unacademy

TLB Access and Mapping

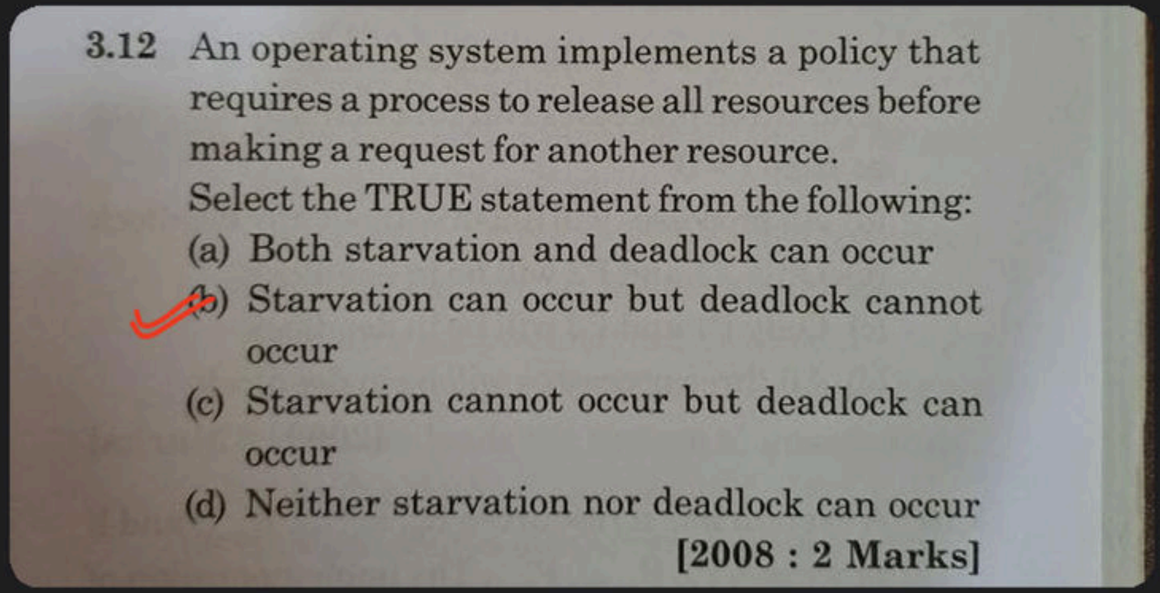
Comprehensive Course on Operating System for GATE - 2024/25

Vishvadeep Gothi • Lesson 29 • May 22, 2023

unacademy

▲ 1 • Asked by Paarth S

Please help me with this doubt



3.12 An operating system implements a policy that requires a process to release all resources before making a request for another resource.
Select the TRUE statement from the following:
(a) Both starvation and deadlock can occur
(b) Starvation can occur but deadlock cannot occur
(c) Starvation cannot occur but deadlock can occur
(d) Neither starvation nor deadlock can occur

[2008 : 2 Marks]

▲ 1 • Asked by Jai

pls explain sir c option

Consider the below statements:

Select one or more answers

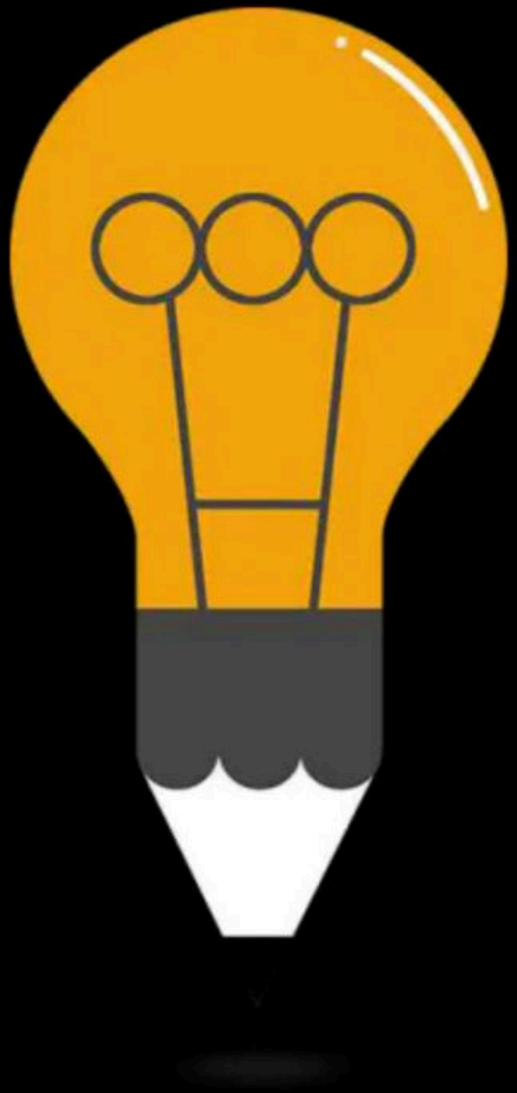
A. User Level Threads requires non-blocking system calls

B. Kernel Level Threads requires non-blocking system calls

C. A full Thread Control Block (TCB) is maintained for each Kernel Level Thread by Kernel

D. Kernel Level Threads are fast and efficient compared to User Level Threads

unacademy



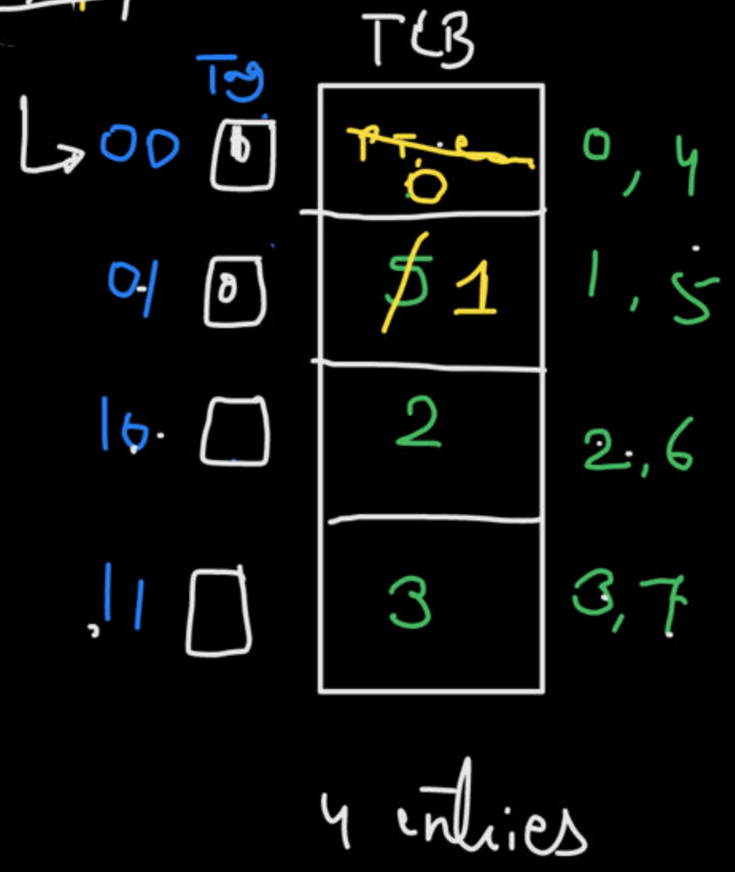
Operating System
TLB Mapping & Segmentation

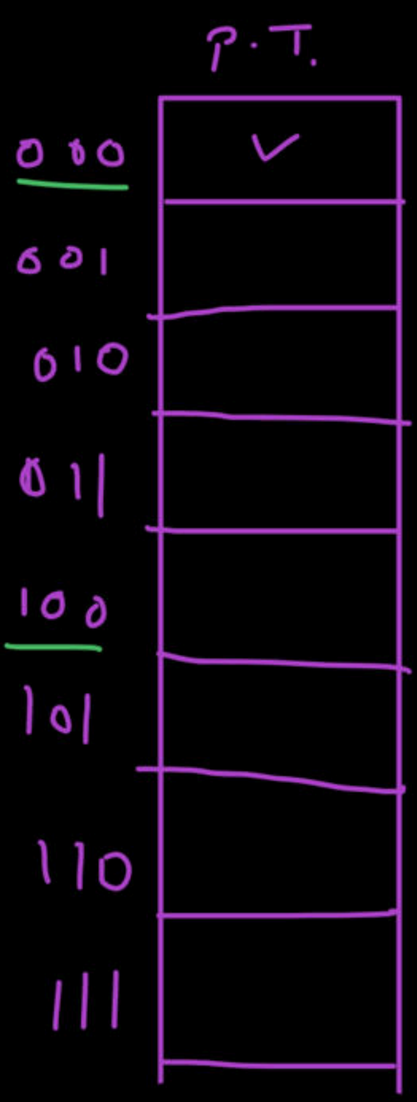
By: Vishvadeep Gothi

TLB Mapping

1. Fully Associative
2. Direct
3. Set-Associative

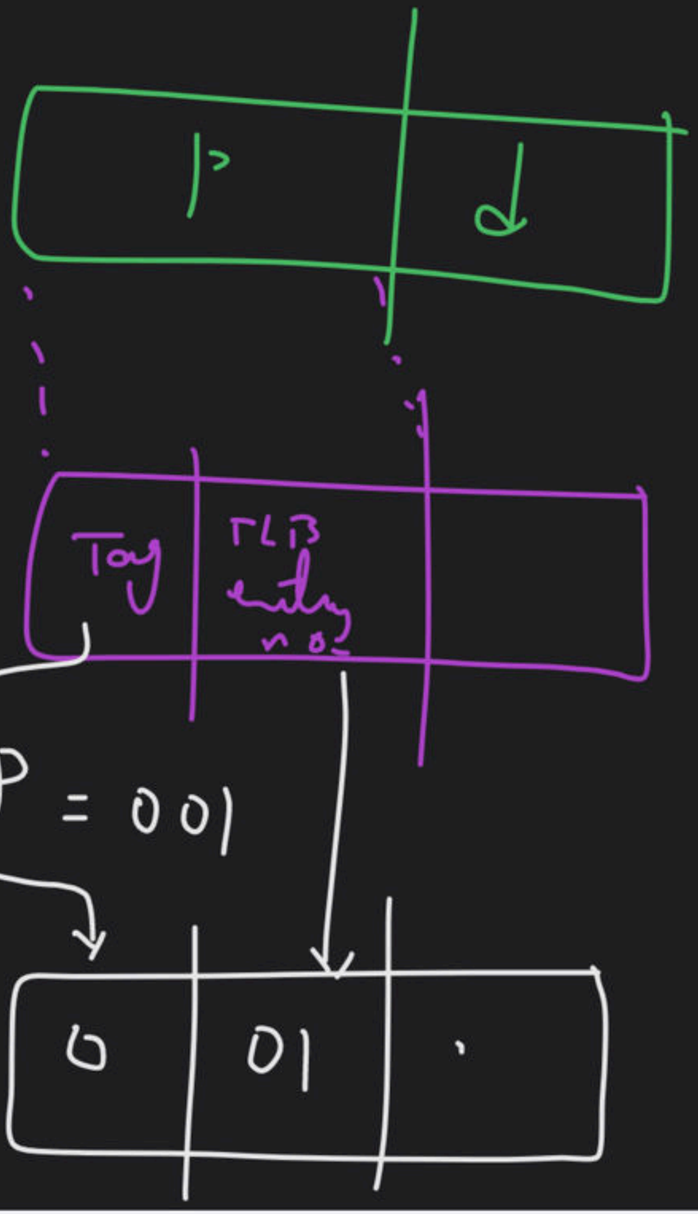
TLB Mapping: Direct





8 page table entries

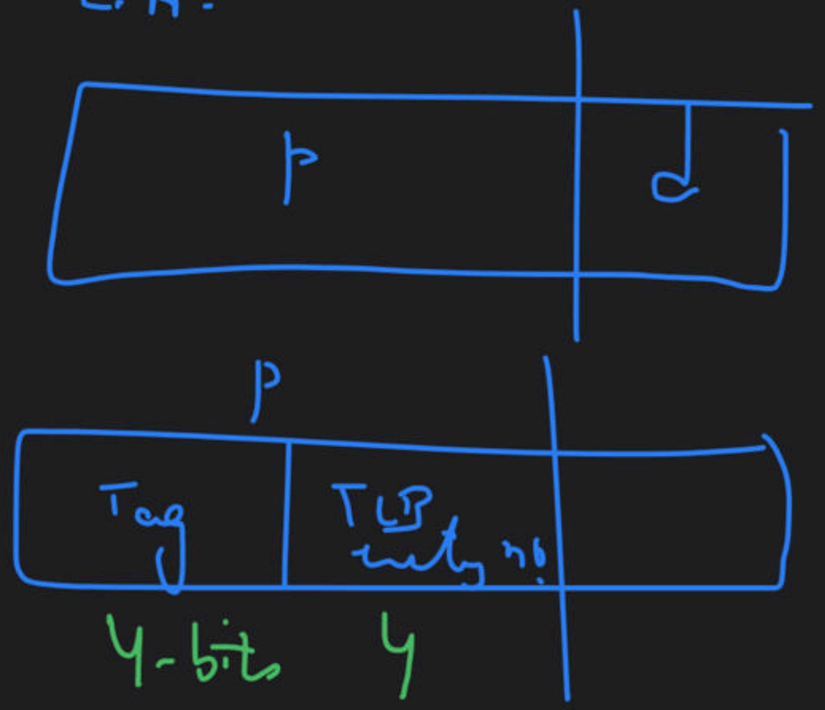
Logical add.



example:- p = 8-bits

TLB => 16 entries => TLB entry no. = 4 bits

L.A.



Ques TLB, 64 entries size

Direct mapping

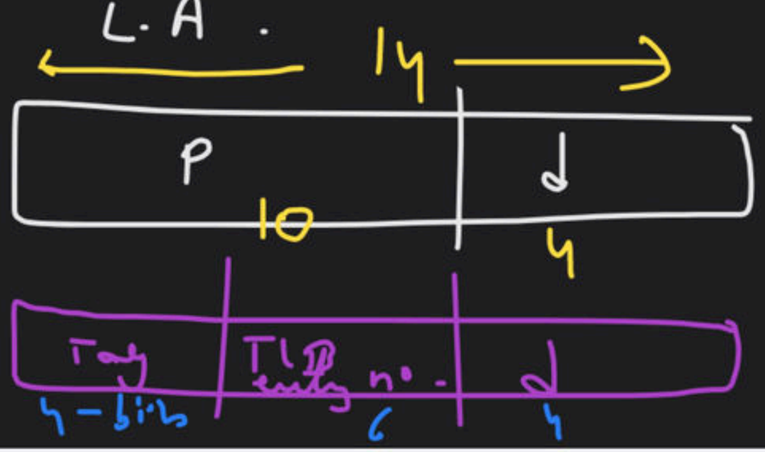
L.A. = 14-bits

Page size = 16 bytes

Tag size = ____ bits

TLB entry no. = 6 bits

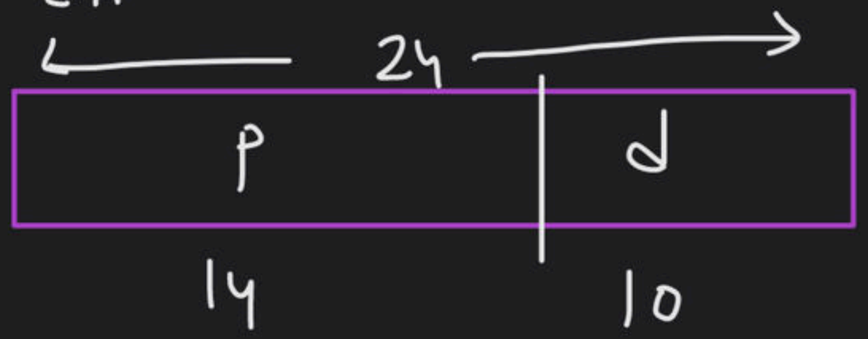
Ans = 4-bits

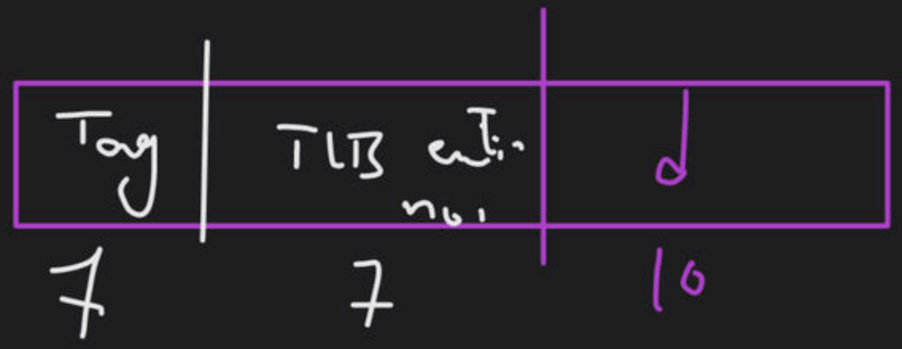


Ques) Consider a direct mapped TLB which can store 128 page table entries. The page size is 1kbytes and logical address space is 16 MB. The tag size in TLB is ____ bits?

Ans = 7

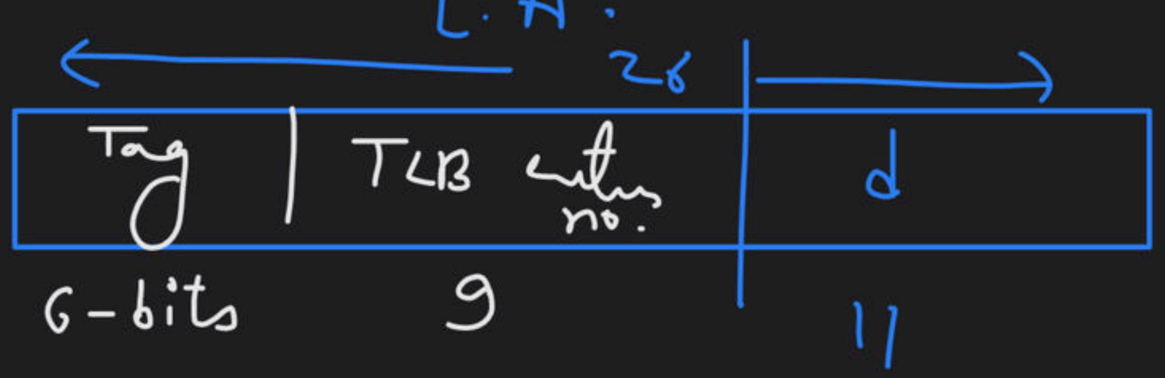
Soln L.A.





Ques) Consider a direct mapped TLB of size 2 kbytes and used to store P.T. entries each of size 4 bytes. Logical addresses are of 26 bits. The Tag size is _____ bits if page size is 2 kbytes.

Soln



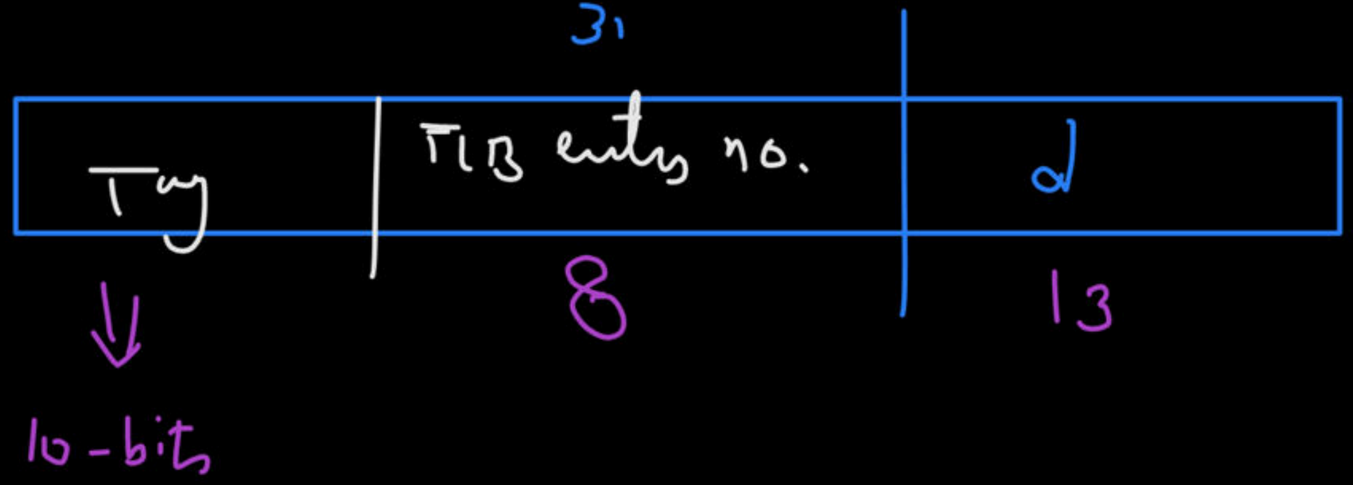
Ans = 6-bits

no. of entries in TLB = $\frac{2KB}{4B} = 2^9$

Question

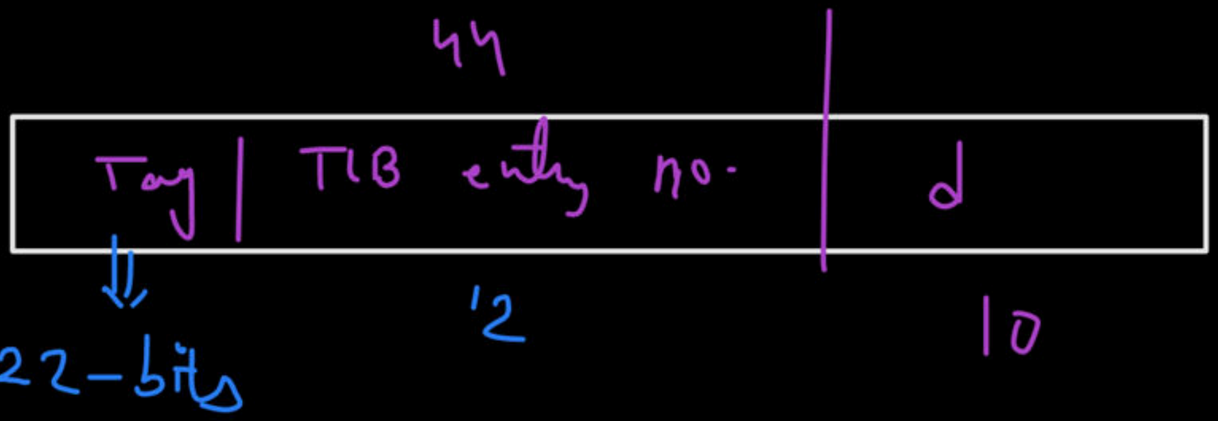
Ans = 10

A computer system implements a 31- bit virtual address, page size of 8 kilobytes, and a 256-entry translation look-aside buffer (TLB) organized as direct mapped. The minimum length of the TLB tag in bits is _____?



Question

A computer system implements a 44- bit virtual address, page size of 1 kilobytes, and a 16KB look-aside buffer (TLB) organized as direct mapped. Each page table entry is of 4bytes. The minimum length of the TLB tag in bits is _____?



$$\text{no. of entries in TLB} = \frac{\text{TLB size}}{\text{1 entry size}}$$

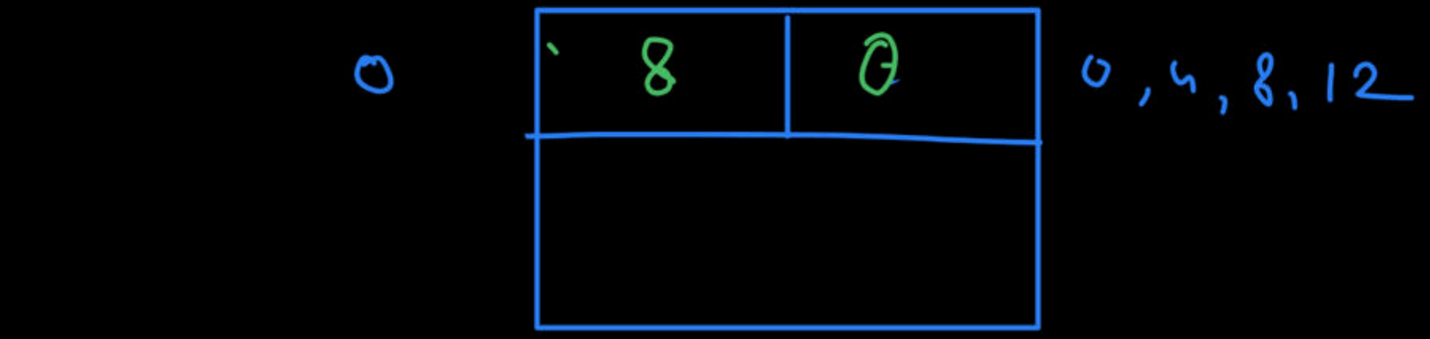
$$= \frac{16\,KB}{4B}$$

$$= 4k$$

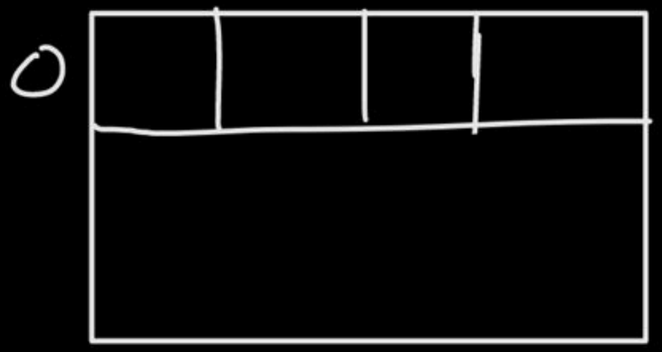
$$= 2^{12}$$

TLB Mapping: Set Associative

2-way set associative mapping ⇒ 1 set 2 entries



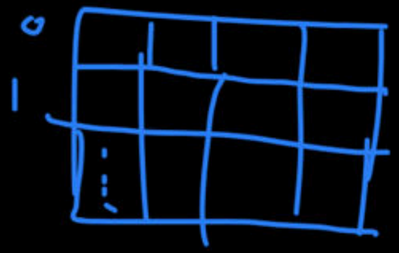
4 way set associative



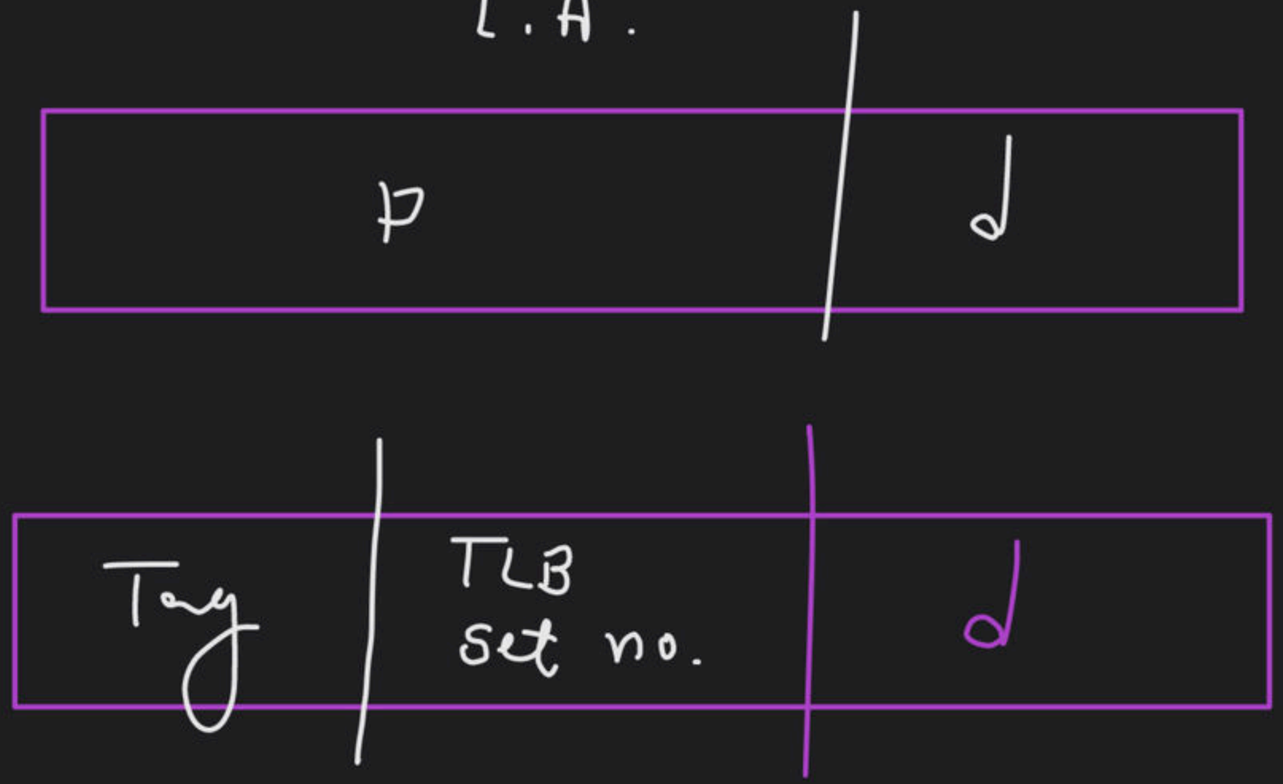
eg:- 4-way set associative

256 entries TLB

no. of sets = $\frac{256}{4}$ = 64



$$\text{no. of sets in } T_{LB} = \frac{\text{no. of entries}}{\text{set associativity}}$$

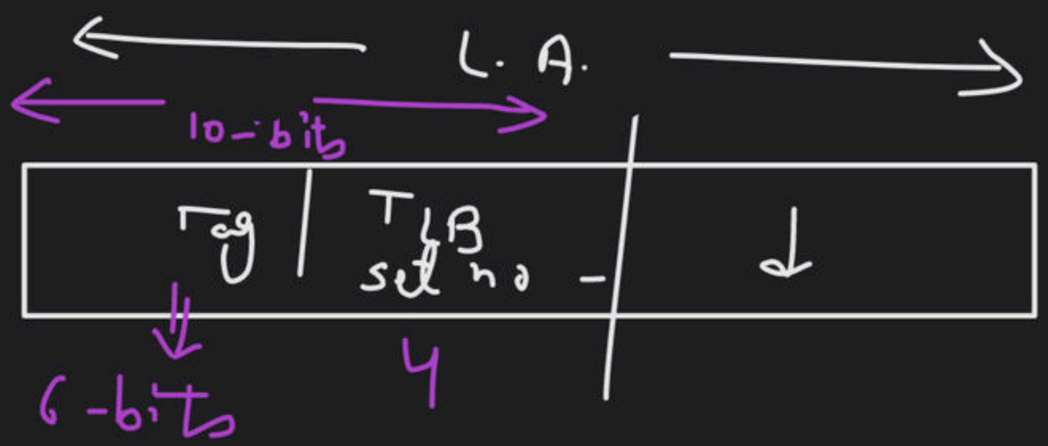


Ans) no. of sets in TLB = 16 ⇒ set no. = 4-bits

p = 10 bits

2-way set associative TLB

Tag bits = ____ bits



Ques) no. of entries in TLB = 1K

4-way set associative TLB

L.A. = 30 bits

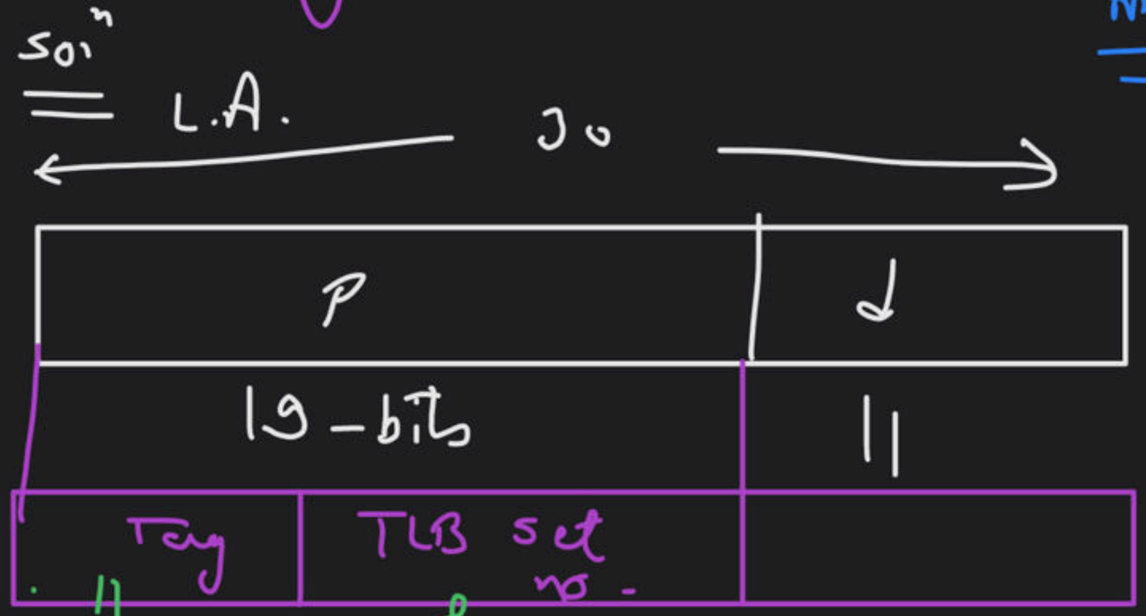
Page size = 2K bytes

Tag = ____ bits => 11 bits

Ans.

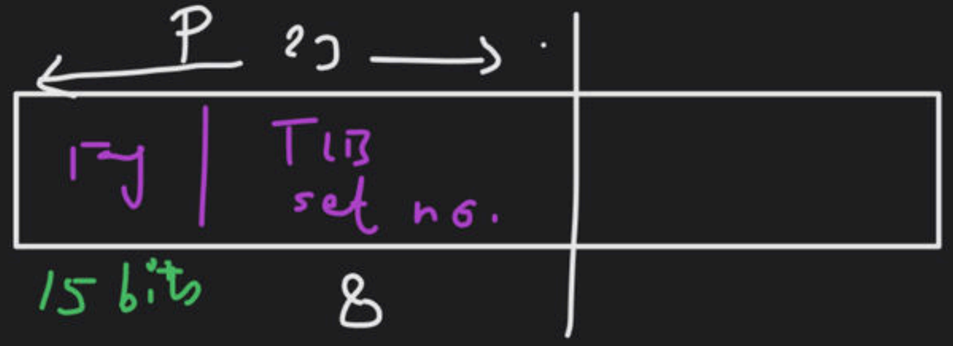
no. of sets in TLB = $\frac{1K}{4}$

$= 2^8$



Ques consider a page table which contains many page table entries. To improve the performance of page table access a TLB is used. The TLB is 4-way set associative and can store 2K bytes. Each page table entry is of size 2 Bytes. If a process has 2^{27} pages. Then Tag size in TLB is ____ bits?

Sol:



no. of entries in TLB $= \frac{2KB}{2B}$

$= 1K$

no. of sets in TLB $= \frac{1K}{4}$

$= 2^8$

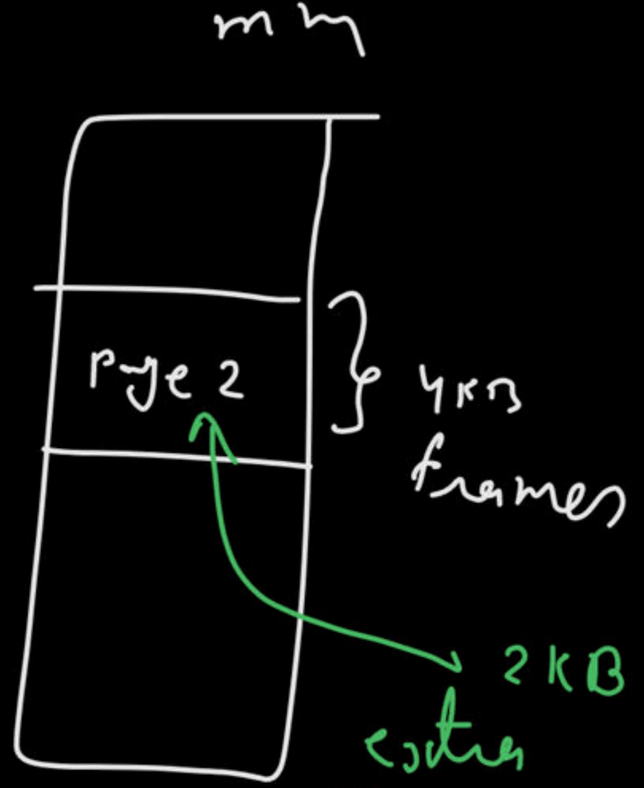
Paging

→ To reduce ⇒ reduce page size.

Paging suffers from internal fragmentation

Assuming, process ⇒ 10KB

Page ⇒ 4K bytes



if TLB, Cache Used

↳ Tags are E.M.A.T. ≠ ?

we will discuss:-

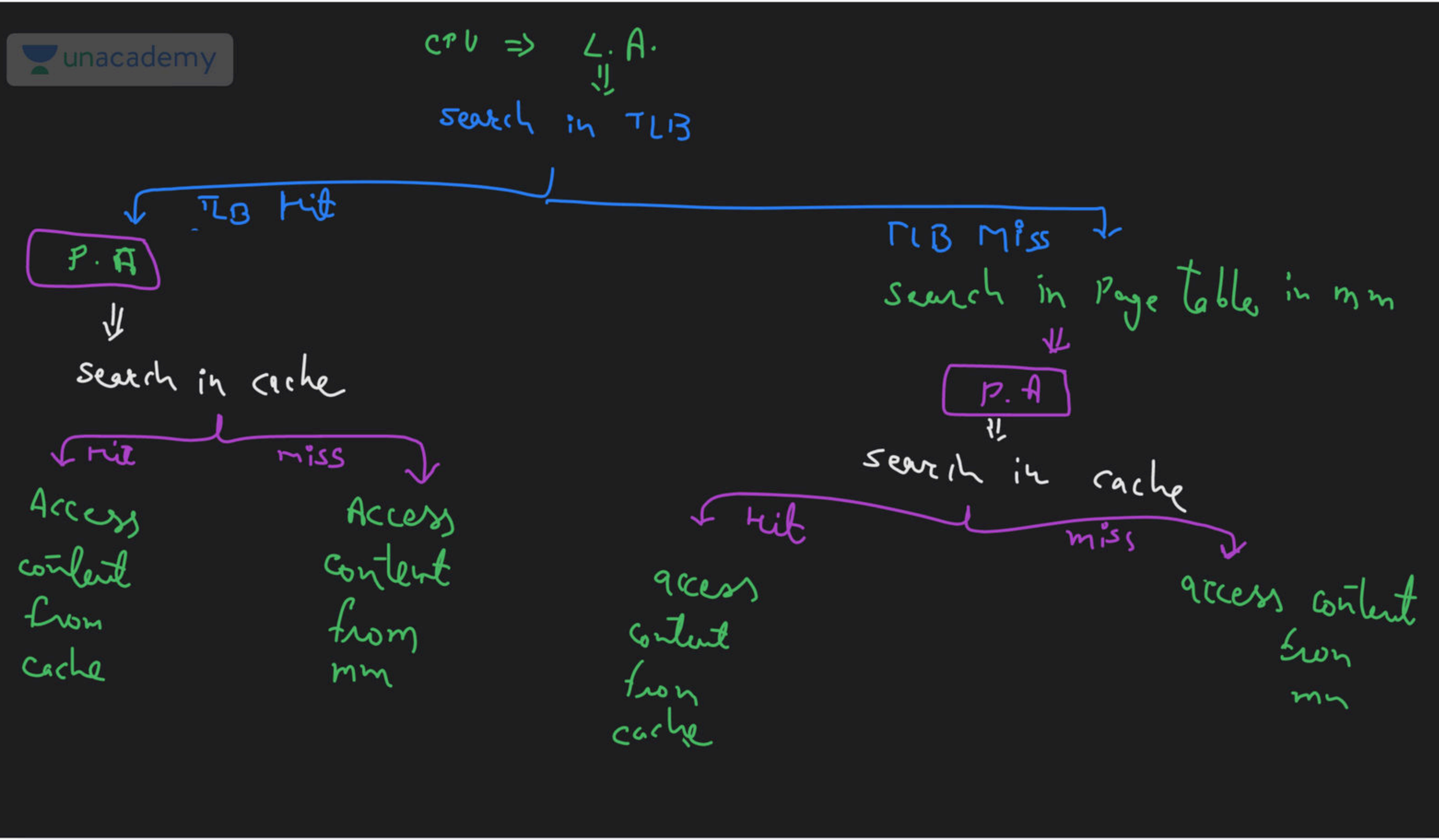
Cache ⇒ physically addressed

↳ Cache memory is accessed only using main memory address.

(we can't search cache with Logical address)

CPU generates ⇒ L.A. ⇒ Translation ⇒ P.A. ⇒ cache search ∴

→ Cache will not store P. T. entries.


CPU ⇒ L.A.
search in TLB
TLB Hit
P.A
search in cache
Hit
miss
Access content from cache
Access content from mm
TLB Miss
search in Page table in mm
P.A
search in cache
Hit
miss
access content from cache
access content from mm

$$E.M.A.T. = H_{TLB} * \left[t_{TLB} + H_{cache} * t_{cm} + (1 - H_{cache}) * (t_{cm} + t_{mm}) \right]$$

$$+ (1 - H_{TLB}) \left[t_{TLB} + t_{mm} + H_{cache} * t_{cm} + (1 - H_{cache})(t_{cm} + t_{mm}) \right]$$

t_{TLB} : To check miss in TLB

t_{mm} : Page table access

Ques) TLB access time = 10 nsec

Cache —11— = 25 nsec

mm —11— = 200 nsec

TLB hit ratio = 80 %

Cache hit ratio = 90 %

E.M.A.T. = ?

Soln

$$EMAT = 0.8\left[10 + 0.9 * 25 + 0.1 * (25 + 200)\right]$$

$$+ 0.2\left[10 + 200 + 0.9 * 25 + 0.1 * (25 + 200)\right]$$

$$= 0.8\left[10 + 45\right] + 0.2\left[10 + 200 + 45\right]$$

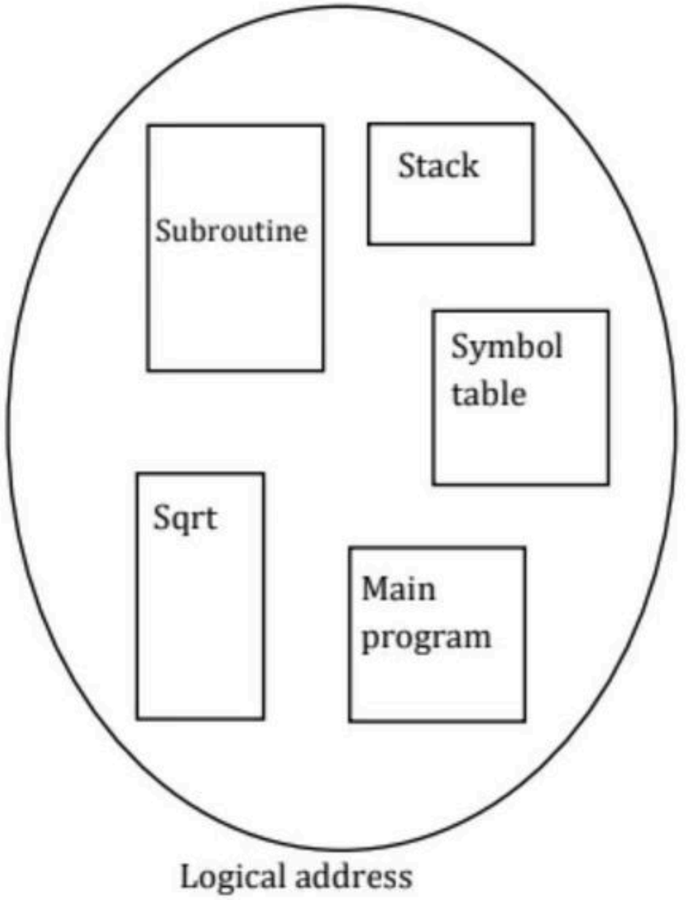
$$= 95 \text{ nsec}$$

Segmentation

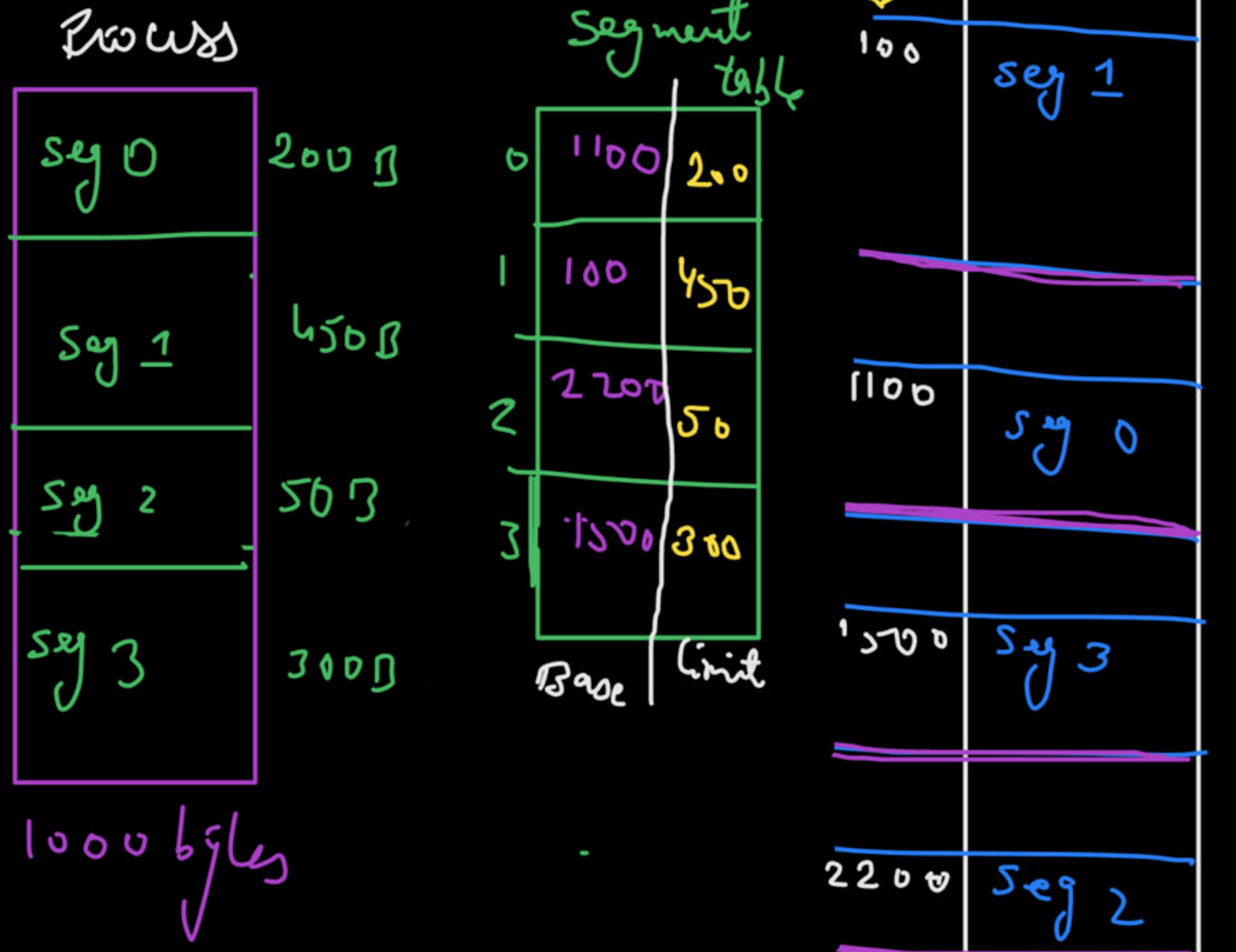
- Divide Process in logically related partitions (Segments) → variable size
- Segments are scattered in physical memory
- → No division in mm.

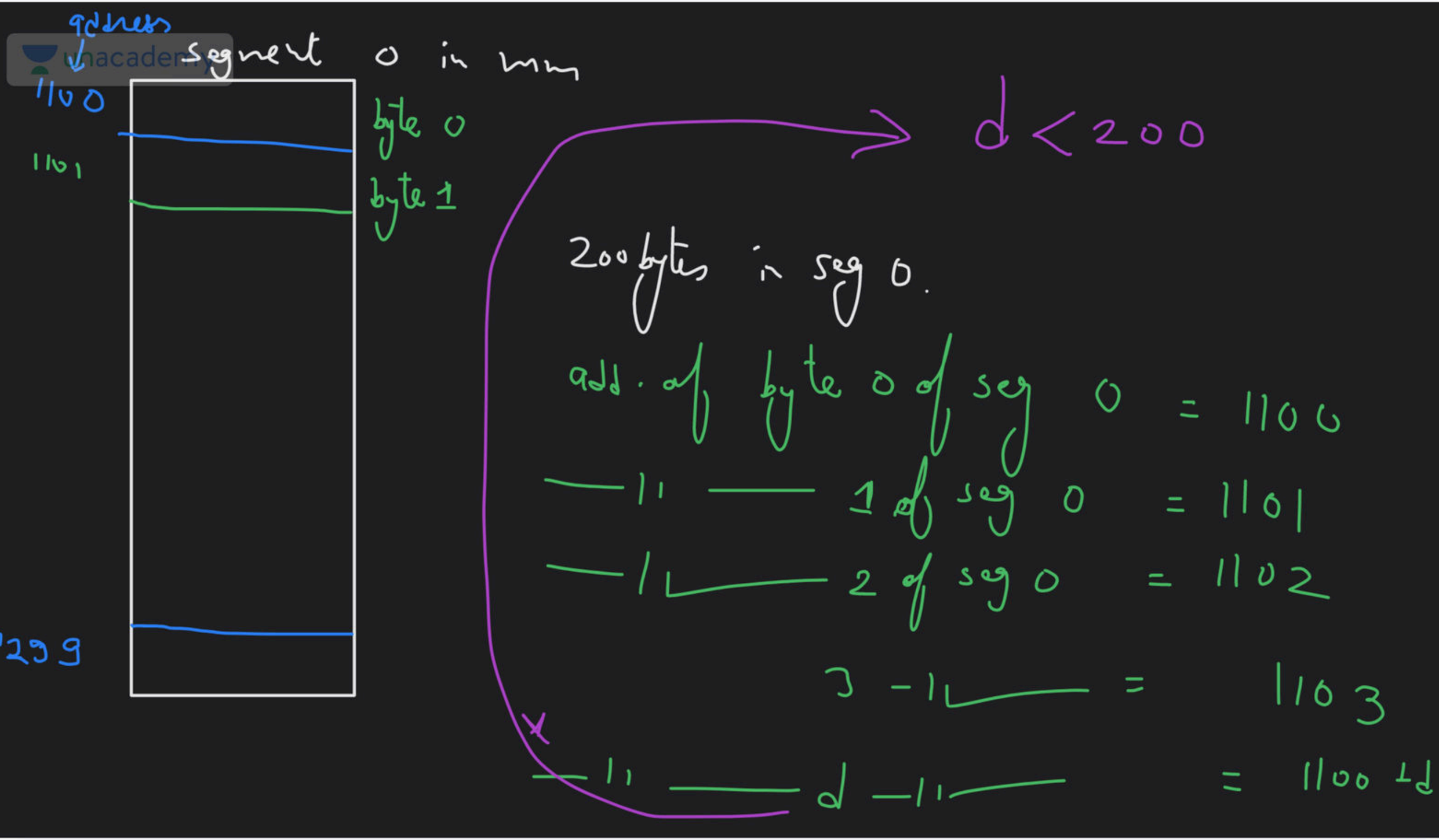
Segmentation

- Divide Process in logically related partitions (Segments)
- Segments are scattered in physical memory



Segmentation



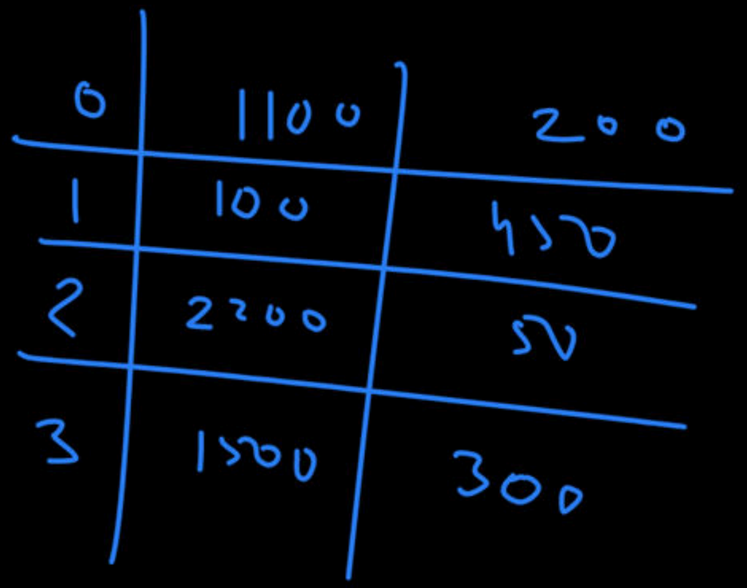

address
segment 0 in mm
1100
1101
byte 0
byte 1
1299
d < 200
200 bytes in seg 0.
add. of byte 0 of seg 0 = 1100
— 1 — 1 of seg 0 = 1101
— 1 — 2 of seg 0 = 1102
3 — 1 — = 1103
— 1 — d — 1 — = 1100 + d

Question

Find physical address for the following requests?

S	d	Physical Address
0	194	1294
2	46	2246
3	298	1798
1	403	503

2 62 seg. fault



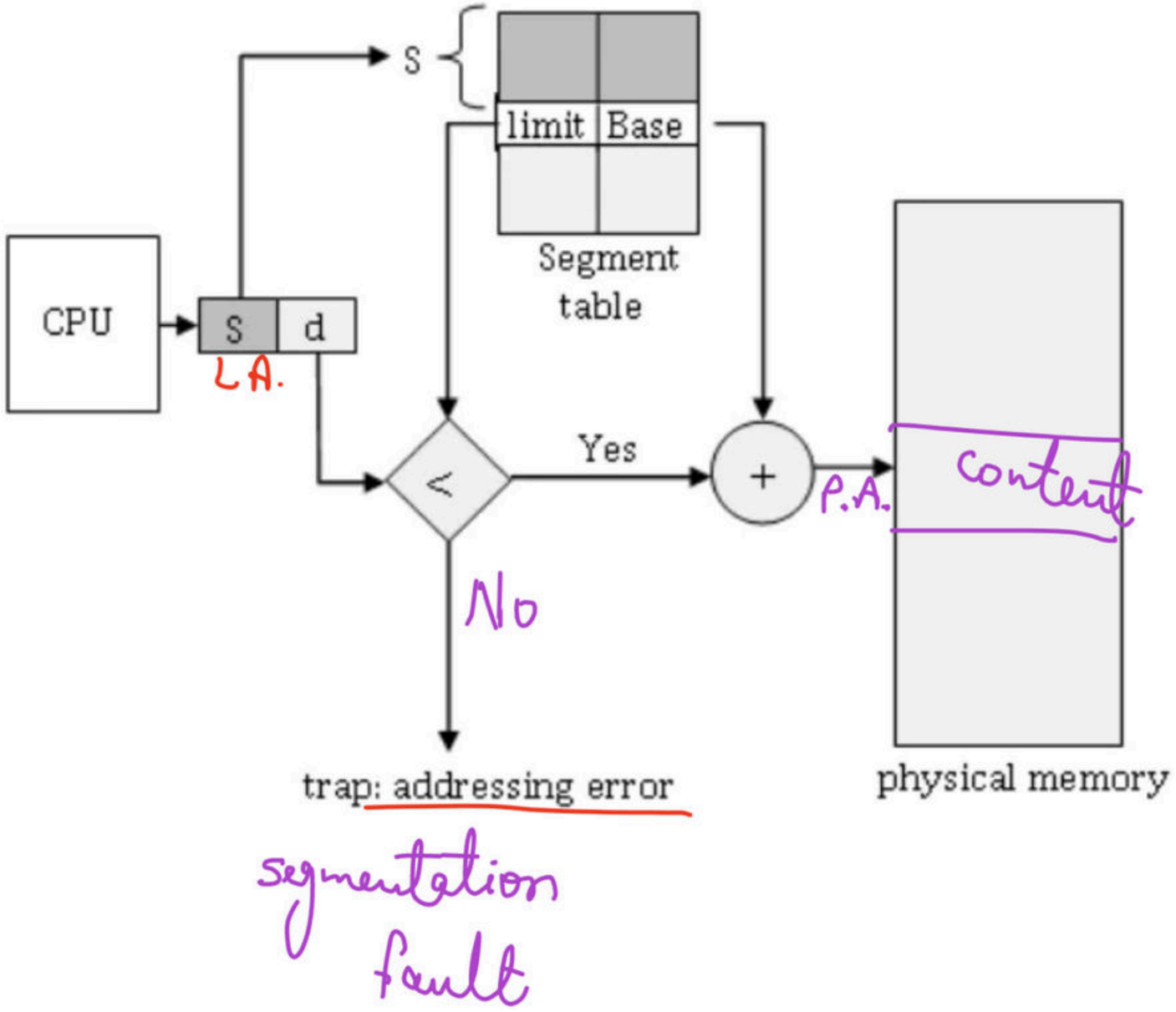
Segmentation

- Size of segment can vary, so along with base, keep limit information also
- Limit defines max number of words within the segment

or

bytes

Segmentation



Segmentation

- Size of segment can vary, so along with base, keep limit information also
- Limit defines max number of words within the segment

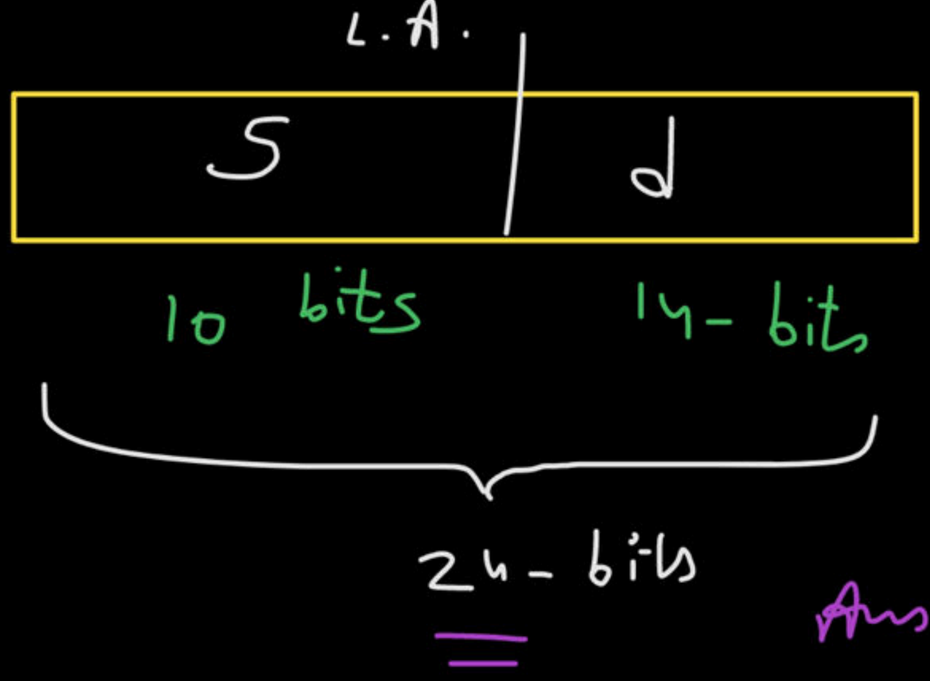
→ no. of bits in offset (d) = $\log_2$ (max allowed seg. size) bits

Question

Maximum segment size = 16KB = 2^{14} B ⇒ d = 14 bits

Number of segments in process = 2^{10}

Logical address = _____ bits ??



Ans.

Segmentation

Segmentation suffers from external fragmentation

Virtual Memory

- Feature of OS
- Enables to run larger process with smaller available memory

Virtual Memory

Demand Paging

Demand Paging:
Bring pages in memory when CPU demands

Page Fault:
When the demanded page is not available in physical memory

How to Ensure the Page hit or fault?

Page Swap Time Saving

Page Swap Time Saving

Effective Memory Access Time

unacademy

Happy Learning.!