PRACTICE PROBLEMS BASED ON SEGMENTED PAGING-

Problem-01:

A certain computer system has the segmented paging architecture for virtual memory. The memory is byte addressable. Both virtual and physical address spaces contain 2^{16} bytes each. The virtual address space is divided into 8 non-overlapping equal size segments. The memory management unit (MMU) has a hardware segment table, each entry of which contains the physical address of the page table for the segment. Page tables are stored in the main memory and consists of 2 byte page table entries. What is the minimum page size in bytes so that the page table for a segment requires at most one page to store it?

Solution-

Given-

- Virtual Address Space = Process size = 2^{16} bytes
- Physical Address Space = Main Memory size = 2^{16} bytes
- Process is divided into 8 equal size segments
- Page table entry size = 2 bytes

Let page size = n bytes.

Now, since page table has to be stored into a single page, so we must have-

Size of page table <= Page size

Size of Each Segment-

Size of each segment

= Process size / Number of segments

= 2^{16} bytes / 8

= 2^{16} bytes / 2^3

= 2^{13} bytes

= 8 KB

Number of Pages Of Each Segment-

Number of pages each segment is divided

= Size of segment / Page size

= 8 KB / n bytes

= (8K / n) pages

Size of Each Page Table-

Size of each page table

= Number of entries in page table x Page table entry size

= Number of pages the segment is divided x 2 bytes

= (8K / n) x 2 bytes

= (16K / n) bytes

Page Size-

Substituting values in the above condition, we get-

(16K / n) bytes <= n bytes

(16K / n) <= n

$n^2 >= 16K$

$n^2 >= 2^{14}$

$n >= 2^7$

Thus, minimum page size possible = 2^7 bytes = 128 bytes.

Problem-02:

Considering problem-01, give the division of virtual address.

Solution-

Number of Bits Required For Segment Number-

Number of segments the process is divided

= 8

= 2^3

Thus, Number of bits required to identify a particular segment in segment table = 3 bits.

Number of Bits Required For Page Number-

Number of pages a segment is divided

= Segment size / Page size

= 8KB / 128 bytes

= 2^{13} bytes / 2^7 bytes

= 2^6 pages

Thus, Number of bits required to identify a particular page in table = 6 bits.

Number of Bits Required For Page Offset-

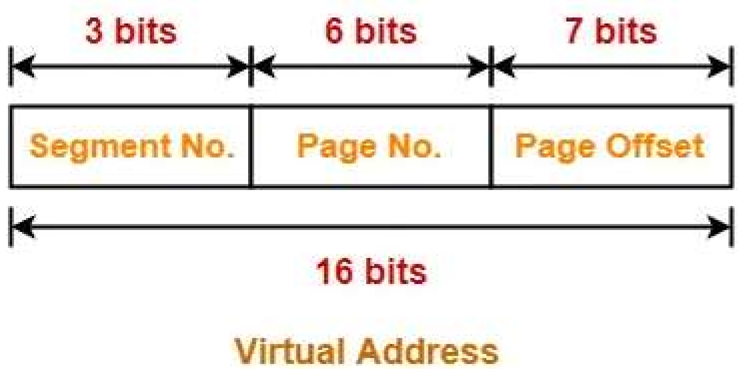
Page size

= 128 bytes

= 2^7 bytes

Thus, Number of bits required for page offset = 7 bits.

Thus, virtual address is divided as-



Problem-03:

A certain computer system has the segmented paging architecture for virtual memory. The memory is byte addressable. Both virtual and physical address spaces contain 2^{16} bytes each. The virtual address space is divided into 8 non-overlapping equal size segments. The memory management unit (MMU) has a hardware segment table, each entry of which contains the physical address of the page table for the segment. Page tables are stored in the main memory and consists of 2 byte page table entries.. Assume that each page table entry contains (besides other information) 1 valid bit, 3 bits for page protection and 1 dirty bit. How many bits are available in page table entry for storing the aging information for the page? Assume that page size is 512 bytes.

Solution-

Given-

- Virtual Address Space = Process size = 2^{16} bytes
- Physical Address Space = Main Memory size = 2^{16} bytes
- Process is divided into 8 equal size segments
- Page table entry size = 2 bytes = 16 bits
- Page table entry besides other information contains 1 valid bit, 3 protection bits, 1 dirty bit
- Page size = 512 bytes

Number of Frames in Main Memory-

Number of frames in main memory

= Size of main memory / Page size

= 2^{16} bytes / 512 bytes

= 2^{16} bytes / 2^9 bytes

= 2^7 frames

Thus, Number of bits required for frame identification in page table entry = 7 bits

Number Of Bits Available For Storing Aging Information-

Number of bits available for storing aging information

= Number of bits in page table entry – (Number of bits required for frame identification + 1 valid bit + 3 protection bits + 1 dirty bit)

= 16 bits – (7 + 1 + 3 + 1) bits

= 16 bits – 12 bits

= 4 bits