CS & IT ENGINEERING

Operating System

Memory Management

Lecture - 4

By- Vishvadeep Gothi sir

Recap of Previous Lecture

Topic Address Translation

Topic Performance of Paging

Topics to be Covered

- Topic: Performance of Paging
- Topic: TLB
- Topic: TLB Mapping

[MCQ]

#Q. Consider a paged memory system where the process size is 16MB and main memory size is 4GB. The page size is 2KB.

2^{24} B

2^{32} B

2^{11} B

A. Number of pages in process? $= 2^{13}$

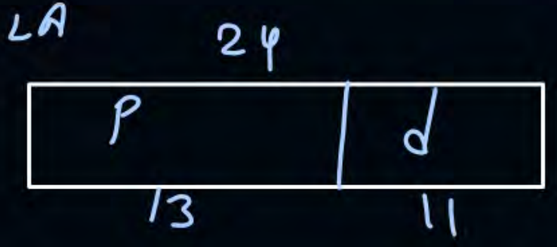
B. Number of frames in main memory? 2^{21}

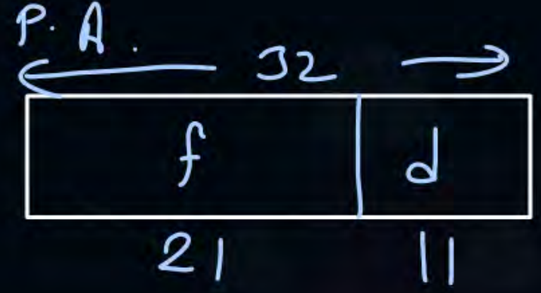
C. Number of bits for page number? 13

D. Number of bits for frames? 21

E. Number of entries in page table? 2^{13}

F. Page table size? $2^{13} * 21$ bits





[MCQ]

L.A.S. 2^{27}

#Q. Consider a paged memory system where the process size is 128MB and main memory size is 2GB. The page size is 1KB.

2^{31}

A Number of pages in process? 2^{17}

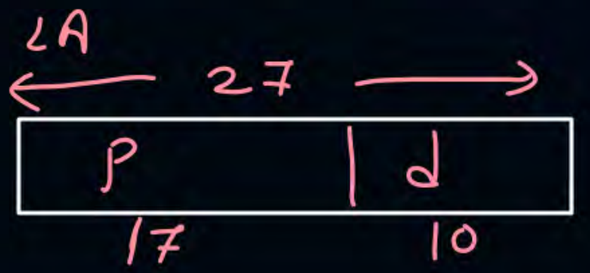
B Number of frames in main memory? 2^{21}

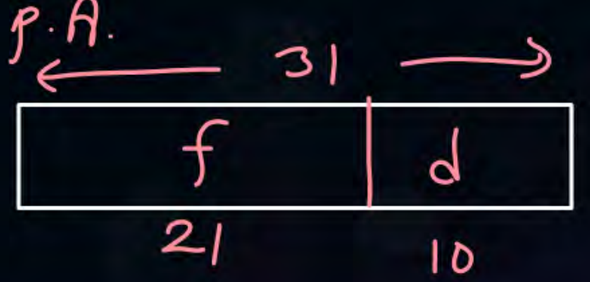
C Number of bits for page number? 17

D Number of bits for frames? 21

E Number of entries in page table? 2^{17}

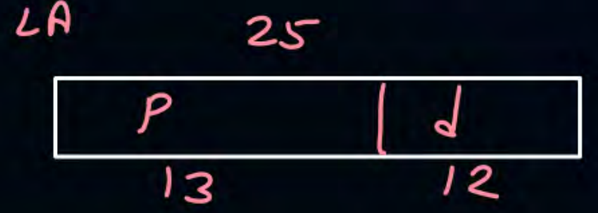
F Page table size? $2^{17} * 21$ bits

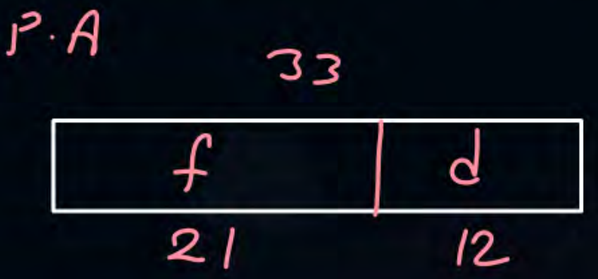




[MCQ]

#Q. Consider a paged memory system where the logical address is 25 bits and physical address is 33 bits. The page size is 4KB.





A. Number of pages in process? 2^{13}

B. Number of frames in main memory? 2^{21}

C. Number of bits for page number? 13

D. Number of bits for frames? 21

E. Number of entries in page table? 2^{13}

F. Page table size? $2^{13} * 21$ bits

[MCQ]

[GATE-2015]

#Q. A computer system implements 8 kilobyte pages and a 32-bit physical address space. Each page table entry contains a valid bit, a dirty bit, three permission bits, and the translation. If the maximum size of the page table of a process is 24 megabytes, the length of the logical address supported by the system is 36 bits?

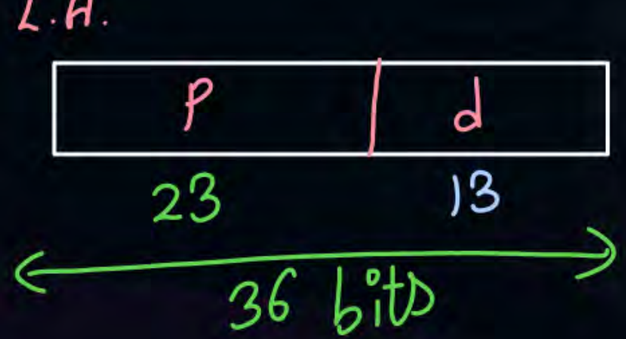
Page size = 8 kbytes = 2^{13} B $\Rightarrow$ d = 13 bits

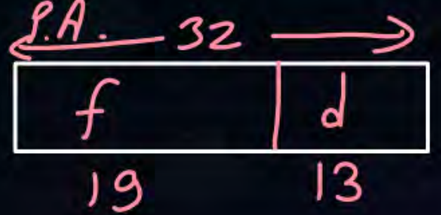
P.A. = 32-bits

Page table entry = 1 valid bit + 1 dirty bit + 3 permission bit + frame no.

Page table size = 24 mega bytes

L.A. = ?





P.T. Size = no. of pages * 1 entry size

24 M bytes = no. of pages * (19 + 1 + 1 + 3) bits

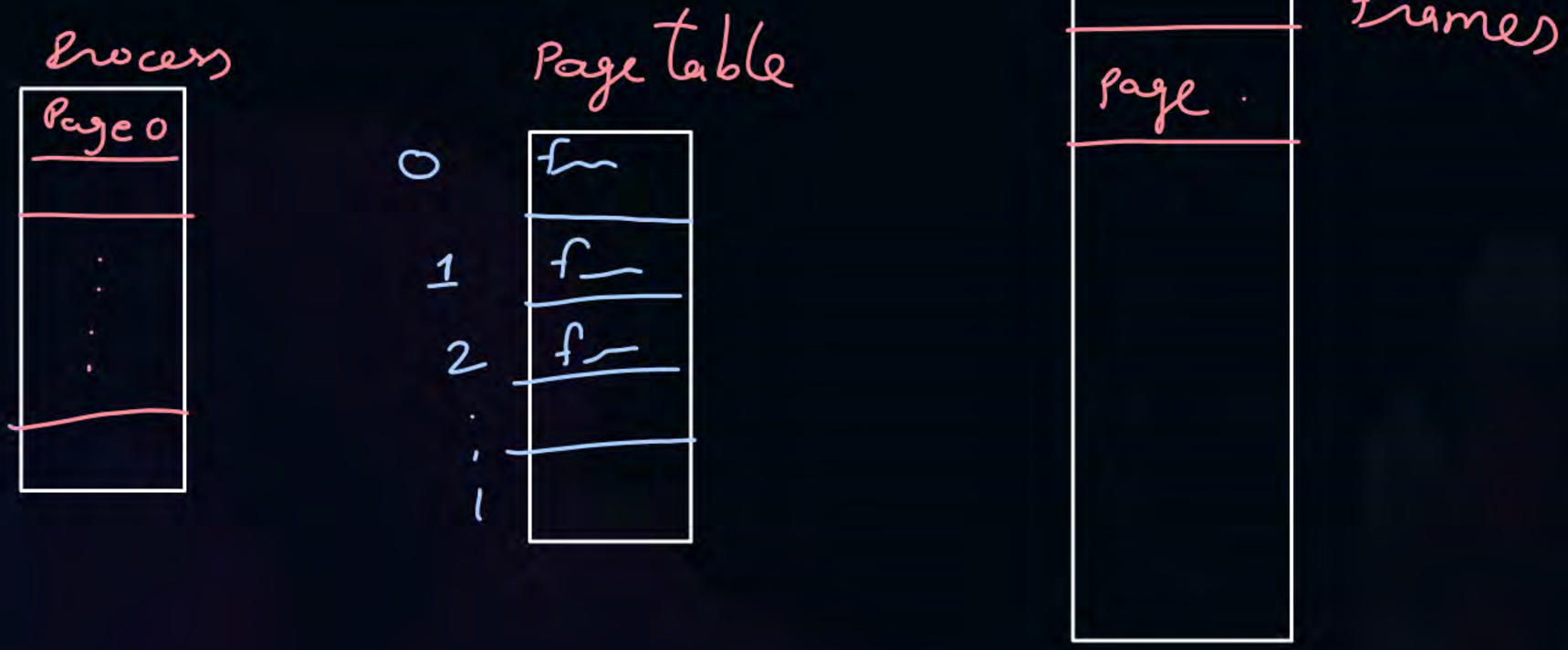
~~24 M * 8 bits~~ = no. of pages * ~~24 bits~~

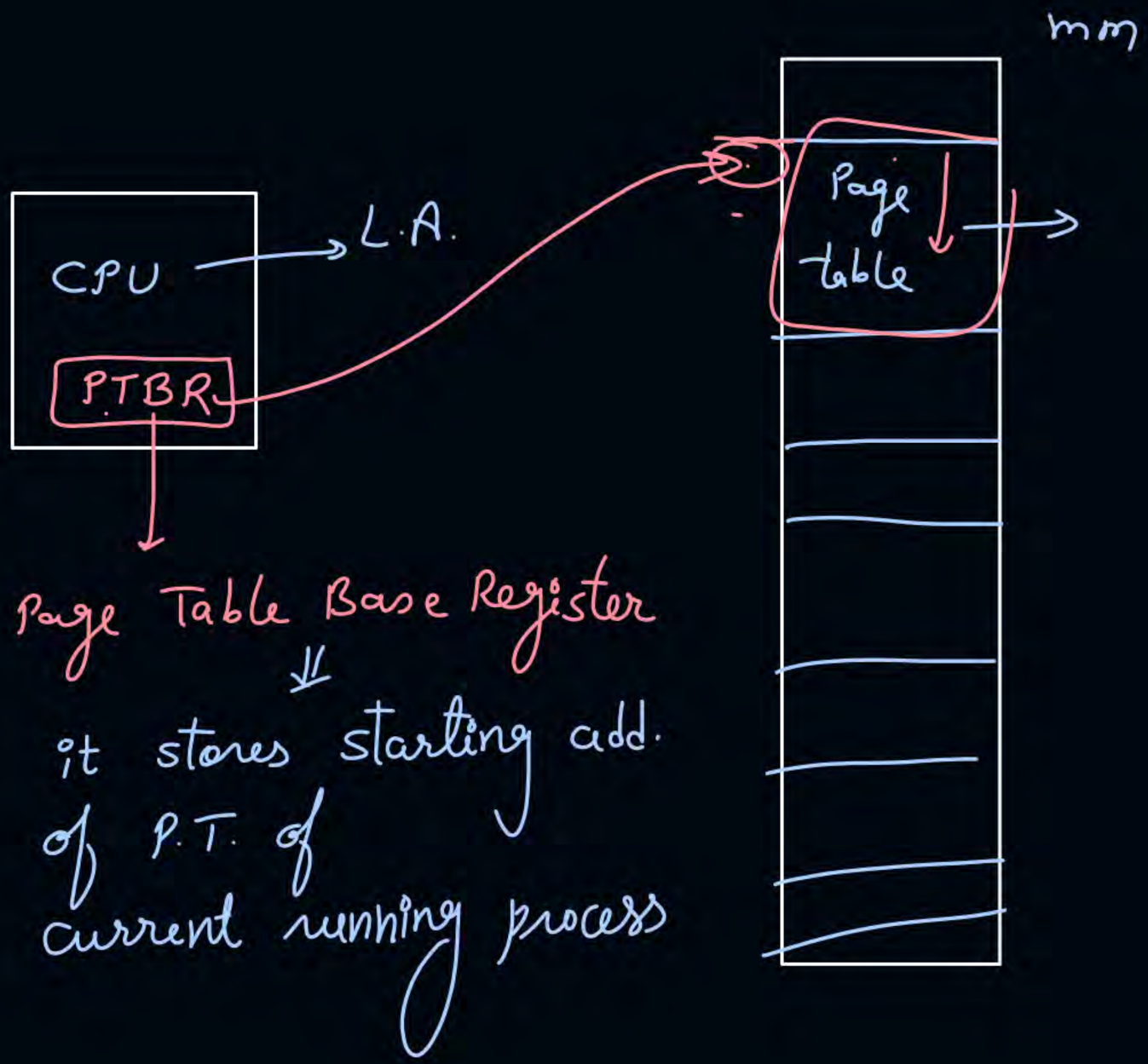
no. of pages = 8 M

= 2^{23} ⇒ page no. = 23 bits

Topic : Paging

Where the Page table Stored? → in main memory only





Page Table Base Register

⇓

it stores starting add. of P.T. of current running process

Topic : Paging

Performance of Paging

Effective mem. access time = 2 * mm access time

(E. M. A. T.)

$= 2 * t_{mm}$

↳ one for page table

one for content

special case :-

if page table is very small and kept in registers

EMAT = t_{mm}

P.T. access time is negligible

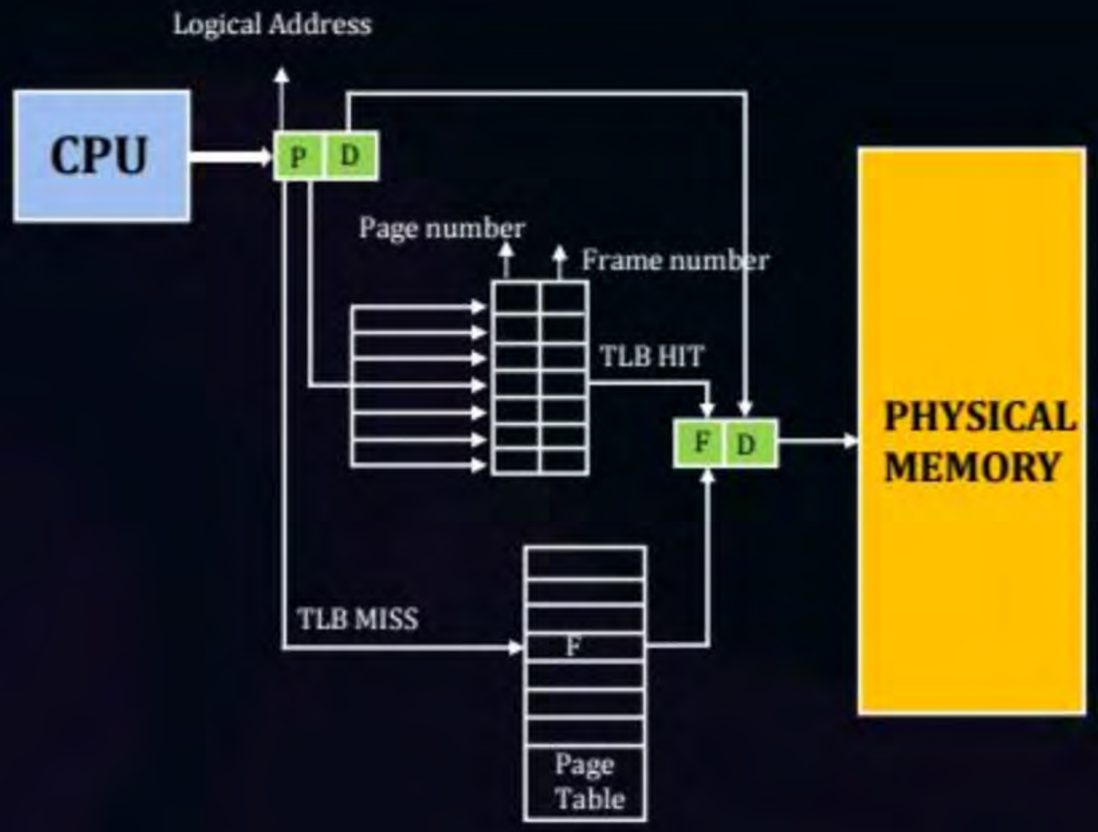
Topic : Paging

TLB (Translation Lookaside Buffer) :- It is a memory hardware, used to store some most frequent and recently referred page table entries.

It is very fast memory, hence it reduces E.M.A.T.

Topic : Paging

TLB (Translation Lookaside Buffer)



CPU generates LA

↓

Search in TLB

Hit

↓

P. A.

↓

Access mm & get the content

⇓

$t_{TLB} + t_{mm}$

Miss

⇓

Access P. T. from mm

⇓

P. A.

↓

Access mm & get the content

⇓

$t_{TLB} + t_{mm} + t_{mm}$

With TLB :- H = TLB hit ratio | $(1-H)$ = TLB miss ratio

$$E.M.A.T. = H * (t_{TLB} + t_{mm}) + (1-H) * (t_{TLB} + t_{mm} + t_{mm})$$

or

$$= t_{TLB} + t_{mm} + (1-H) * t_{mm}$$

$h = 80\% = 0.8$

$t_{TLB} = 50ns$

$t_{mm} = 500ns$

$EMAT \text{ with } TLB = 50 + 500 + 0.2 * 500 = 650\ ns$

$EMAT \text{ without } TLB = 2 * 500$

$= 1000\ nsec$

Topic : Paging

How TLB Stores Entries?

↓

which page table entry is stored in which place in TLB

⇓

for this TLB mapping is done

2 mins Summary

Topic Performance of Paging

Topic TLB

Topic TLB Mapping

Happy Learning

THANK - YOU