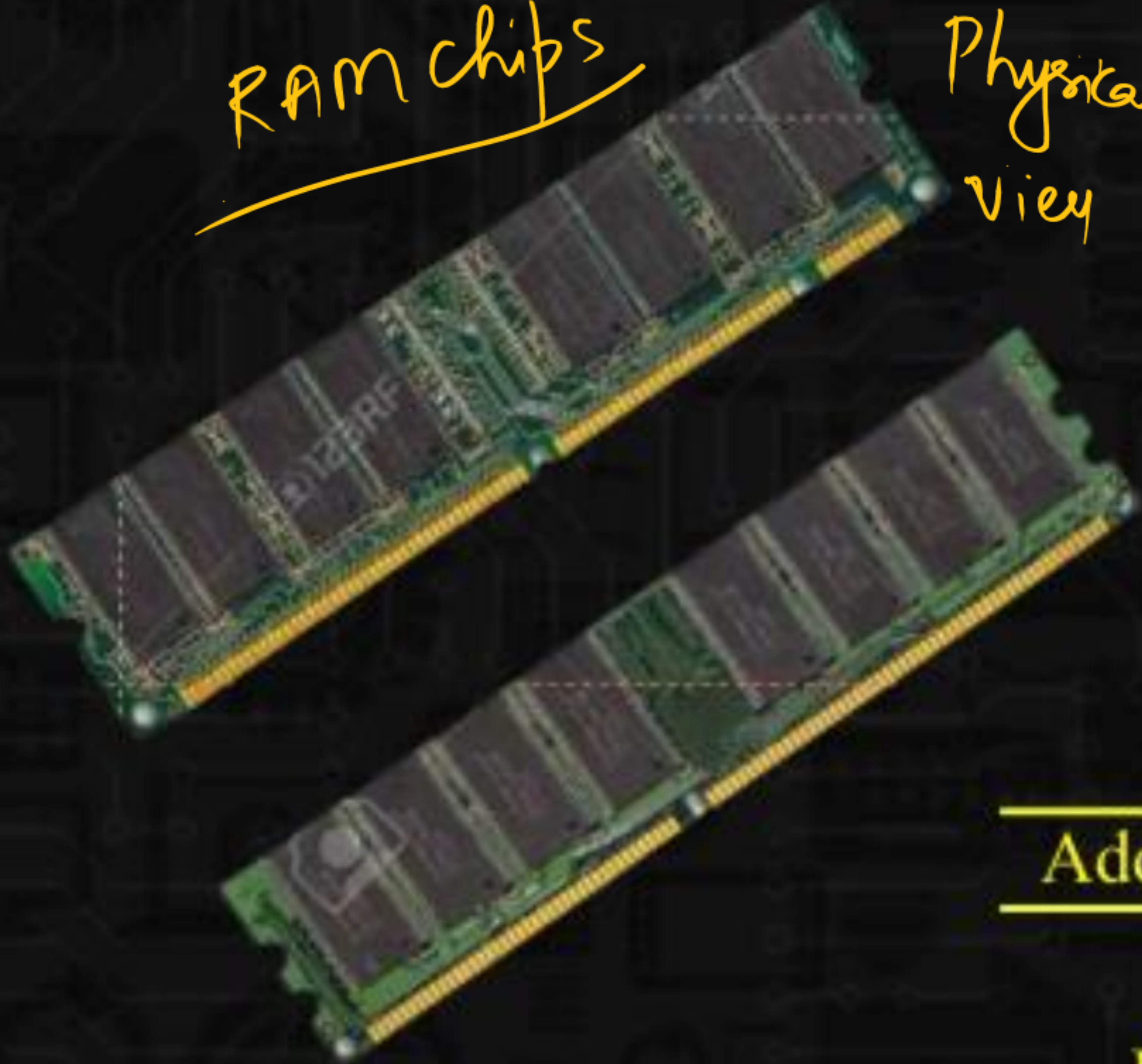
$\boxed{\begin{matrix} P_1 - 1 \\ P_2 - 1 \\ P_3 - 1 \end{matrix}} + 1$

$P_1 - 2$
$P_2 - 2$
$P_3 - 0$

X

Is this deadlock

RAM chips



Physical view

Memory Management

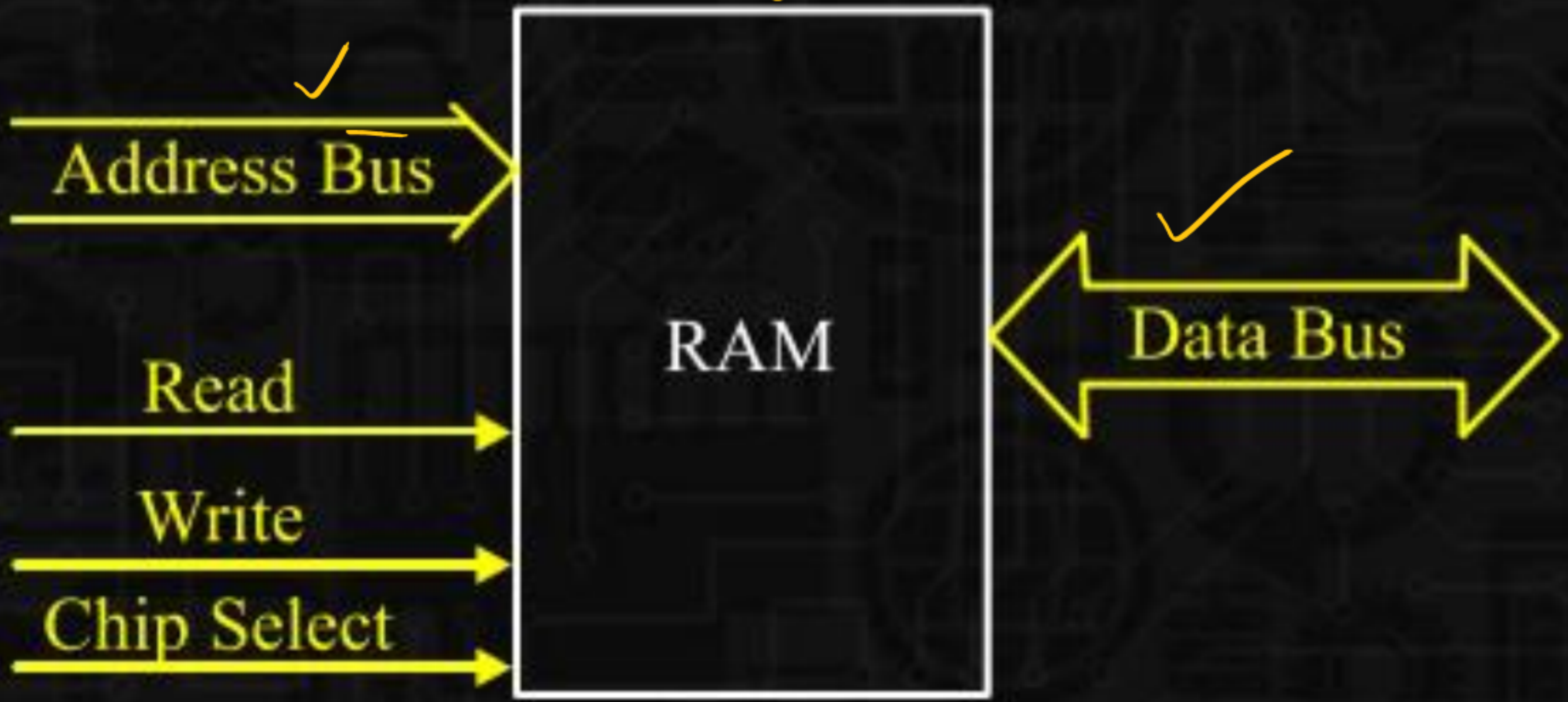
Primary (Main)

RAM + ROM + Cache + {Registers}

R/W Memory

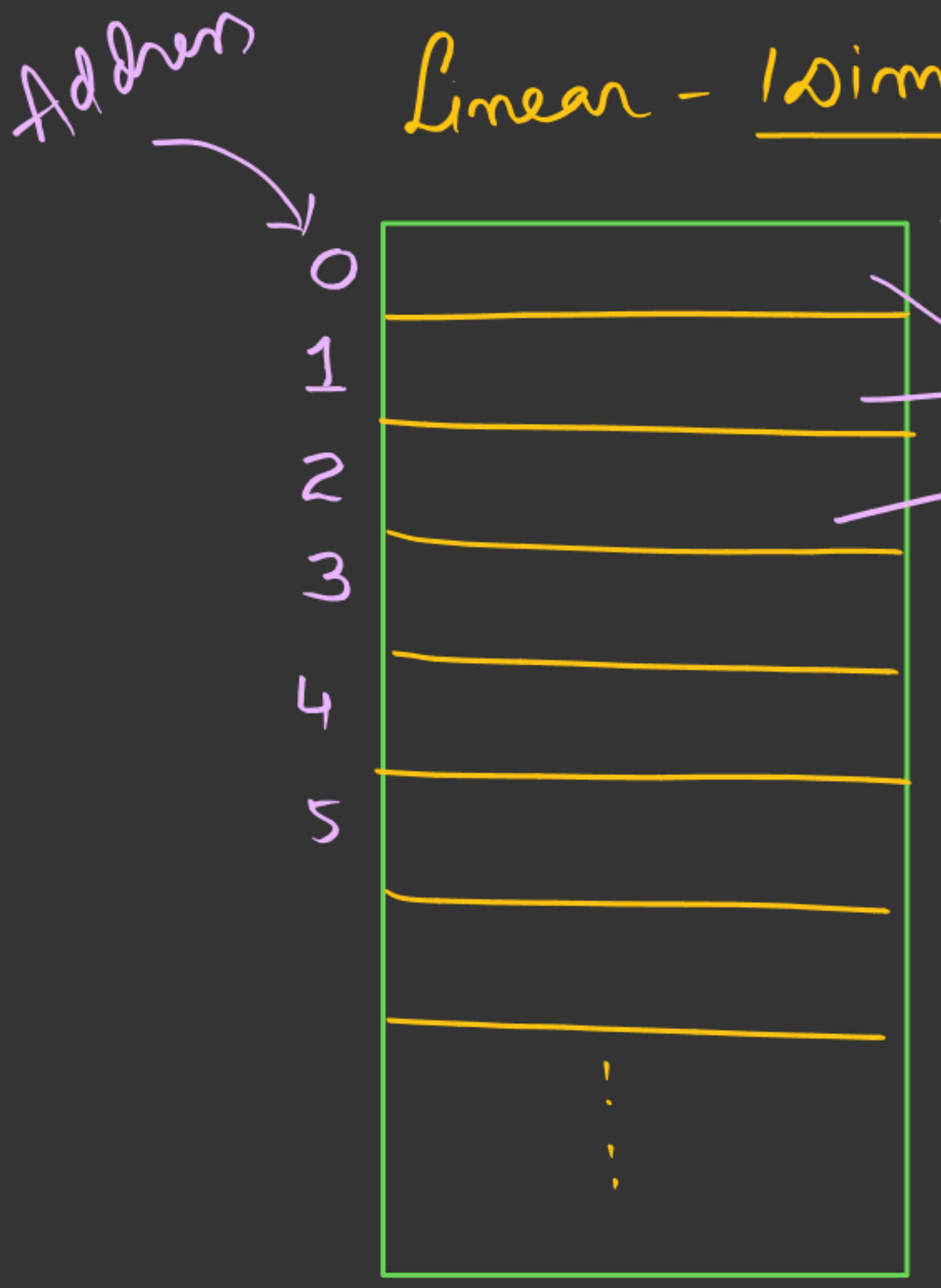
File System

Block Diagram



Linear - 1Dimensional view of Memory (Abstract view)

→ Array of words



Words / Locations / Cells

Total No. of words = 'N'

⇒ Memory Capacity

m-bits

x

Fixed length Address (bits)

word length/width of word

(m) bits / Bytes

word

Instn + Data

1 Byte = 8 bits

1 Nibble = 4 bits

Mem is specified : No. of words × width of word

$(N \times m)$

1. Relation b/w N & n :

→ Binary bits : (0, 1)

→ Using 3 binary bits, How many combinations are possible (2^3)

4 bits $\Rightarrow$ 16 (2^4)

$\boxed{\text{'}n\text{' bits} \Rightarrow 2^n}$

Binary	Decimal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

'8'

n : Size of Address in bits

m : word length / width word

N : Total # of words

$N \propto n$

1) $N = 8$ words

$n = 3$ bits

$2^3 = 8W$

000	0	
001	1	
010	2	
011	3	
100	4	
101	5	
110	6	
111	7	

$2^5 = 32W$

$2^6 = 64W$

$2^7 = 128W$

$2^8 = 256W$

$2^9 = 512W$

$2^{40} \sim 10^{12} = 1T$

$2^{50} \sim 10^{15} = 1P$

$2^{10} = 1024 \approx 10^3 = 1KW$

$2^{20} \sim 10^6 = 1M$

$2^{30} \sim 10^9 = 1G$

$$\boxed{\begin{array}{c} N = 2^n W \\ \hline n = \log_2 N \text{ bits} \end{array}}$$

K = Kilo

B = Bytes

2) $N = 256$

$n = \log_2 256 = \log_2 2^8$

$= 8$ bits

3) $N = 512$

$n = 9$ bits

4) $n = 6$ bits

$N = 2^6 = 64W$

5) $N = 8KB$

$n = 3 + 10 = 13$ bits

6) $n = 15$ bits

$N = 2^{15} = 2^5 \times 2^{10}$

$= 32KB$

7) $N = 16GB$

$n = 4 + 30$

$= 34$ bits

$N = 2^n$

$n = \log_2 N$ bit

1) $N = 500\,GB$

$n = 9 + 30 = 39$ bits

2) $N = 1000\,KB$

$n = 10 + 10 = 20$ bits

3) $N = \log_2 M$ Bytes

$n = \log_2(\log_2 M)$ bits

4) $n = 31$ bits

$N = 2^{31} = 2\,GB$

5) $n = 23$ bits

$N = 2^{23} = 8\,MB$

$= 2^3 \times 2^{20} = 8\,MB$

6) $n = x$ bits

$N = 2^x\,B$

7) $n = \log_2 z$ bits

$N = 2^{\log_2 z}$

$= z^{\log_2 2}$

$\boxed{N = z \text{ Bytes}}$

$n = \log_2 z$ bits

II. $N \propto n \propto m$

1) $n = 13$ bits

$m = 8$ bits ✓

$N = 8K\ W$

$= 8K\ B$

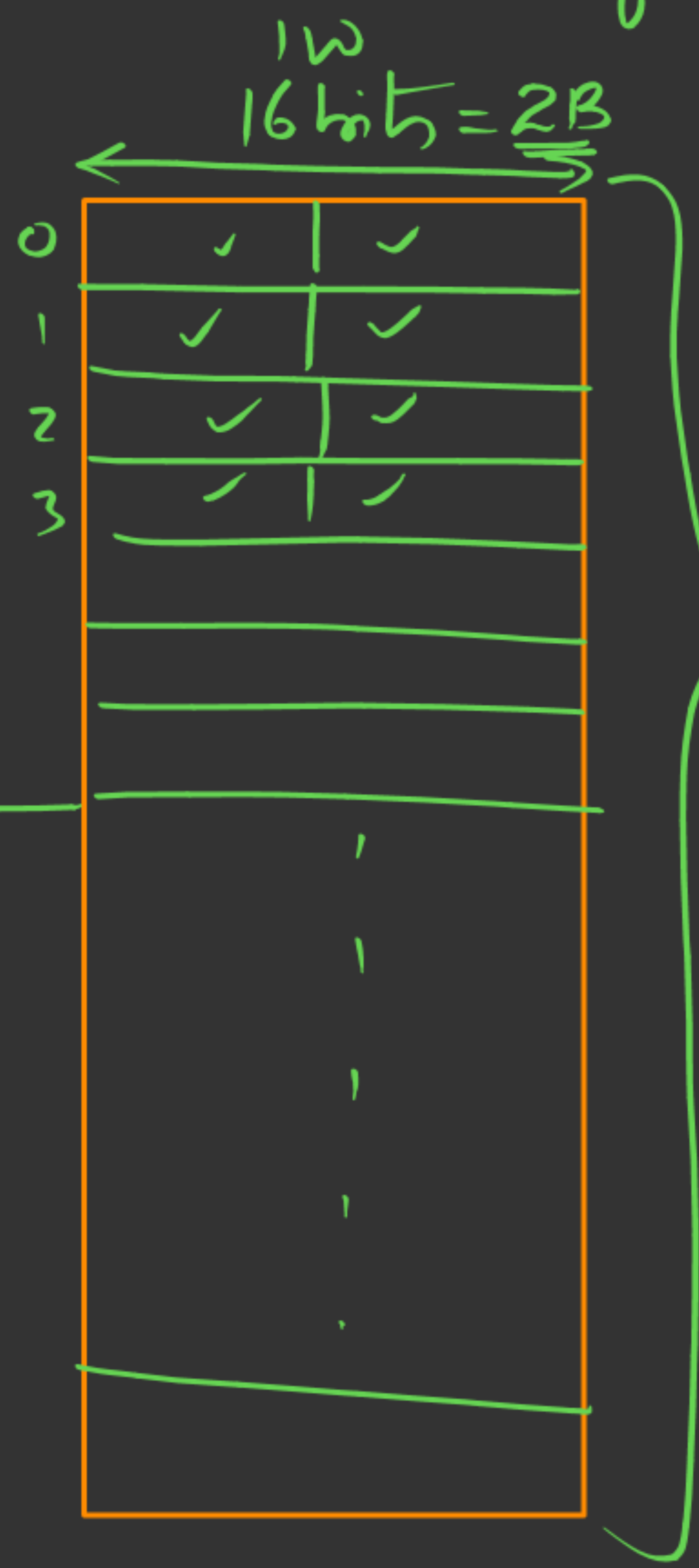
$\boxed{1W = 1B}$

2) $n = 18$ bits ✓ is used to select one word of size 16 bits

$m = 16$ bits

$N = 256KW$ = Word view

$= 512KB$ = Byte view



$2^{18} = 256KW$

$256 \times 2KB$

$512KB$

3) $n = 33$ bits $= 2^{33}$

$m = 64$ bits $\Rightarrow 1W = 8B$

$N = 8G\ W$ ✓

$= 64G\ B$ ✓

3) $N = 64KB$

$m = 4B$

$N = \frac{64K\cancel{B}}{4\cancel{B}} W$

$= 16KW$

$n = 14\text{bits}(W)$

$n = 16\text{bits}(B)$

$= 8B \Rightarrow n = 3\text{bits}$

Memory

B1	B2
B3	B4
B5	B6
B7	B8

$1W = 2B$

$N = \frac{8B}{2B} = 4W \ (2\text{bits})$

$N \sim$ Words / Bytes

$m \sim$ bits/Bytes

$n \sim$ bits

3) $n = 23$ bits

($m = 32$ bits $= 4B$)

$1W = 4B$

$N = 2^{23} = 2^3 \times 2^{20} = 8MW$

$N = 8M \times 4\ B$

$= 32MB$

$N = 2^n W$

$n = \log_2 N$ bits

4) $N = 256MB$

$m = 128B$

$N = 2MW$

$n = 21$ bits (W)

$n = 28$ bits (B)

$$\frac{2\cancel{5}6M\cancel{B}}{1\cancel{2}8\cancel{B}} = 2$$

Last Twist

$N = \underline{8\,G\text{ bits}}$

$\underline{m = 64\text{ bits}}$

$$\frac{8\,G\,\cancel{\text{bits}}}{64\,\cancel{\text{bits}}} = \frac{2^{33}}{2^{6}} = 2^{27}$$

$N = 2^{27} = \underline{128\,MW}$ ✓

$N = \frac{8\,G\,\cancel{\text{bits}}}{\cancel{8}\,\cancel{\text{bits}}} = \underline{1\,GB}$ ✓

Memory

16 bits

$\underline{1W = 2\text{ bits}}$

$$\frac{16\text{ bits}}{2\text{ bits}} = \underline{8W}$$


THANK YOU GW
SOLDIERS !