MODULE 4 : INTEL 80386DX PROCESSOR

- 4.1 Architecture of 80386 microprocessor Control
- 4.2 80386 registers-General purpose Registers, EFLAGS and registers
- 4.3 Real mode, Protected mode, virtual 8086 mode
- 4.4 80386 memory management in Protected Mode - Descriptors and selectors, descriptor tables, the memory paging mechanism

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Features of 80386 :

- 80386 has a "32 bit" address bus
  - This means it can access a total of 2^32= 4GB of physical memory. .
  - The memory has an address range of 0000 0000H… FFFF FFFFH.
  - Though the total address bus is of 32 bits, only the higher 30 bits from A31 –A2 are released by the up.
  - The lower 2 lines A1 and A0 are used internally by the uP to produce the four bank-enable signals $\overline{BE3}$.. $\overline{BE0}$
- Data Bus :
  - 80386 has a "32-bit" data bus. This means 80386 can transfer 32-bit data at a time.
  - It also has a 32-bit ALU, which means 80386 can operate on 32-bit numbers in one cycle.
  - Hence 80386 is called a "32-bit up".
  - 32-bit data is stored in 4 consecutive locations.
  - To transfer 32-bit data in one operation 80386 memory is divided into 4 banks of 1 GB each. The banks are enabled by 4 bank-enable signals: $\overline{BE3}$.. $\overline{BE0}$ produced by the µP.
- Address Pipelining
  - 80386 performs address pipelining, by putting address of the next machine cycle on the address bus, during T2 state of the current machine cycle.
  - This makes the decoder delay transparent and is especially useful for interfacing slower devices as it reduces the number of wait states.

MODULE 4 : INTEL 80386DX PROCESSOR

Features of 80386 :

- Virtual Memory
  - 80386 supports Virtual Memory which is implemented using Segmentation and Paging.
  - It can access a total Virtual Memory of 64 TB (2^46).
- Protection
  - 80386 use a protected model for accessing both memory and I/O. It uses 4 Privilege Levels.
- Multitasking
  - 80386 allows multitasking using timesharing.
  - Here several tasks can execute simultaneously by taking a small time slice of the uP. this gives higher system performance.
- I/0 Addressing
  - 80386 uses a 16-bit I/O address and hence can access up to 2^16 i.e. 65536 I/O devices with address0000 h FFFF h

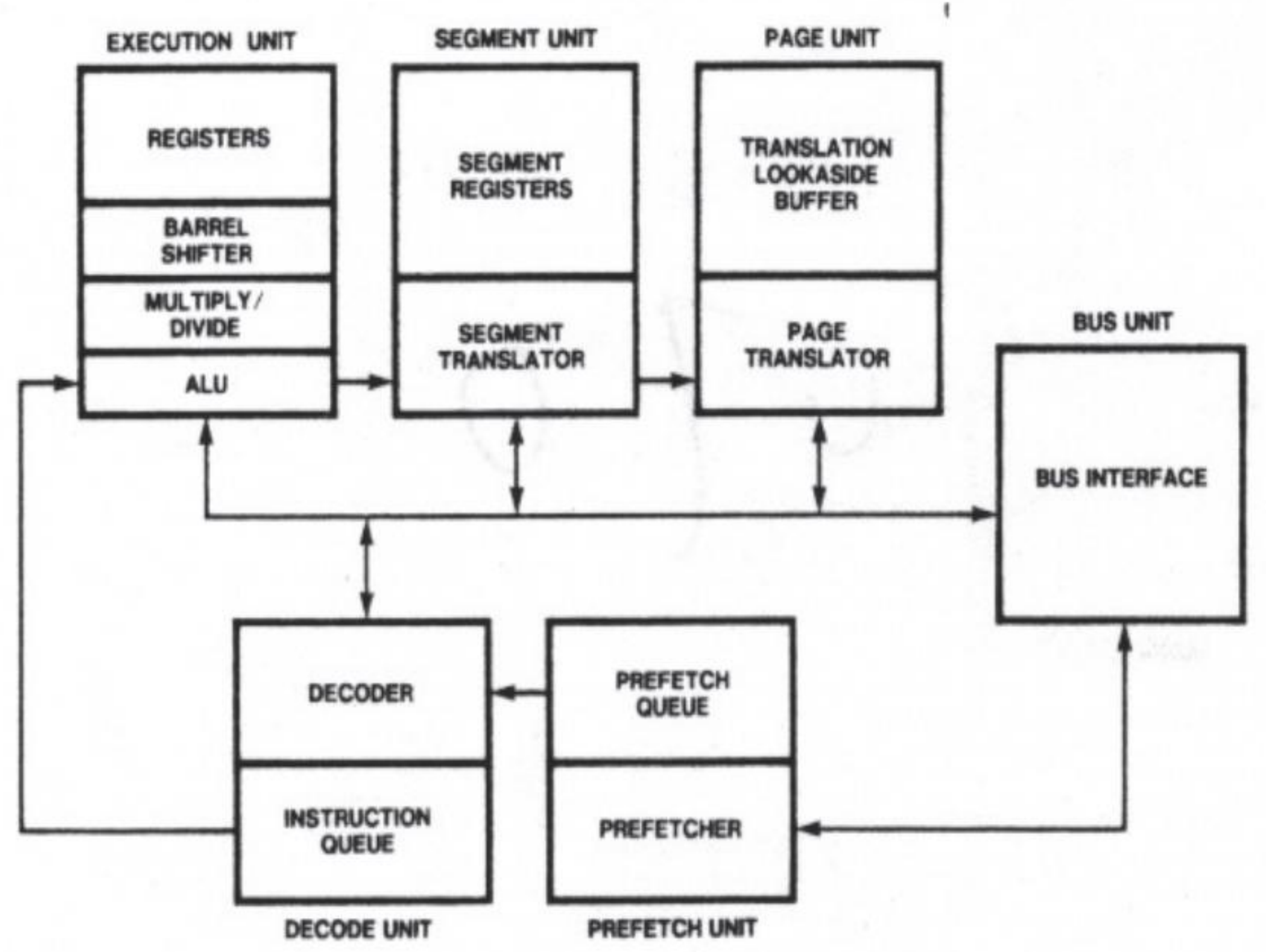
MODULE 4 : INTEL 80386DX PROCESSOR

❑ 80386 DX VS 80386 SX:

Sr No	80386 DX	80386 SX
1	80386 DX has a 32 bit data bus.	80386 SX has a 16 bit data bus.
2	Due to 32 bit data bus, the execution speed is higher. Hence the name: DX - Double Execution speed	Due to 16 bit data bus, the execution speed is lower. Hence the name: SX - Single Execution speed
3	32-bit transfers require 4 Memory Banks	16-bit transfers require 2 Memory Banks
4	4 Bytes are fetched at once in the pipelining queue.	2 Bytes are fetched at a time in the pipelining queue.
5	Has dynamic data bus sizing of 16-bit and 32-bit data bus, using BS16 signal.	No such option available as the data bus is only of 16-bits.Hence BS16 signal not useful
6	Used for high performance	Used for low cost memory and I/O system design.

MODULE 4 : INTEL 80386DX PROCESSOR

- Architecture:



MODULE 4 : INTEL 80386DX PROCESSOR

❑ Architecture:

- Bus Unit (Bus Interface Unit)
  - The Bus unit is responsible for transferring data in and out of the up.
  - It is connected to the external memory and I/O devices, using the system bus.
  - It gets requests from Prefetch unit for fetching instructions and from execution unit for transferring data.
  - If both requests occur simultaneously preference is given to execution unit.

MODULE 4 : INTEL 80386DX PROCESSOR

- Architecture:
- Prefetch Unit
  - The Pre-fetch unit fetches further instructions in advance to implement pipelining.
  - It fetches the next 16 bytes of the program and stores it into the Prefetch Queue.
  - It refills the queue when at least 4 bytes are empty as 80386 has a 32 bit data bus.
  - During a branch, the instructions in the queue are invalid and hence are discarded.
- Decode Unit
  - 80386 µP has a separate unit for decoding instructions called the Decode Unit.
  - It decodes the next three instructions and keeps them ready in the Decode Queue.
  - The decoded instructions are stored in Micro-Coded form.

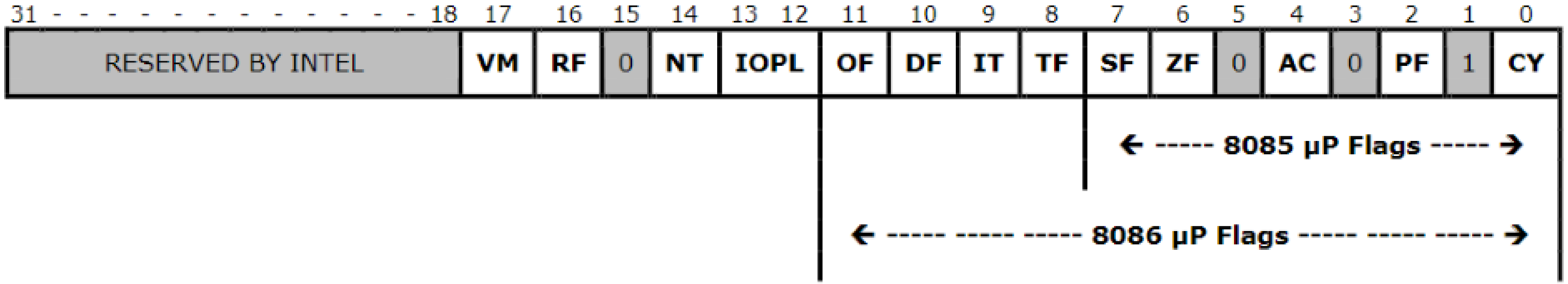
MODULE 4 : INTEL 80386DX PROCESSOR

❑ Architecture:

- Execution Unit
  - Execution Unit performs the main task of executing instructions
  - Normally, execution requires Arithmetic or Logic operations performed by a 32-bit ALU.
  - It also has dedicated circuits for 32-bit multiplication and division.
  - A 64-bit barrel shifter is also provided for faster shifts during multiplication and division.
  - Operands for the ALU can either be provided in the instruction, or can be taken from memory or could be taken from the 32-bit registers like EAX, EBX etc.
  - Additionally there is a 32-bit Flag register (EFLAGS) giving the Status of the current result.
- Memory Unit
  - The Memory unit converts Virtual Address (Logical address) to Physical Address
  - 80386 P implements 64 Terra bytes of Virtual memory using Segmentation Memory Unit is sub-divided into Segmentation Unit and Paging Unit.
  - Segmentation is compulsory, while Paging is optional.
  - The Segmentation Unit converts the Logical Address into a Linear Address.
  - The Paging Unit converts the Linear Address into a Physical Address.
  - If Paging is not used, then the Linear Address itself is the Physical Address.

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Flag register:



31 - - - - - - - - - - - - - 18	17	16	15	14	13 12	11	10	9	8	7	6	5	4	3	2	1	0
RESERVED BY INTEL	VM	RF	0	NT	IOPL	OF	DF	IT	TF	SF	ZF	0	AC	0	PF	1	CY

← ----- 8085 µP Flags ----- →
← ----- ----- ----- 8086 µP Flags ----- ----- ----- →

MODULE 4 : INTEL 80386DX PROCESSOR

Flag register:

- VM: Virtual mode
  - This flag is used to make 80386 operate in Virtual 8086 Mode (V6)
  - If VM=1, enter Virtual 8086 Mode.
  - V86 mode is basically used to run 8086 programs in a faster environment of 80386 uing multitasking and protection.
  - V86 mode can only be entered if uP is working in Protected Mode.
  - Once in Virtual 8086 Mode, we can return back to Protected Mode by making VM 0.
  - A special program called Virtual 8086 Monitor is responsible for switching back and forth between ProtectedMode and Real Mode.
- RF: Resume Flag
  - Resume flag is useful during debugging.
  - If RF = 1, then any debug fault in the next instruction will be ignored. RF is automatically reset after the next instruction.
  - In 80386 µP, some fault handlers (ISRs) return back to the same instruction that caused the fault instead ofrning back to the next instruction. By keeping RF = 1, we ensure that the program resumes after such a faultinstead of repeatedly generating breakpoint faults on the same instruction.

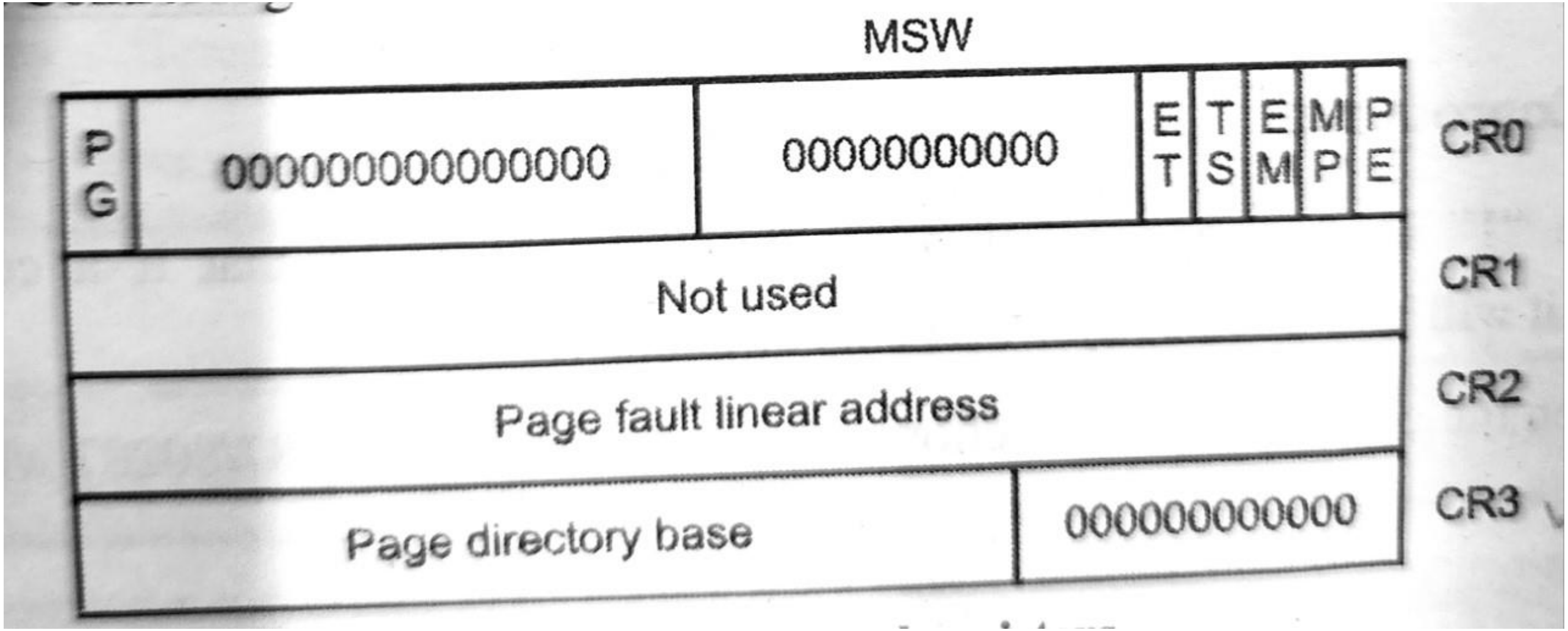
MODULE 4 : INTEL 80386DX PROCESSOR

❑ Flag register:

- NT: Nested Task
  - NT flag is used to indicate that the current task is nested i.c. it is invoked by another task.
  - If NT =1 then, the current task is Nested and its TSS (Task State Segment) has a valid "back link" to the TSS of the previous task.
  - This bit is automatically set whenever a Nested task is initiated, and thereafter can only be reset by software.
  - NT flag is checked by the IRET instruction to know whether it should perform an "Intra-task" return or an"Inter-task" return.
- IOPL: I/O Privilege Level
  - These bits are used to assign I/O Privilege Level.
  - 80386 uP has four privilege levels used for protection mechanism.
  - Privilege Level=0 is the highest Privilege Level and 3 is the lowest.
  - IOPL bits define the numerically maximum Privilege Level (logically lowest) at which a task must be running to access I/O devices.
  - If IOPL bits = 00 then only highest privileged tasks running at PL-0 can perform I/O instructions.

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Control Words :



		MSW						
PG	000000000000000	00000000000	ET	TS	EM	MP	PE	CR0
Not used								CR1
Page fault linear address								CR2
Page directory base		000000000000						CR3

Module 4 : Intel 80386DX Processor

- Control Register 0 :
  - PG: Paging Enable
    - This bit is made "1" to enable paging mode.
    - 80386 P implements Virtual Memory using the techniques of segmentation and paging. Though segmentation is compulsory, paging is optional. Paging is enabled using the PG bit of CR0. The default value of PG bit is "0".
  - ET: Extension Type
    - This bit is used to indicate the type of Math Co-Processor used with 80386.
    - If ET-1, then 80387 Math Co-Processor is used. If ET = 0, then 80287 Math Co-Processor is used

MODULE 4 : INTEL 80386DX PROCESSOR

- Control Registers 0
- TS: Task Switched
  - This bit is made "1" to indicate if a Task Switch is performed.
  - 80386 P implements "Multitasking", and thus it switches between various tasks, giving the programmer the impression that all tasks are running concurrently.
  - This significantly improves the overall system performance.
  - If TS = 1, it means a task switch is performed. Now the TSS of the current task has a back-link to theprevious task.
- EM: Emulate Coprocessor
  - This bit is made "1" in the absence of a Math Co-Processor so that if a coprocessor instruction encountered, then it will be executed by an on chip emulator.
  - If this bit is 0, then the coprocessor instruction will be executed by 80387/80287 whichever is present in desystem.
- MP: Math Co-processor Present
  - This bit is made "I" to indicate that a Math Co-Processor like 80387 or 80287 is present.Note: Out of EM bit and MP bit, only one of them must be "1“
- PE: Protection Enable
  - This bit is made "1" to enter protected mode. On reset, by default this bit is "0". It is the only bit of CRO which is also available in Real Mode

MODULE 4 : INTEL 80386DX PROCESSOR

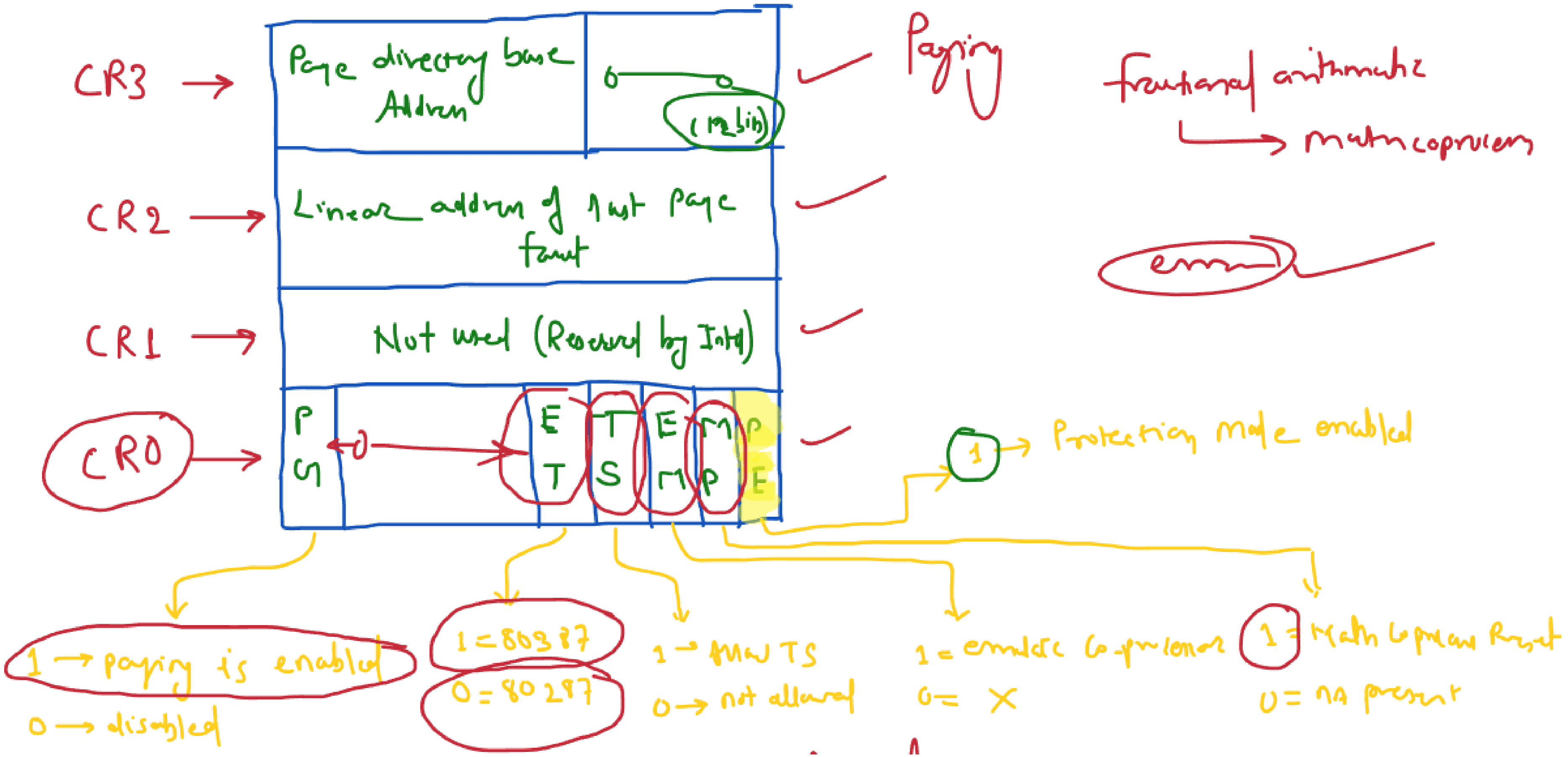
- Control Register 1 : Not in used
- Control Register 2 and 3 : Used only for Paging

MODULE 4 : INTEL 80386DX PROCESSOR

➢ Control Registers:

Control Registers of 80386

Monday, April 18, 2022 10:54 AM



MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Operating modes
  - ➢ Real Mode
  - ➢ Protected Mode
  - ➢ Virtual Mode

MODULE 4 : INTEL 80386DX PROCESSOR

- Real Mode :
  - It is the default mode selected when 80386 is reset. In this mode, 80386 uP simply behaves as a fast 8086 machine.
  - All registers are just like 8086. Even the memory used is only 1 MB, just like in 8086.
  - Physical addresscalculation is also like in 8086:

    Physical Address = Base Address x 10H + Offset Address

    Eg: Segment address (Base address)=5142H and offset address =0006H then the Physical address will be = 51426H
  - This mode is basically used to run the monitor program (BIOS) on reset.
  - Once the required registers are initialized, we can switch to Protected mode by making PE bit = 1 in CRO.

MODULE 4 : INTEL 80386DX PROCESSOR

- Real Mode :
- Segment Registers
  - There are 6, 16-bit segment registers containing the base addresses of their respective segments.
  - The registers are CS, SS, DS, ES, FS and GS.
- Offset Registers
  - There are 5, 16-bit offset registers containing the offset addresses for various segments.
  - The registers are IP, SP, BP SI and DL.
  - The 32-bit extended form (ESI, EDI etc) of these registers is not used in Real Mode.

Module 4 : Intel 80386DX Processor

- Real Mode :
- Data Registers
  - There are 4, 16-bit data registers used as General Purpose Registers during programming.
  - The registers are AX, BX CX and DX. They can be also used as 8, 8-bit registers AL, AH, BL, BH, CL, CH,DL and DH.
  - The 32-bit extended form (EAX, EBX etc) of these registers is not used in Real Mode.
- Flags
  - Only the lower 12-bits of the Flag Register are used in the Real Mode.
  - The 32-bit extended form of Flag Register i.e. EFLAGS is available only in Protected Mode.

MODULE 4 : INTEL 80386DX PROCESSOR

- Real Mode :
- Control Registers
  - Only the LSB of CRO is available in Real Mode.
  - The remaining bits of CRO and the remaining Control Registers are available only in Protected Mode.
  - The LSB of CRO is "PE" bit which should be made="1" to begin Protected Mode.
- Debug Registers and Test Registers: These registers are not available in Real Mode.
- Memory Range: The size of memory available in real mode is 1 MB and has address range from 00000H...FFFFFH, just like in 8086.
- I/O Range: A total of 64K I/O addresses are available having a range from 0000H... FFFFH, just like in 8086.

MODULE 4 : INTEL 80386DX PROCESSOR

- Protected Mode
- 80386 uP provides a very advanced mode of operations called the Protected Mode.
- In Protected Mode, 80386 uP provides dedicated hardware to prevent user programs from affecting other user programs and also safeguards the Operating System from being affected by user programs.
- There are Four Privilege Levels, assigned to programs and data to define their privileges.
- Level 0: This level is assigned to the Operating System Kernel (Main part of the Operating System).
  - It is the most privileged level.
  - Any program at this level can access all the data at any Privilege Level, whereas a data at this Privilege Level can only be accessed by a program at Privilege Level 0.

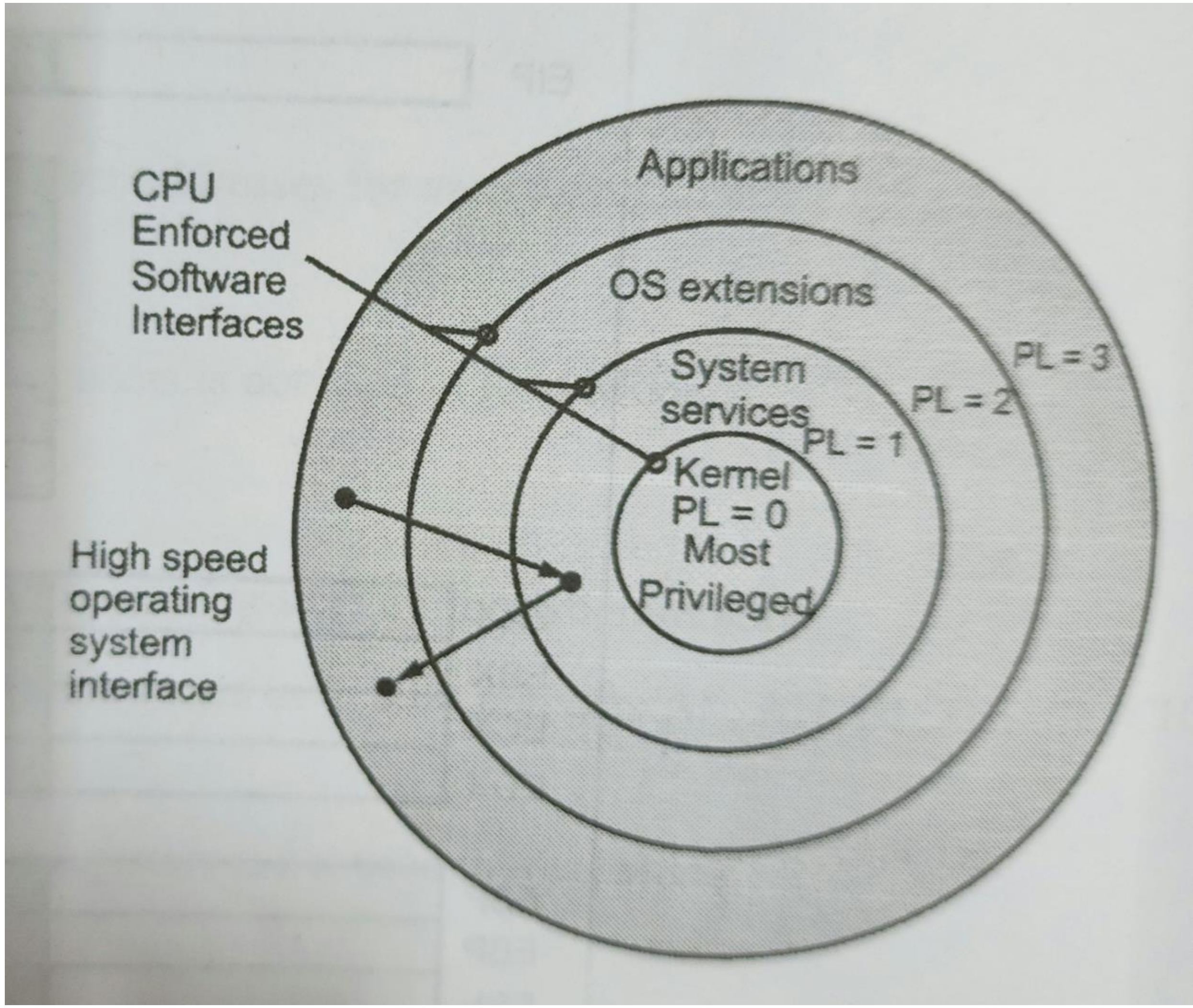
MODULE 4 : INTEL 80386DX PROCESSOR

❑ Protected Mode

- Level 1 : This level is assigned to the System Services such as File Handling, Device Drivers..
  - It is the 2nd most privileged level.Any program at this level can access the data at any Privilege Level which is lower than this level(numerically higher), whereas a data at this Privilege Level can only be accessed by a program at Privilege Level 0 or Privilege Level 1.
- Level 2: This level is assigned to the Custom Extensions of the OS.
  - It is the 3rd most privileged level.
  - Any program at this level can access the data at any Privilege Level which is lower than this level(numerically higher), whereas a data at this Privilege Level can only be accessed by a program at Privilege Level 0, 1 or 2.
- Level 3: This level is assigned to all the User Application and Programs.
  - It is the least privileged level.
  - Any program at this level can normally access the data at Privilege Level 3, whereas a data at this Privilege Level can be accessed by a program at any Privilege Level 0…3.

MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Protected Mode
- ➢ Privilege Levels



MODULE 4 : INTEL 80386DX PROCESSOR

- Protected Mode :
- Segment Registers
  - There are six, 16-bit segment registers.
  - Unlike Real mode, now these segment registers do not directly give the segment base address. Instead, these registers give a "Selector" which is basically an index in the GDT or in the LDT from which the descriptor is loaded. It is the descriptor which finally gives the base address of the segment.
- Offset Registers
  - There are five, 32-bit extended offset registers. The 16-bit offset address registers of Real Mode are now extended and hence give 32 bit offset addresses.
  - EIP gives the offset address for Code access.
  - ESP and EBP are used to give offset address for Stack operations.
  - ESI and EDI give offset address for several Data operations.

- ❑ Protected Mode :
- ➢ General Purpose Registers
  - ➢ There are four, 32-bit GPRS.
  - ➢ The 16-bit GPRS of Real Mode are now extended and hence used as 32-bit registers.
  - ➢ Besides being used as general purpose registers, they also serve some special purposes like, AX - acts as an accumulator in MUL/DIV/String operations, CX-is the default "count" register for several instructions etc.
- ➢ Flag Register80386 has a 32-bit flag register called EFLAGS used in Protected Mode.

MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Virtual Mode
- ✓ Virtual and Physical address space :
- ➢ Physical Memory is the total memory that can be directly connected to the CPU using its address bus
- ➢ Since 80386 uP has a 32 bit address bus, the total physical memory that can be connected is 2^32 = 4GB.
- ➢ Virtual Memory is the total memory space that can be addressed by the CPU registers.
- ➢ Virtual address is basically the combination of a 16-bit segment register and a 32 bit offset register.

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Virtual Mode

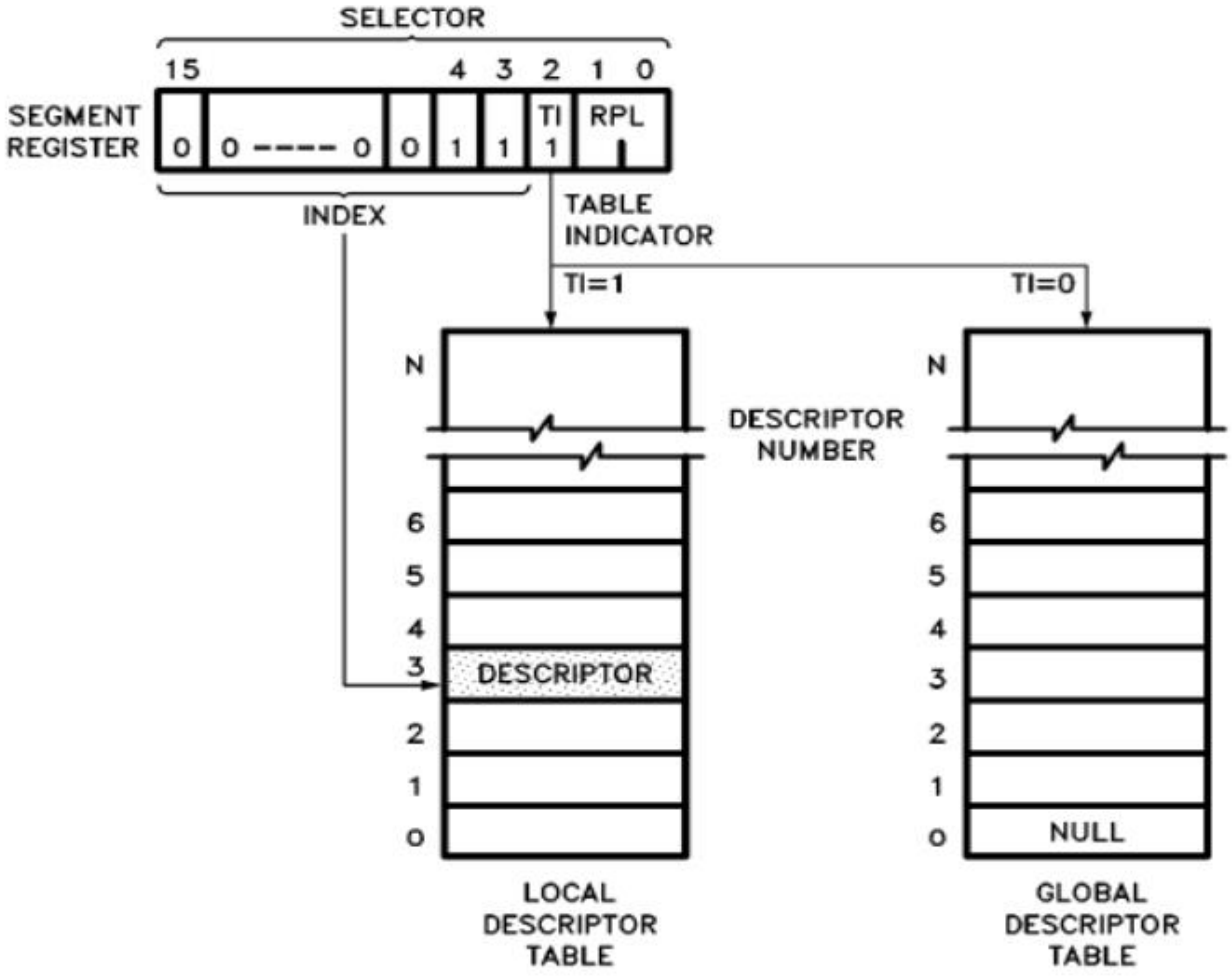
- In Real Mode, the segment register gives the starting address of the segment.
- But in Protected Mode, the segment register just gives a selector which selects a Descriptor for the segment
- Though the selector is of 16-bits, only 14 bits are used as two bits give the Privilege Level used for protection as seen above. Each selector value corresponds to a different segment. Hence there can be max 2segments.
- The locations within the segment are identified by their offset addresses. Since offset addresses are 32 bit,each segment can be max 2^32 = 4GB. Hence the max total Virtual memory that can be accessed = Max number of segments x Max size of each segment = 2^14 x 2^32 =2^46 =26 x 240 = 64 x 1TB = 64 Terra Bytes.
- Out of 14 bits of the selector, one bit is used as a table identifier (TI) if T1 = 1 then use LDT, if TI = 0 thenuse GDT

MODULE 4 : INTEL 80386DX PROCESSOR

- Virtual Mode
- So basically there are 2^13(ie. 8K) Descriptors in LDT and 8K Descriptors in GDT.
- This means the total 64TB Virtual space is divided into 32TB of Global space and 32 TB of Local Space.
- The total 48 bit address having 16 bit segment address and 32 bit offset address is called Virtual address. It is converted into a 32 bit physical address using two translations: Segment translation (compulsory) and page translation (optional).
- Segment translation converts 48-bit Virtual address into 32 bit Linear Address which is further converted into a 32 bit Physical Address by Page translation.

MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Virtual Mode
- ✓ Segment Translation :

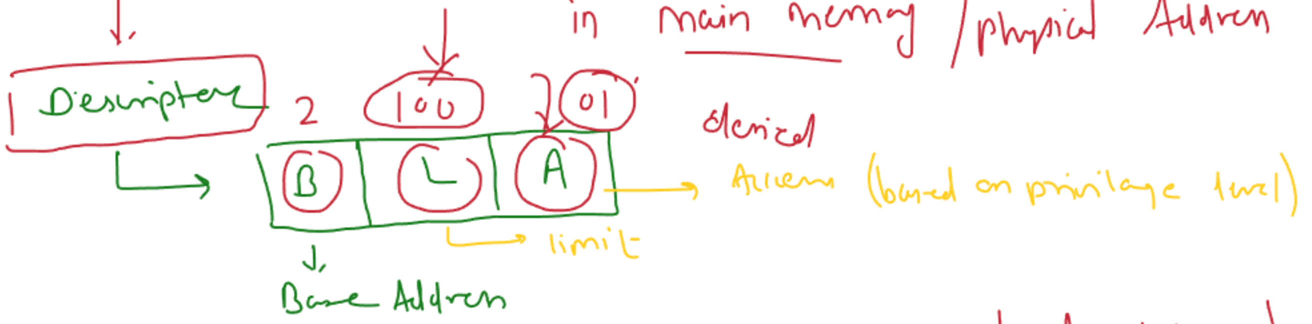


MODULE 4 : INTEL 80386DX PROCESSOR

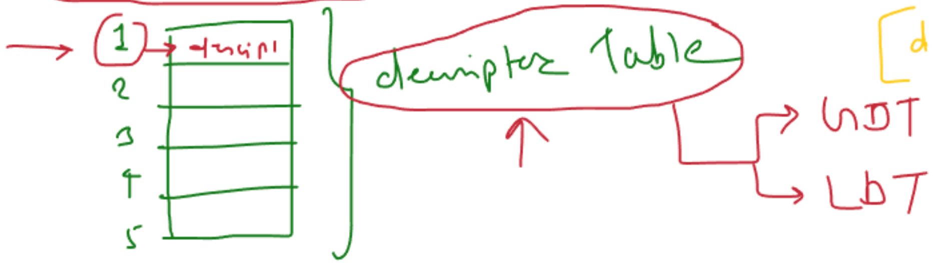
❑ Virtual Mode :

★ Every segment has descriptor (no of segments = no of descriptor)

↳ Base address of file/segment if it is present in main memory/physical Address



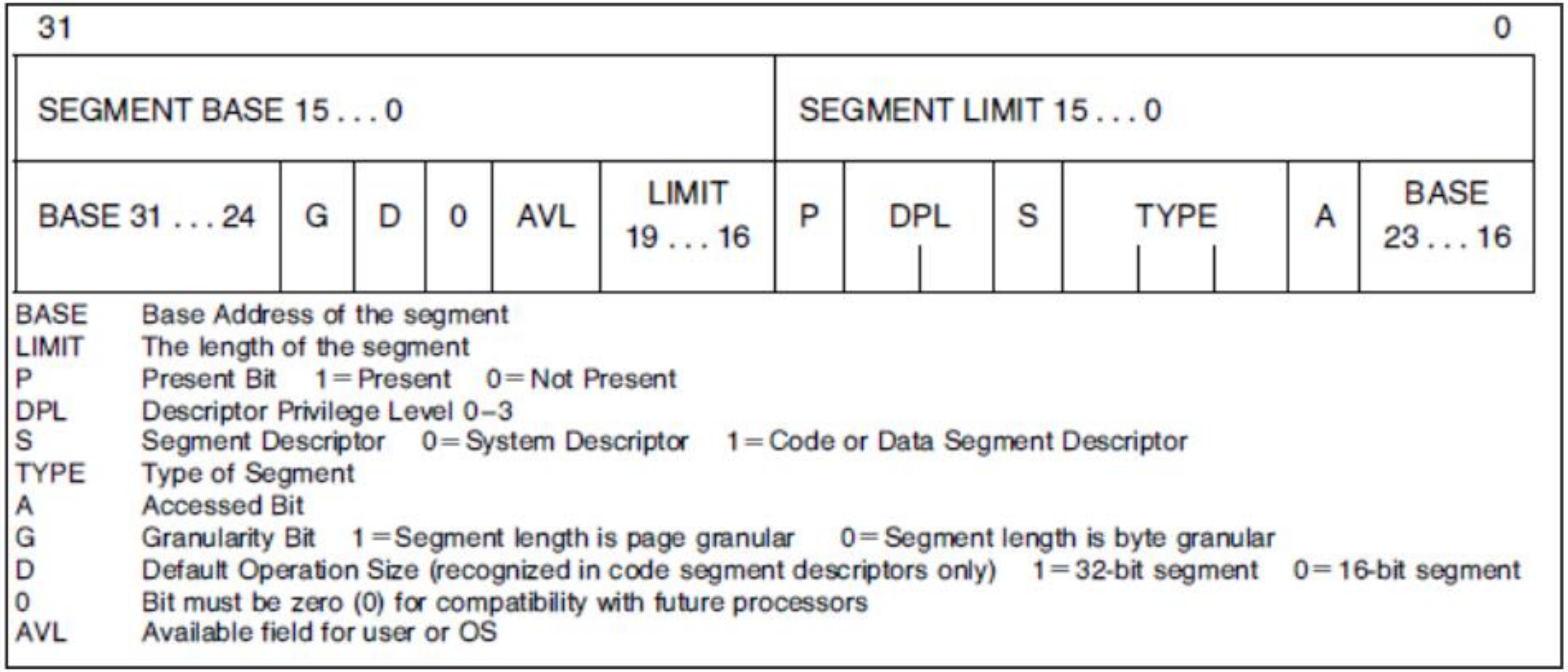
★ All descriptors are kept in table called descriptor table



[descriptor no = segment no]

❑ Virtual Mode :

Basic structure of a segment descriptor



31 0

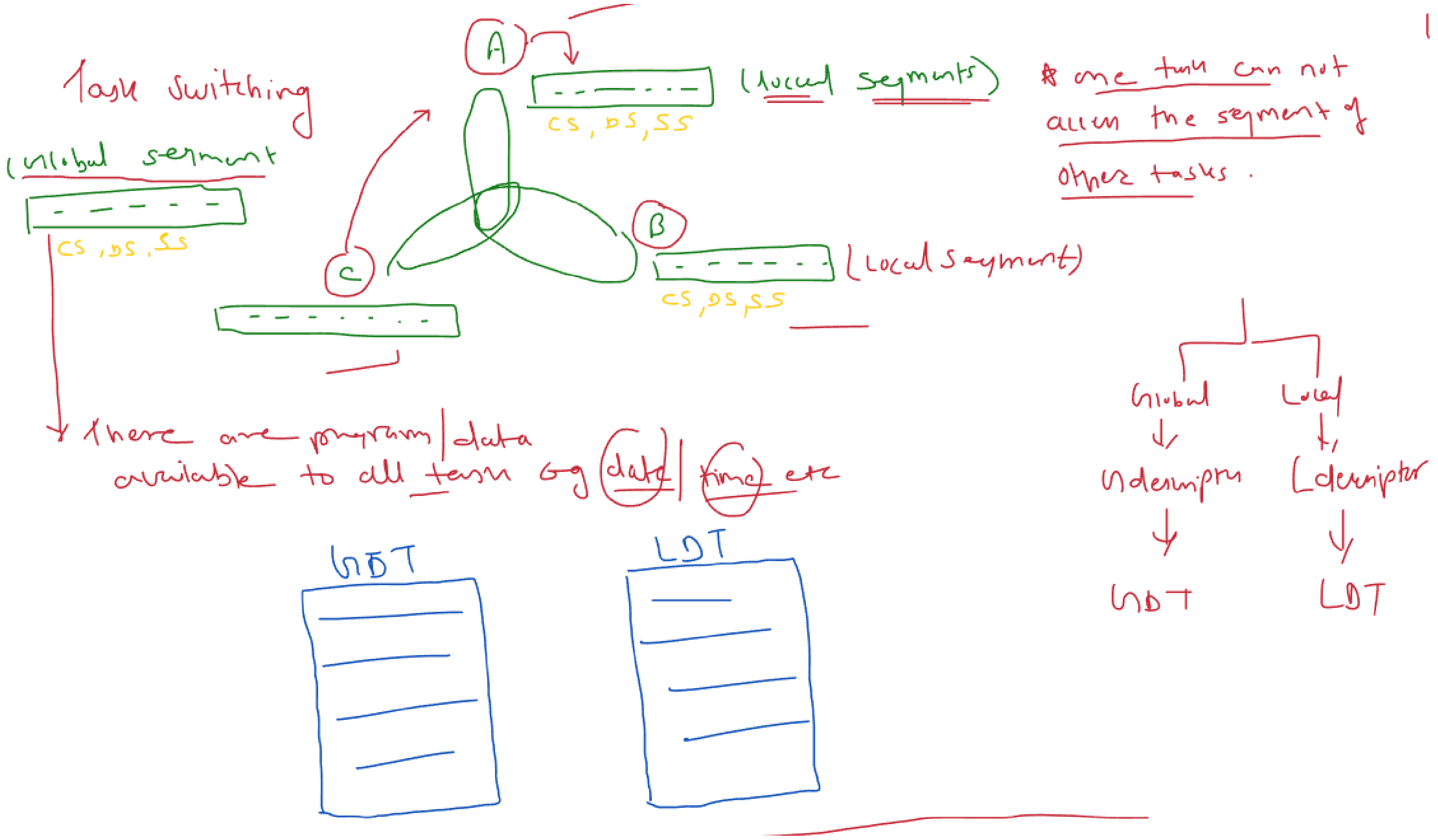
SEGMENT BASE 15 . . . 0	SEGMENT LIMIT 15 . . . 0

BASE 31 . . . 24	G	D	0	AVL	LIMIT 19 . . . 16	P	DPL	S	TYPE	A	BASE 23 . . . 16

BASE	Base Address of the segment
LIMIT	The length of the segment
P	Present Bit 1 = Present 0 = Not Present
DPL	Descriptor Privilege Level 0–3
S	Segment Descriptor 0 = System Descriptor 1 = Code or Data Segment Descriptor
TYPE	Type of Segment
A	Accessed Bit
G	Granularity Bit 1 = Segment length is page granular 0 = Segment length is byte granular
D	Default Operation Size (recognized in code segment descriptors only) 1 = 32-bit segment 0 = 16-bit segment
0	Bit must be zero (0) for compatibility with future processors
AVL	Available field for user or OS

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Virtual Mode :

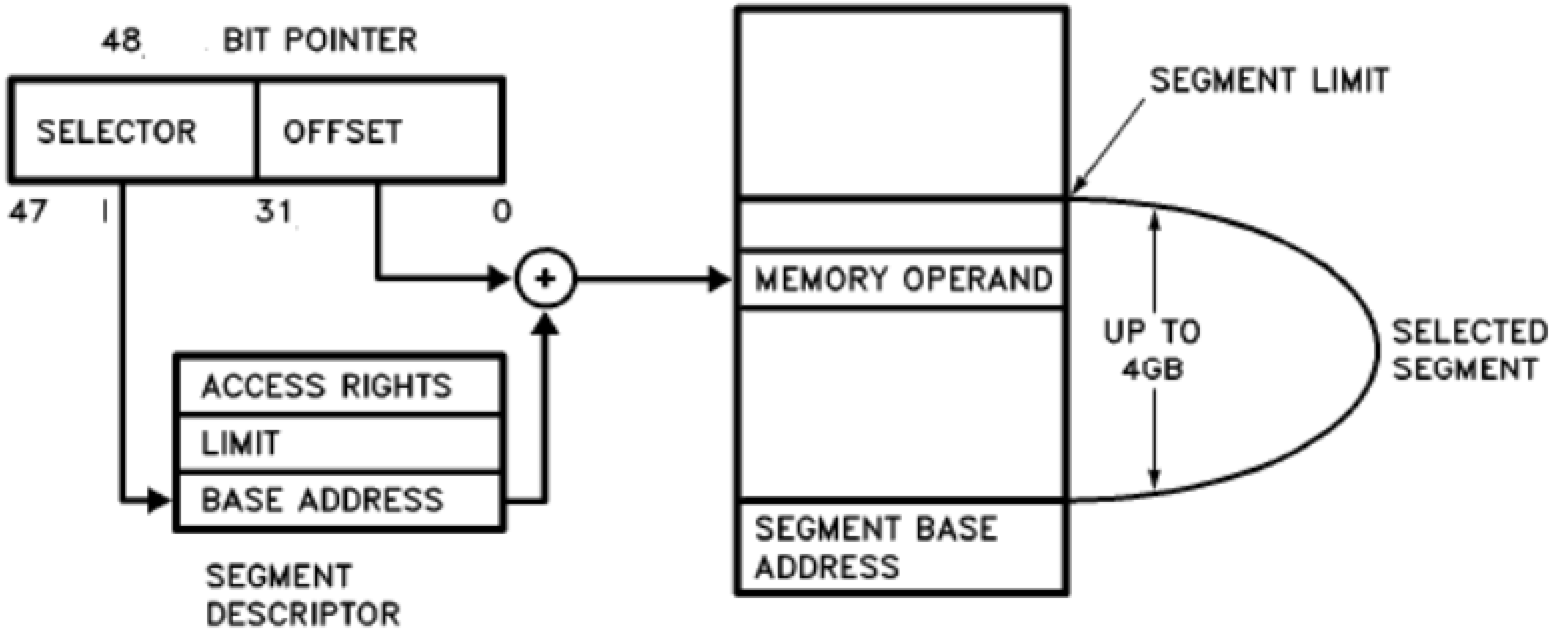


* one task can not access the segment of other tasks.

There are program/data available to all tasks eg date/time etc

MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Virtual Mode :
- ✓ Segment Translation :



MODULE 4 : INTEL 80386DX PROCESSOR

□ Virtual Mode :

✓ Page Translation :

- ➢ The Virtual memory space is divided into equal size blocks of 4KB called “pages”
- ➢ A physical memory space (also called main memory) is also divided into equal size blocks of 4kb called page frames (also simply called pages)
- ➢ As Physical Memory is of 4 GB and page size is 4 KB there are total 1 M pages (2) in the Physical Memory.
- ➢ A page from Virtual Memory is loaded into any available page frame of Physical Memory.
- ➢ Whenever a page is required to be accessed, the up first checks if the desired page is present in the Physical Memory. If so, it is called a "HIT" and the operation is performed on the Physical Memory.
- ➢ A "Page Fault" (MISS) occurs when the desired page is not present in the Physical Memory.
- ➢ On a Page Fault the desired page is loaded form Virtual Memory into any available page frame of Physical Memory.

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Virtual Mode :

✓ Page Translation :

- ➢ If no page frame is available, then a "Page Replacement" is performed by replacing an old page from thePhysical Memory with the new desired page from the Virtual Memory. Various algorithms like FIFO (First in first out), LRU (Least recently used) or LFU (Least frequently used) are used to determine which page of the Physical Memory must be replaced.
- ➢ Once the page to be replaced is decided, a "Dirty Bit" is checked to determine if the page is modified in the Physical Memory.
- ➢ If Dirty bit = 1, then the page has been modified (is "Dirty") and hence must be copied back into VirtualMemory before being replaced else the modified information will be lost.
- ➢ If the Dirty bit = 0, then the page is not modified and hence can be directly replaced without being copied back into Virtual Memory

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Virtual Mode :

✓ Page Translation :

➢ Since a page of Virtual Memory can be loaded into any page frame of Physical Memory, a "Page Table" is required to give the mapping between Virtual Memory page number and Physical Memory page framenumber.

➢ Simply speaking the Page table tells which page of Virtual Memory is present in which page of Physical Memory.

➢ But since there are too many page frames in the Physical Memory (220 i.e. IM), the page table will becometoo large and searches will become extremely slow.

➢ Hence the mechanism is further subdivided.

➢ Instead of having straight IM (20) entries in the page table, there are IK (2) entries in a page table and

➢ (22 x 20......!)there are 1K (2) such page tables.

➢ Each page table is of 4KB and has IK "Page Table Entries" (PTES) each of size 4 bytes. Each PTE gives information about a Page Frame.

➢ The PTE has following information:

MODULE 4 : INTEL 80386DX PROCESSOR

□ Virtual Mode :

✓ Page Translation :

✓ 20 bit page frame address: Gives the upper 20 bits of the starting address of the page frame. Lower 12 bits are 0...0 as the page is of 4 KB and starts from a4 KB aligned location.

✓ D: Dirty bit indicates whether the page has been modified (1) or not (0).

✓ A: Accessed Bit tells whether the page has been accessed or not (1 means accessed). This is used by replacement algorithms

✓ U/S: User or Supervisor and R/W: Read or Read and Write give protection information

✓ P: Present bit indicates whether the page is present in the Physical MemoryIf P=1 then the page is present and the 20 bit address field is valid, else the page is not present in thePhysical Memory and the 20 bit address field is obviously invalid

Module 4 : Intel 80386DX Processor

- ❑ Virtual Mode :
- ✓ Page Translation :
- ✓ Information about all the page tables is stored in the "Page Directory".
- ✓ The page directory is of 4KB and has IK "Page Directory Entries" (PDEs) each of size 4 bytes. Each PDEgives information about a Page Table.
- ✓ The PDE has following information:
  - ✓ 20 bit page table address: Gives the upper 20 bits of the starting address of the corresponding page table. Lower 12 hits are 0...0 as the page table is of 4 KB and starts from a 4 KB aligned location
  - ✓ D: Dirty bit (explained above)
  - ✓ A: Accessed Bit (explained above)
  - ✓ U/S: User or Supervisor and R/W: Read or Read and Write
  - ✓ P: Present bit (explained above)

MODULE 4 : INTEL 80386DX PROCESSOR

❑ Virtual Mode :

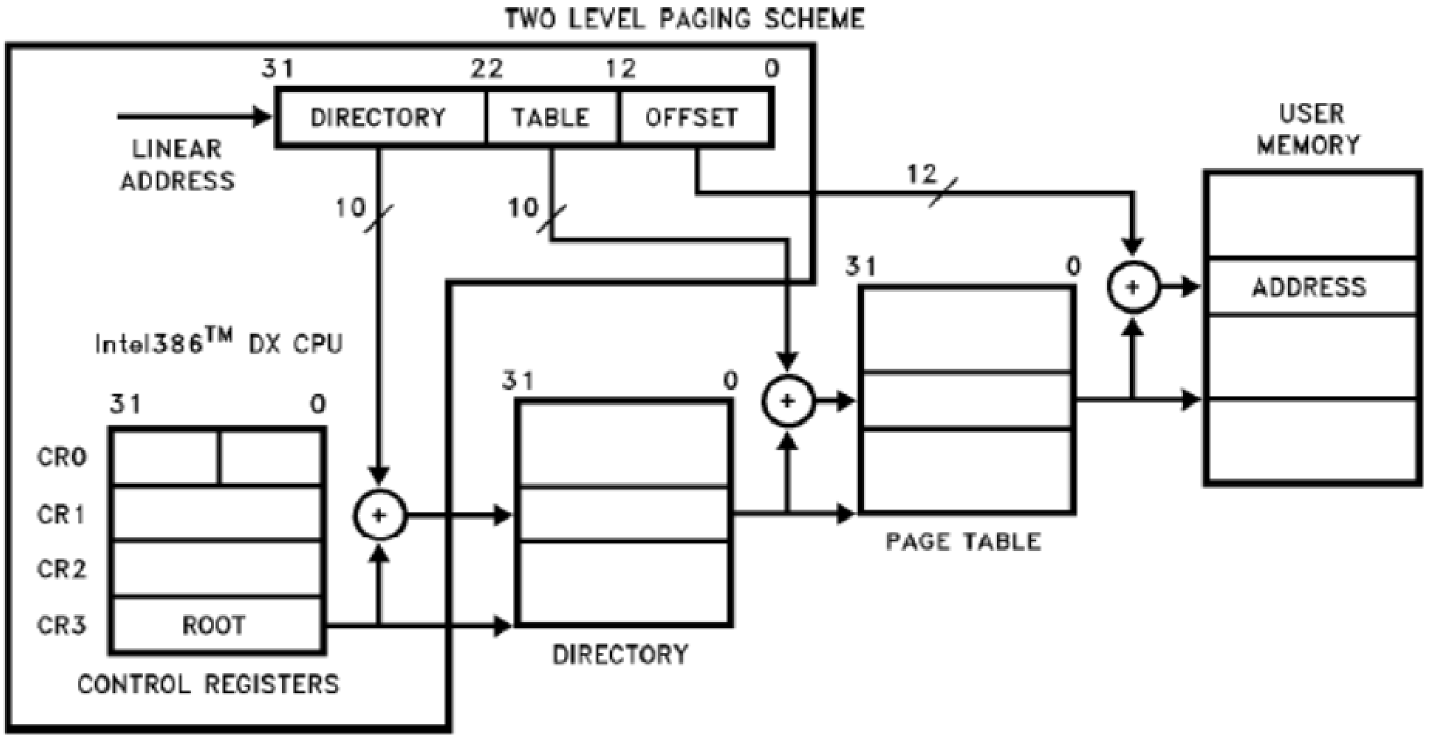
- ✓ The Page Directory is of 4 KB and begins from a 4 KB aligned location.
- ✓ The address of the page directory is given by the PGBR (page Directory Base Register) field in CR3.
- ✓ The 32 bit Linear Address can be divided into three parts:
- ✓ The higher 10 bits select one PDE out of IK PDEs in the page directory.This gives the starting address in the page table.
- ✓ The next 10 bits select one PTE out of IK PTEs in the page table.
- ✓ This gives the starting address of the page frame.
- ✓ Finally, the lowest 12 bits (offset) select a location within the 4KB page.
- ✓ This means, to access any location, uP must first access a PDE in the page directory then a PTE in the page table, then access the page. This can make the process very slow. To speed up the process a "TranslationLook-aside Buffer" is used (called TLB).

MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Virtual Mode :
- ✓ Page Translation :

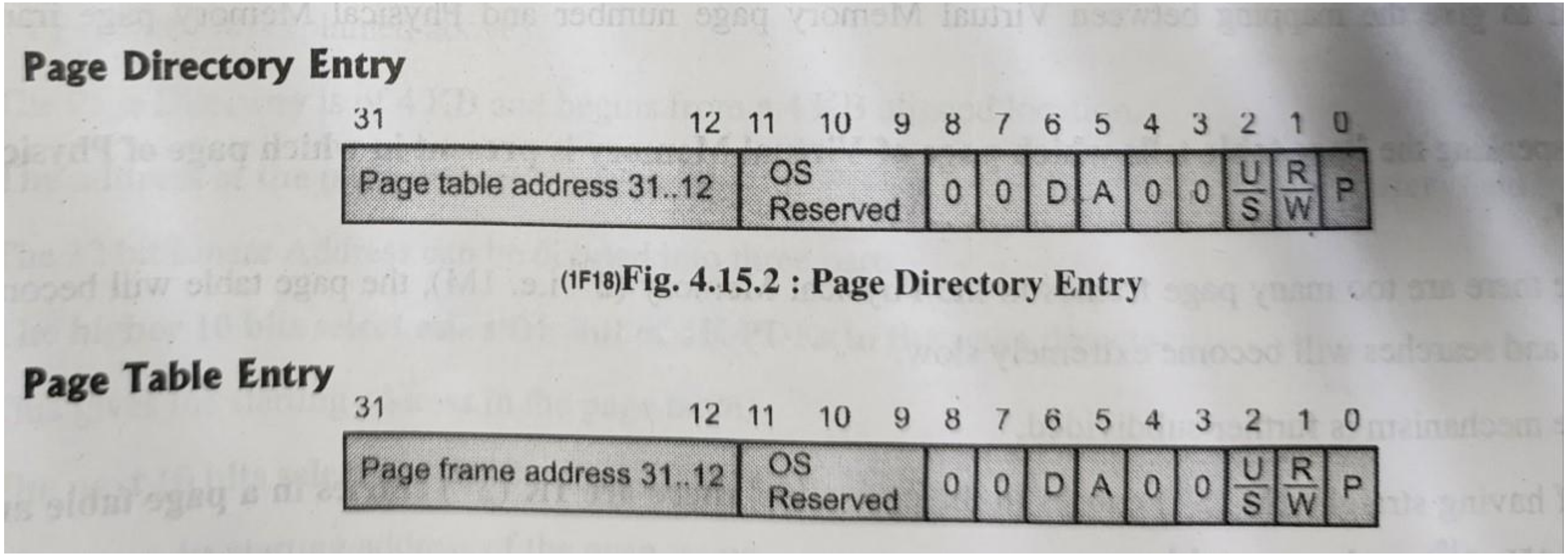
PAGE TRANSLATION

Paging Mechanism:



MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Virtual Mode :
- ✓ Page Translation :



Page Directory Entry

31 12	11 10 9	8	7	6	5	4	3	2	1	0
Page table address 31..12	OS Reserved	0	0	D	A	0	0	U S	R W	P

(1F18)Fig. 4.15.2 : Page Directory Entry

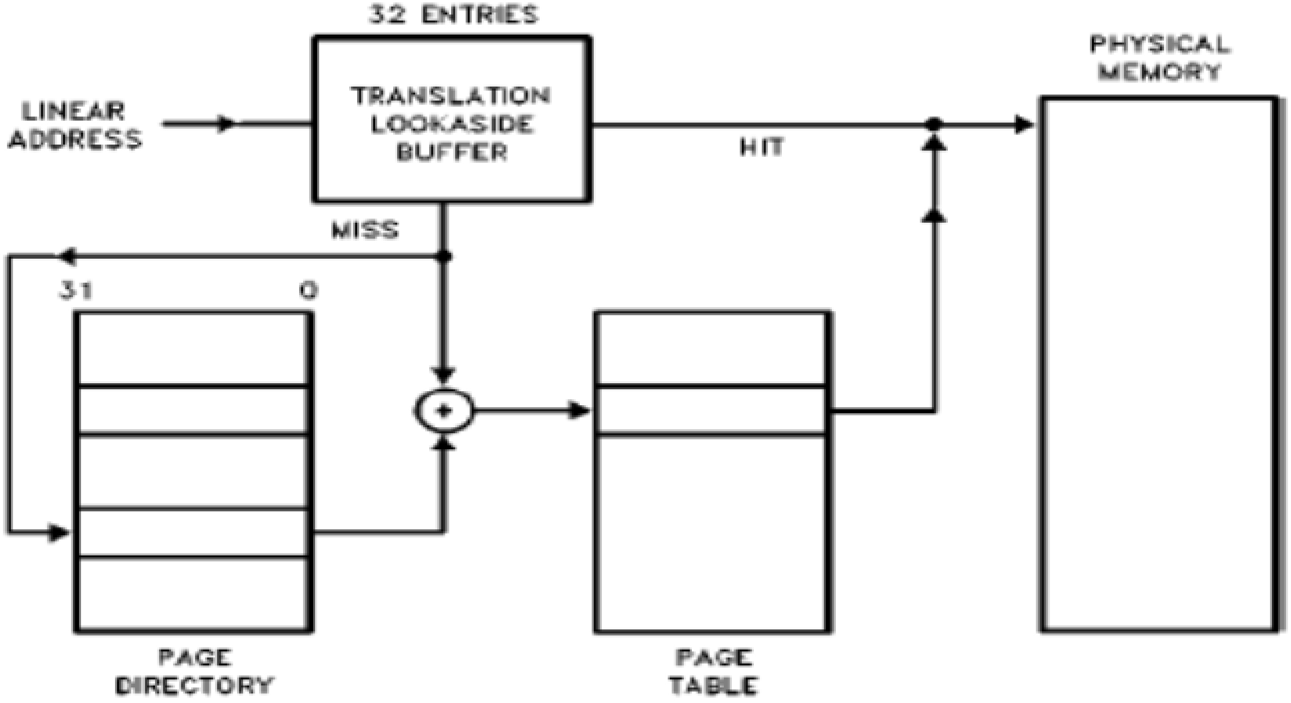
Page Table Entry

31 12	11 10 9	8	7	6	5	4	3	2	1	0
Page frame address 31..12	OS Reserved	0	0	D	A	0	0	U S	R W	P

MODULE 4 : INTEL 80386DX PROCESSOR

- ❑ Virtual Mode :
- ✓ Page Translation :

Translation Look-aside Buffer (TLB):


32 ENTRIES
TRANSLATION LOOKASIDE BUFFER
LINEAR ADDRESS
HIT
MISS
PHYSICAL MEMORY
31
0
PAGE DIRECTORY
PAGE TABLE