Important Instructions to examiners:

1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills.
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate’s answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate’s understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q1 a) Attempt any three of the following **12M**

i) Summarize the features of 80286? (any four)

Ans:

(any 4 features-1 m each)

- The 80286 microprocessor is an advanced version of the 8086 microprocessor that is designed for multiuser and multitasking environments
- The 80286 addresses 16 M Byte of physical memory and 1G Bytes of virtual memory by using its memory-management system
- The 80286 is basically an 8086 that is optimized to execute instructions in fewer clocking periods than
  The 8086
- Like the 80186, the 80286 doesn’t incorporate internal peripherals; instead it contains a memory management
  Unit (MMU)
- The 80286 operates in both the real and protected modes
- In the real mode, the 80286 addresses a 1MByte memory address space and is virtually identical to 8086
- In the protected mode, the 80286 addresses a16MByte memory space
- The clock is provided by the 82284 clock generator, and the system control signals are provided by the 82288 system bus controller
- The 80286 contains the same instructions except for a handful of additional instructions that control the memory-management nit

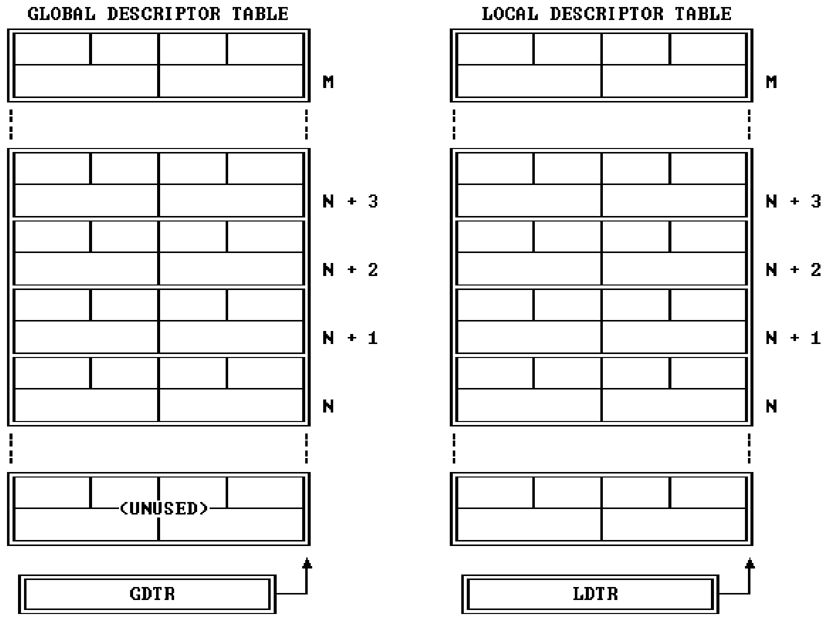
ii) Explain the system address register of microprocessor 80386?

Ans: (Diagram 2M & Explain 2M)

- 80386 has 32registers resources in different categories, system address register is one of them.
- System Address Registers: Four special registers are defined to refer to the descriptor tables supported by 80386.
    - The 80386 supports four types of descriptor tables
- Global descriptor table (GDT),
- Interrupt descriptor table (IDT),
- Local descriptor table (LDT) and
- Task State Segment(TS)
- The addresses of these tables and segments are stored in special registers known as System Address and System Registers namely as….
- GDTR Global Descriptor Table Register
    - IDTR Interrupt Descriptor Table Register : This register points to a table of entry points for interrupt handlers :

    - These registers point to the segment descriptor tables i.e. they hold linear base address &16 bit limit of GDT and IDT respectively.
- Task Register
- This register points to the information needed by the processor to define the current task
    - LDTR Local Descriptor Table Register Above two registers hold 16 bit selector for LDT & TSS descriptor respectively.

Figure 5-5. Descriptor Tables



iii) Describe the function of special interrupts of X 86 processor.

Ans:

Special interrupts are given as below: (1M)

- Divide by zero
- Single step
- NMI(non maskable interrupt)
- Breakpoint
- Overflow
- INTR

Function: (3M)

- The ISR for divide-error can do anything the user wishes to recover from the error.
- Single step can be used for debugging.
- The processor cannot avoid NMI, computer must respond to it after suspending on going operation.
- When INTO instruction is executed with the overflow flag set, overflow interrupt is executed.
- INTR is the interrupt which is to be acknowledged by INTA where even external Interrupt controller can be connected which allows more hardware interrupts to be connected.

iv) List the silent features of 80486

Ans:

80486 Basic Features **(Any 4 features-1 m each)**

- The 80486 microprocessor is an improved version of the 80386 microprocessor that contains an 8K-byte cache and an80387arithmetic co-processor; it executes many instructions in one clocking period
- The 80486 microprocessor executes a few new instructions that control the internal cache memory.
- A new feature found in the 80486 in the BIST (built-in self-test) that tests the microprocessor, coprocessor, and cache at reset time
- If the 80486 passes the test, EAX contains a zero.
- Additional test registers are added to the 80486 to allow the cache memory to be tested
- These new test registers are TR3 (cache data), TR4 (cache status), and TR5 (cache control)
- Complete 32 bit architecture- address bus, data bus & registers
- On chip 8Kbyte code & Data cache.

Q1b) Attempt any one if the following **6M**

i) Illustrate the virtual addressing mode concept of 80286.

Ans: **(Diagram: 2 marks, explanation: 4 marks**)

- In computing, **protected mode**, also called **protected virtual address mode**, is an operational mode of x86-compatible central processing units (CPU).
- It allows system software to use features such as virtual memory, paging and safe multi-tasking designed to increase an operating system's control over application software.
- Protected mode may only be entered after the system software sets up several descriptor tables and enables the Protection Enable (PE) bit in the control register 0 (CR0).
- Protected mode gives extended physical & virtual memory address space, memory protection mechanisms to support operating system & virtual memory.
- It also provides memory management and protection mechanisms and associated instructions

Diagram

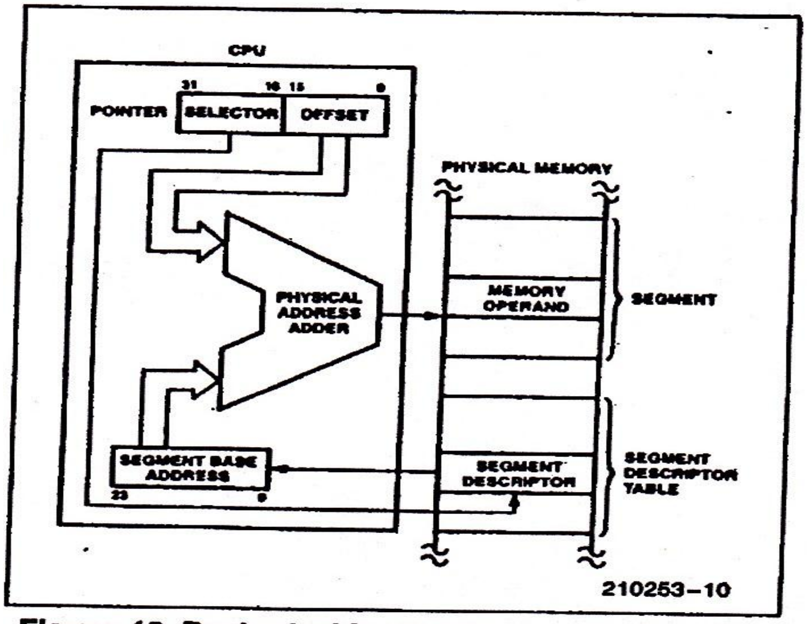


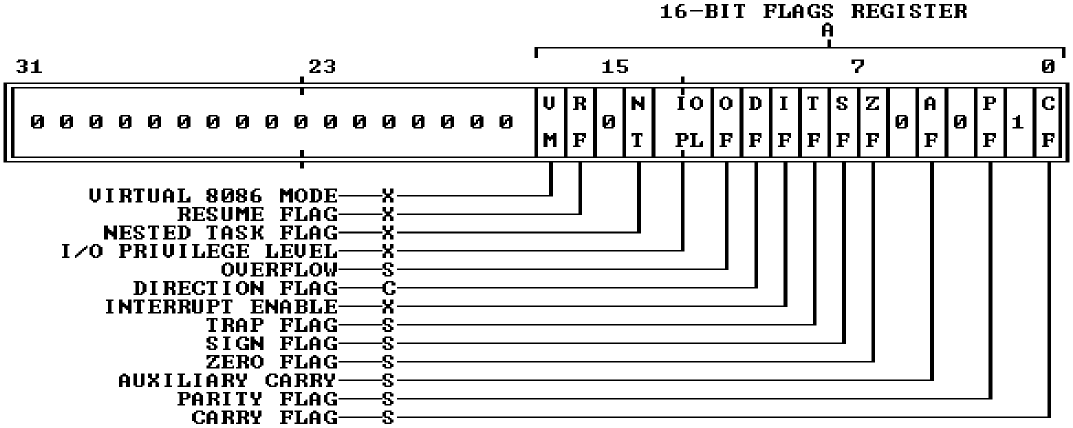
Figure 10. Protected Mode Memory Addressing

ii) With a suitable figure explain the flag register format of 80386.

Ans:

Systems Flags: **(diagram: 2marks, explanation: 4marks)**

Figure 2-8. EFLAGS Register



S = STATUS FLAG, C = CONTROL FLAG, X = SYSTEM FLAG

NOTE: 0 OR 1 INDICATES INTEL RESERVED. DO NOT DEFINE

Virtual 8086 Mode, bit 17)

The VM bit provides Virtual 8086 Mode within Protected Mode. If set while the 80386 is in Protected Mode, the 80386 will switch to Virtual 8086 operation, handling segment loads as the 8086 does, but generating exception 13 faults on privileged opcodes.

RF (Resume Flag, bit 16)

The RF flag is used in conjunction with the debug register breakpoints. It is checked at instruction boundaries before breakpoint processing. When RF is set, it causes any debug fault to be ignored on the next instruction. RF is then automatically reset at the successful completion of every instruction.

NT (Nested Task, bit 14)

This flag applies to Protected Mode. NT is set to indicate that the execution of this task is nested within another task. If set, it indicates that the current nested task's Task State Segment (TSS) has a valid back link to the previous task's TSS. This bit is set or reset by control transfers to other tasks.

IOPL (Input/Output Privilege Level, bits 12-13)

This two-bit field applies to Protected Mode. IOPL indicates the numerically maximum CPL (current privilege level) value permitted to execute I/O instructions without generating an exception 13 fault or consulting the I/O Permission Bitmap. It also indicates the maximum CPL value allowing alteration of the IF (INTR Enable Flag) bit when new values are popped into the EFLAG register.

OF (Overflow Flag, bit 11)

OF is set if the operation resulted in a signed overflow. Signed overflow occurs when the operation resulted in carry/borrow into the sign bit (high-order bit) of the result but did not result in a carry/borrow out of the high order bit, or vice-versa. For 8/16/32 bit operations, OF is set according to overflow at bit 7/15/31, respectively.

DF (Direction Flag, bit 10)

DF defines whether ESI and/or EDI registers post decrement or post increment during the string instructions. Post increment occurs if DF is reset. Post decrement occurs if DF is set.

IF (INTR Enable Flag, bit 9)

The IF flag, when set, allows recognition of external interrupts signaled on the INTR pin.

When IF is reset, external interrupts signaled on the INTR are not recognized. IOPL indicates the maximum CPL value allowing alteration of the IF bit when new values are popped into EFLAGS or FLAGS.

TF (Trap Enable Flag, bit 8)

TF controls the generation of exception 1trap when single-stepping through code. When TF is set, the 80386 generates an exception 1 trap after the next instruction is executed. When TF is reset, exception 1 traps occur only as a function of the breakpoint addresses loaded into debug registers DR0±DR3.

SF (Sign Flag, bit 7)

SF is set if the high-order bit of the result is set, it is reset otherwise. For 8-, 16-, 32-bit operations, SF reflects the state of bit 7, 15, 31 respectively.

ZF (Zero Flag, bit 6)

ZF is set if all bits of the result are 0. Otherwise it is reset.

AF (Auxiliary Carry Flag, bit 4)

The Auxiliary Flag is used to simplify the addition and subtraction of packed BCD quantities. AF is set if the operation resulted in a carry out of bit 3 (addition) or a borrow into bit 3 (subtraction). Otherwise AF is reset. AF is affected by carry out of, or borrows into bit 3 only, regardless of overall operand length: 8, 16 or 32 bits.

PF (Parity Flags, bit 2)

PF is set if the low-order eight bits of the operation contains an even number of ``1's'' (even parity). PF is reset if the low-order eight bits have odd parity. PF is a function of only the low-order eight bits, regardless of operand size.

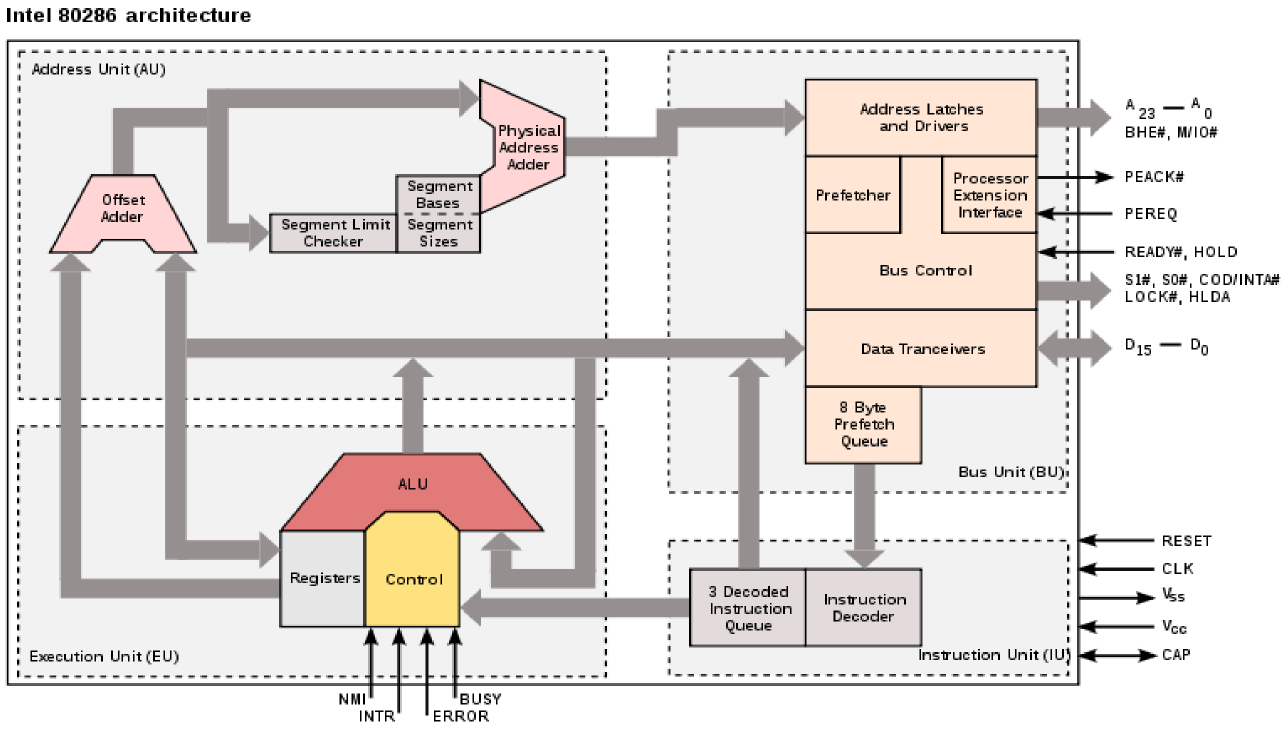
CF (Carry Flag, bit 0)

CF is set if the operation resulted in a carry out of (addition), or a borrow into (subtraction) the high-order bit. Otherwise CF is reset. For 8-, 16- or 32-bit operations, CF is set according to carry/borrow at bit 7, 15 or 31, respectively.

Q 2 Attempt any Two of the following 16M

a) With neat labeled diagram explain the internal architecture of 80286.

Ans: **(Diagram:4 marks, explanation: 4 marks)**



The 80286 is an advanced, high performance microprocessor with specially optimized capabilities for multiple user and multi-tasking systems. The 80286 has built in memory protection that supports operating system and task isolation as well as program and data privacy with tasks. A 12 MHz 80286 provides six times or more throughput than the standard 5 MHz 8086.The 80286 includes memory management capabilities that map 230 (one gigabyte) of virtual address space per task into 224 bytes (16 megabytes) of physical memory.

.Using 8086 real address mode, the 80286 is object code compatible with existing 8086,88 software. In protected virtual address mode, the 80286 is source code compatible with 8086, 88 software and may require upgrading to use virtual addresses supported by the 80286's integrated memory management and protection mechanism

The 8086, 88,186,and 286 CPU family all contain the same basic set of registers ,instructions ,and addressing modes. The 80286 processor is upward compatible with the 8086, 8088, and 80186 CPU's.

Register Set

The 80286 base architecture has fifteen registers. These registers are grouped into the following four categories:

General Registers: Eight 16-bit general purpose register used to contain arithmetic and logical operands. Four of these (AX,BX,CX and DX) can be used either in their entirety as 16-bit words or split into pairs of separate 8-bit registers.

Segment Registers: Four 16-bit special purpose registers select, at any given time, the segments of memory that are immediately addressable for code stack, and data.(For usage ,refer to memory organization.)

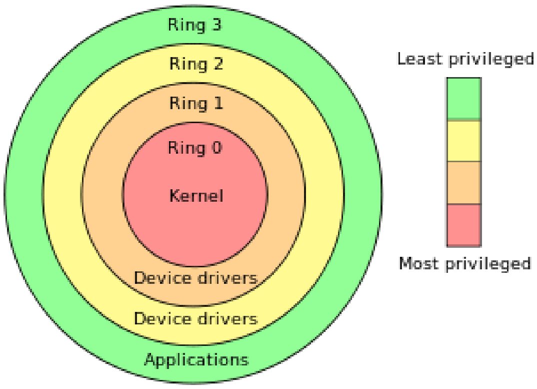
Base and Index Registers: Four of the general purpose registers may also be used to determine offset addresses of operands in memory. These registers may contain base addresses or indexes to particular locations with a segment. The addressing mode determines the specific registers used for operand address calculations.

Status and Control Registers: The 3 16-bit special purpose registers in figure 3A record or control certain aspects of the 80286 processor state including the instruction pointer, which contains the offset address of the next sequential instruction to be executed.

b) Explain privilege protection and system segment , gate descriptor of microprocessor

Ans: (Explanation in detail is not expected)

Privilege Protection levels in 80286 **(3M)**



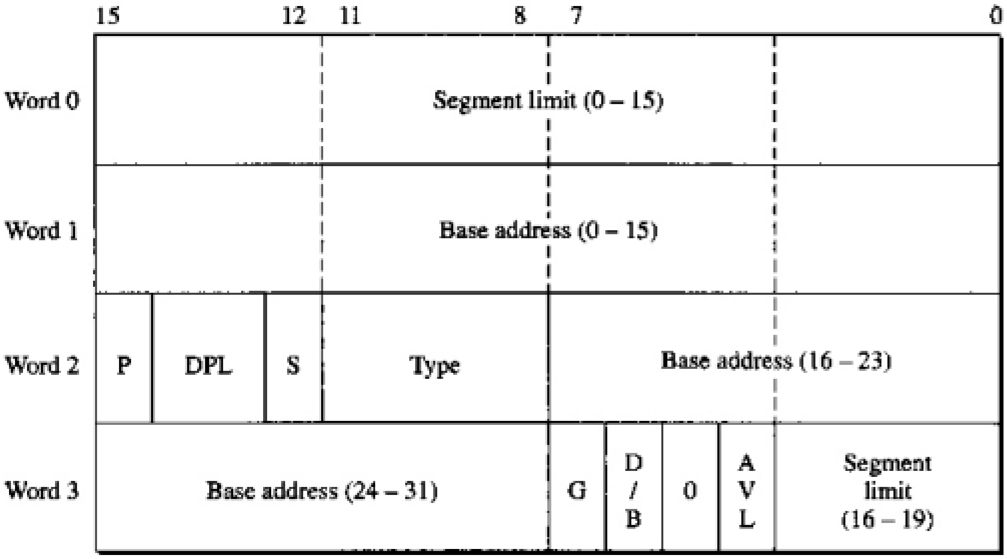
In protected mode, there are four privilege levels or rings, numbered from 0 to 3, with ring 0 being the most

Privileged and 3 being the least.

- The use of rings allows for system software to restrict tasks from accessing data, call gates or executing privileged instructions
- In most environments, the operating system and some device drivers run in ring 0 and applications run in ring 3.System Segment Descriptors

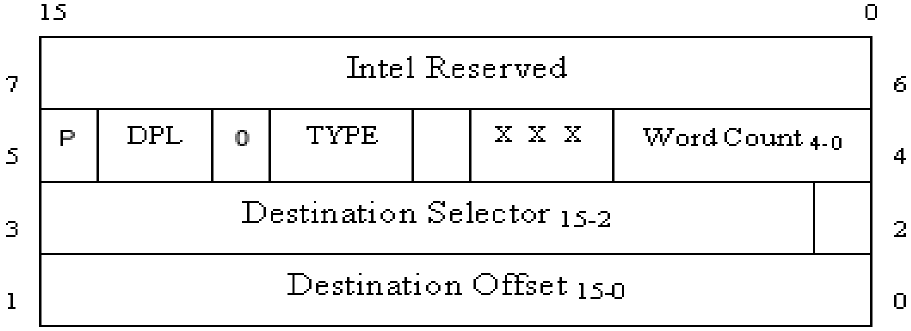
SYSTEM SEGMENT DESCRIPTORS (3M)

- In addition to code and data segment descriptors, the protected mode 80286 defines System Segment Descriptors. These descriptors define special system data segments which contain a table of descriptors (Local Descriptor Table Descriptor) or segments which contain the execution state of a task (Task State Segment Descriptor).
- Figure gives the formats for the special system data segment descriptors. The descriptors contain a 24-bit base address of the segment and a 16-bit limit. The access byte defines the type of descriptor, its state and privilege level. The descriptor contents are valid and the segment is in physical memory if P=1. If P= 0,the segment is not valid.



GATE DESCRIPTORS (2M)

- Gates are used to control access to entry points within the target code segment. The gate descriptors are call gates, task gates, Interrupt gates and trap gates. Gates provide al level of Indirection between the source and destination of the control transfer. This indirection allows the CPU to automatically perform protection checks and control entry point of the destination. Call gates are used to change privilege levels (see Privilege),task gates are used to perform a task switch, and interrupt and trap gates are used to specify interrupt service routines. The interrupt gate disables interrupts (resets IF) while the trap gate does not.



c) Explain the paging in 80386. Explain enabling and disabling paging.

Ans: **Memory Paging**: **(4M)**

Through paging, system software can restrict and control a task's access to pages, which are sections of memory. In many operating systems, paging is used to create an independent virtual address space for each task. This prevents one task from manipulating the memory of another

- Paging is one of the memory management techniques used for virtual memory multitasking operating system.
- The segmentation scheme may divide the physical memory into a variable size segments but the paging divides the memory into a fixed size pages.
- The segments are supposed to be the logical segments of the program, but the pages do not have any logical relation with the program.
- The pages are just fixed size portions of the program module or data.
- The advantage of paging scheme is that the complete segment of a task need not be in the physical memory at any time. Only a few pages of the segments, which are required currently for the execution, need to be available in the physical memory.
- Thus the memory requirement of the task is substantially reduced, relinquishing the available memory for other tasks. • Whenever the other pages of task are required for execution, they may be fetched from the secondary storage. • The previous page which are executed, need not be available in the memory, and hence the space occupied by them may be relinquished for other tasks.

Thus paging mechanism provides an effective technique to manage the physical memory for multitasking systems.

Paging Unit: The paging unit of 80386 uses a two level table mechanism to convert a linear address provided by segmentation unit into physical addresses. The paging unit converts the complete map of a task into pages, each of size 4K. The task is further handled in terms of its page, rather than segments. The paging unit handles every task in terms of three components namely page directory, page tables and page itself.

- The CR3 is used as page directory physical base address register, to store the physical starting address of the page directory.

The paging system operates in both real and protected mode.
It is enabled by setting the **PG** bit to 1 (left most bit in **CR0**).
(If set to 0, linear addresses are physical addresses).

Enabling and Disabling Paging on the 80386 **(4M)**

When the 80386 microprocessor is brought out of reset, it first executes in real mode. To use paging, the processor must be executing in protected mode. To enable paging, follow these steps:

1. Set up the page directory and the page tables in memory with the desired values.

2. Load CR3, the page directory base, with the base address of the page directory. Loading CR3 also invalidates any information stored in the TLB.

3. Execute a MOV CRO, EAX instruction where bit 31 is set to 1, and the other bits are unchanged. This can be accomplished with the sequence MOV EAX, CRO; OR EAX, 80000000H. It is possible to enable paging at the same time protected mode is entered. The instruction sequence in which the transition to paging will occur must have its linear address mapped to its physical address.

4. The instruction prefetch queue should be flushed by performing a JMP $2 instruction.

Once paging is turned on, all linear addresses are paged to the correct physical address. The address translation information is then automatically cached into the TLB each time the hardware performs a page translation from the tables in memory. Should you change any of the page table information in memory, or decide to use a different set of page tables, you must perform a MOV CR3, xxxxxxxx to invalidate the current TLB entries so that the new paging information will be used.

When disabling paging you must do the following:

1. Locate the instruction that performs the translation on a page whose linear address is the same as the physical address. This prevents an unpredictable instruction prefetch from occurring between changing the paging status and the next instruction.

2. A MOV CRO, EAX where bit 31 is forced to 0 is executed. This can be accomplished with the sequence MOV EAX, CRO: AND EAX, 7FFFFFFFH.

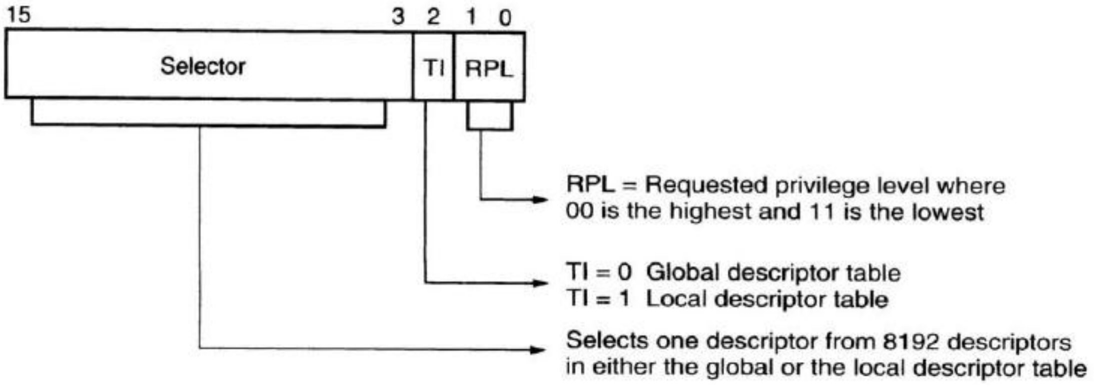
3. MOV CR3, EAX to invalidate the TLB entries.

After paging is disabled, the linear address and physical address are the same.

Q.3) Attempt Any Four of the Following 16M

a) Draw and Discuss selector in 80286 (2M for diagram and 2 M for Description)

Ans:


15
3
2
1
0
Selector
TI
RPL
RPL = Requested privilege level where 00 is the highest and 11 is the lowest
TI = 0 Global descriptor table
TI = 1 Local descriptor table
Selects one descriptor from 8192 descriptors in either the global or the local descriptor table

Selector is used to select the descriptor from 8K descriptor table either global or local. Using it processor can access virtual memory. It contains RPL, TI bits as shown in diagram

Bits	**Name**	**Function**
1-0	Request Privilege level(RPL)	Indicates selector privilege level required.
2	Table indicator (TI)	TI=0 use global descriptor table (GDT) TI=1 use local Descriptor table (LDT)
3-15	Selector	Selector descriptor entry in table

b) Describe the functions of following Pins (1M for each Pin)

PEREQ: This pin extends the memory management and protection capabilities of 80286 to processor extension. The PEREQ input requests the 80286 to perform a data operand transfer for a processor extension.

M/IO': It distinguishes memory access from IO access. When this pin is high memory access is performed while if this pin is low IO access is performed.

ERROR’: An Active ERROR’ input causes the 80286 to perform a processor extension interrupt when executing WAIT or some ESC instructions. This input is low and asynchronous to system clock and have internal pull up resistors.

BHE’: When BHE’ is low it indicates transfer or data on the upper byte of the data bus D_{8-15}

c) Distinguish between LDTR and GDTR (2M description and 2M for Diagram)

LDTR: Local Descriptor Table Register: System address register used to hold address of Local descriptor Table. It holds 16 bit selector for LDT

System Segment Register	Descriptor registers		
15 ⟷ 0	← → 32bitLinearbase address	← → 32 bit segment limit	← → Attributes
Selector			

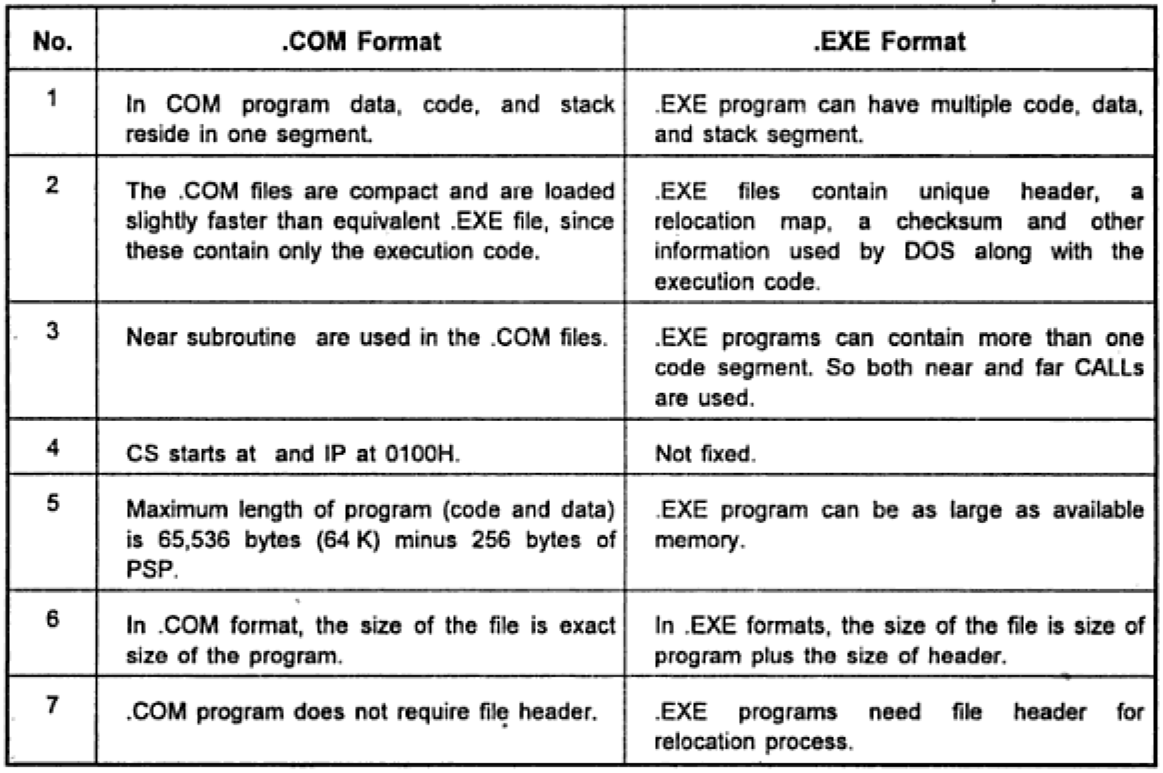
GDTR: Global Descriptor Table Register: System address register used to hold address of Global descriptor Table. It holds 32-bit linear base address and 16-bit limit of GDT

47 ⟷ 16 32 bit linear address	15 ⟷ 0 16 it limit

d) Differentiate between .COM and .EXE programs (Any four Points) (1M each Point)

Sr. No.	Parameters	.COM	.EXE
1.	Maximum size	64k – length of PSP (256 bytes) & mandatory word (2 byte)	No limit
2.	Entry point	PSP = 0100H	Derived by end statement
3.	AL at entry	00 if default FCB # 1 has valid drive. OFFH for invalid drive	00 if default FCB # 1 has valid drive. OFFH for invalid drive.
4.	AH at entry point	00 if default FCB # 2 has valid drive else OFFH – for invalid drive	00 if default FCB # 2 has valid drive else OFFH for invalid drive
5.	CS at entry point	PSP	Segment containing module with entry point
6.	IP at entry point	0100	Offset of entry point within it‟s segment
7.	DS at entry point	PSP	PSP
8.	ES at entry point	PSP	PSP
9.	SS at entry point	PSP	Segment with stack
10.	SP at entry point	OFFFEH or top word available in memory	Size of segment defined with stack attribute
11.	Stack entry	Zero word	Initialize or uninitialized
12.	Stack Size	64k – length of PSP (256 bytes) & Mandatory word (2 byte)	Defined in stack attribute
13.	Subroutine call	Unusually Near	NEAR OR FAR
14.	Exit Method	INT 21H – 4CH	INT 21H – 4CH
15.	Size of a file	Exact of program	Size of program & header

OR



No.	.COM Format	.EXE Format
1	In COM program data, code, and stack reside in one segment.	.EXE program can have multiple code, data, and stack segment.
2	The .COM files are compact and are loaded slightly faster than equivalent .EXE file, since these contain only the execution code.	.EXE files contain unique header, a relocation map, a checksum and other information used by DOS along with the execution code.
3	Near subroutine are used in the .COM files.	.EXE programs can contain more than one code segment. So both near and far CALLs are used.
4	CS starts at and IP at 0100H.	Not fixed.
5	Maximum length of program (code and data) is 65,536 bytes (64 K) minus 256 bytes of PSP.	.EXE program can be as large as available memory.
6	In .COM format, the size of the file is exact size of the program.	In .EXE formats, the size of the file is size of program plus the size of header.
7	.COM program does not require file header.	.EXE programs need file header for relocation process.

d) List all general purpose registers of 80238 and state their functions
(2M Listing and 2M for Functions)

This question has a printing mistake it should have been 80286 instead of 80238 if students have assumed 80286 and written the answer marks can be allocated to the students for correct answer.

In 80286 general purpose register i.e AH, AL, BH, BL, CH, CL, DH & DL. These register can be used as 8 bit registers individually or can be used as a 16 bit in pair to have AX, BX, CX& DX.

AX is called as 16 Bit Accumulator
BX is called as 16 Bit Base Register
CX is called as 16 Bit Counter Register
DX is called as 16 Bit Counter Register

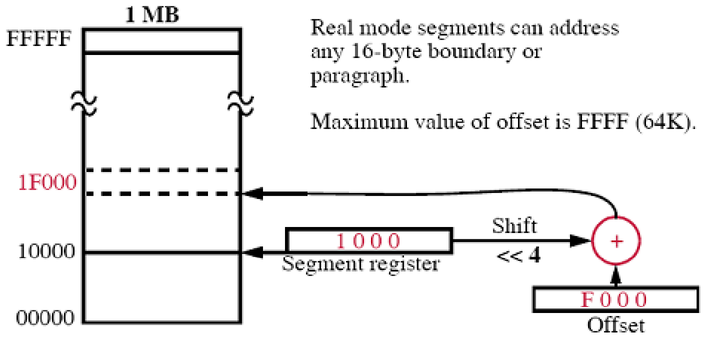
4a) Attempt any Three of the following 12M

i) State the different operating modes of 80286 processor. Describe real mode in brief.
(1M for listing 3M for explaining Real Address Mode)

Ans.

The operating modes of 8086 are 1) Real Address Mode 2) Protected virtual Address Mode

The real mode operation allows up to address only 1 MB memory space. This 1 MB memory is called real memory a conventional memory. In real addressing mode it operates in Dos environment it uses combination of segment address & offset address to access 1 MB memory i.e. segment address located within are of the segment register defines beginning address of 64 kb memory segment.
The offset address selects any location within 64 kb memory segment offset. Sometimes it is also known as displacement.
Figure shows the example of memory segment beginning at memory location 10000 H & ends at 1FFFF H i.e. 64 kb. These segment registers hold the starting address of segment i.e. 1000 & 0 is added internally by SHIFT operation of address generation unit (AGU). The offset i.e. F000H is also added to generate physical address or effective address 1F000H.

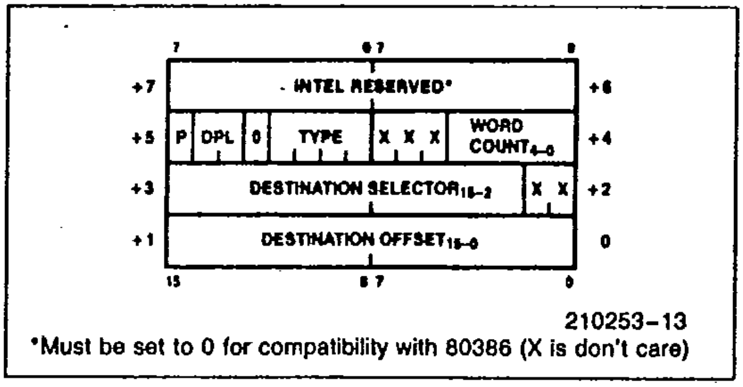

1 MB
FFFFF
1F000
10000
00000
Real mode segments can address any 16-byte boundary or paragraph.
Maximum value of offset is FFFF (64K).
1 0 0 0
Segment register
Shift
<< 4
+
F 0 0 0
Offset

ii) With Neat Diagram explain Gate Descriptor of 80386 Microprocessor.

(2M for Diagram and 2M for Explanation)

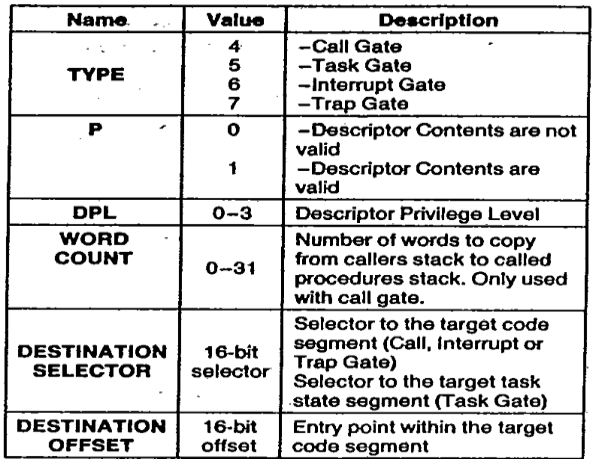
Ans:

Gate Descriptor



210253–13

*Must be set to 0 for compatibility with 80386 (X is don't care)



Name	Value	Description
TYPE	4 5 6 7	–Call Gate –Task Gate –Interrupt Gate –Trap Gate
P	0 1	–Descriptor Contents are not valid –Descriptor Contents are valid
DPL	0–3	Descriptor Privilege Level
WORD COUNT	0–31	Number of words to copy from callers stack to called procedures stack. Only used with call gate.
DESTINATION SELECTOR	16-bit selector	Selector to the target code segment (Call, Interrupt or Trap Gate) Selector to the target task state segment (Task Gate)
DESTINATION OFFSET	16-bit offset	Entry point within the target code segment

iii) What is interrupt vector table describe in brief.

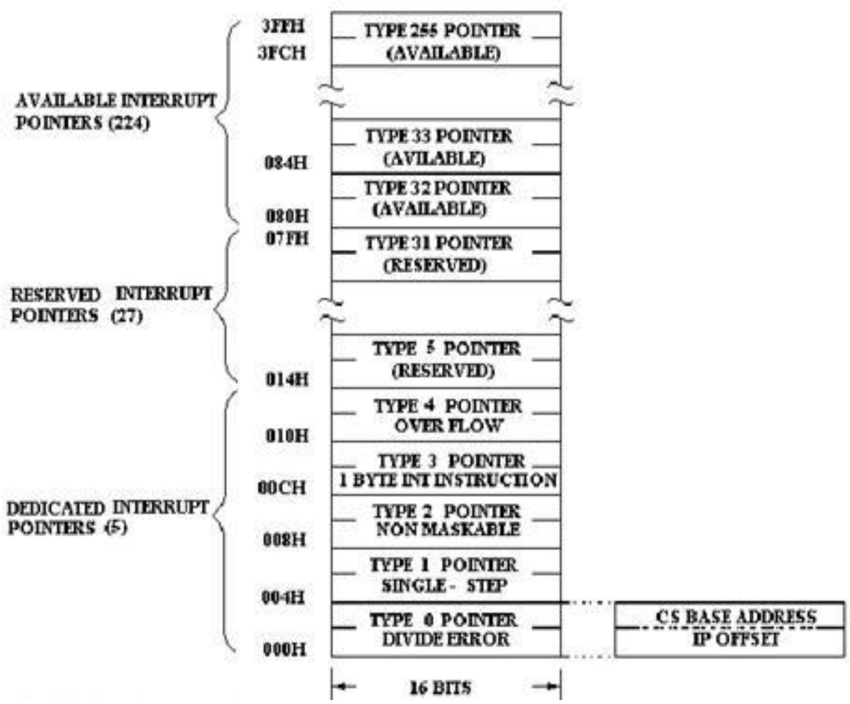
Ans: **(2M Diagram and 2M for Explanation)**

In 8086 1 kb from 00000 to 003ff are reserved for interrupt routine as shown in figure known as interrupt vector.

It supports 256 interrupt procedures containing 16 bit IP & CS address.

The type 0- 4 are predefined or dedicated interrupts. Types 5 – 31 are reserved for Intel. Type 32 – 225 is available interrupt or user defined interrupts.

It fetches the ISR from IT. The INTn instruction calls the ISR (procedure) that begins at the address represented in vector number ‘n’


3FFH
3FCH
TYPE 255 POINTER (AVAILABLE)
AVAILABLE INTERRUPT POINTERS (224)
TYPE 33 POINTER (AVILABLE)
084H
TYPE 32 POINTER (AVAILABLE)
080H
07FH
TYPE 31 POINTER (RESERVED)
RESERVED INTERRUPT POINTERS (27)
TYPE 5 POINTER (RESERVED)
014H
TYPE 4 POINTER OVER FLOW
010H
TYPE 3 POINTER 1 BYTE INT INSTRUCTION
00CH
DEDICATED INTERRUPT POINTERS (5)
TYPE 2 POINTER NON MASKABLE
008H
TYPE 1 POINTER SINGLE - STEP
004H
TYPE 0 POINTER DIVIDE ERROR
000H
CS BASE ADDRESS
IP OFFSET
16 BITS

iv) Distinguish between software and hardware interrupts. (Any two points two mark each)

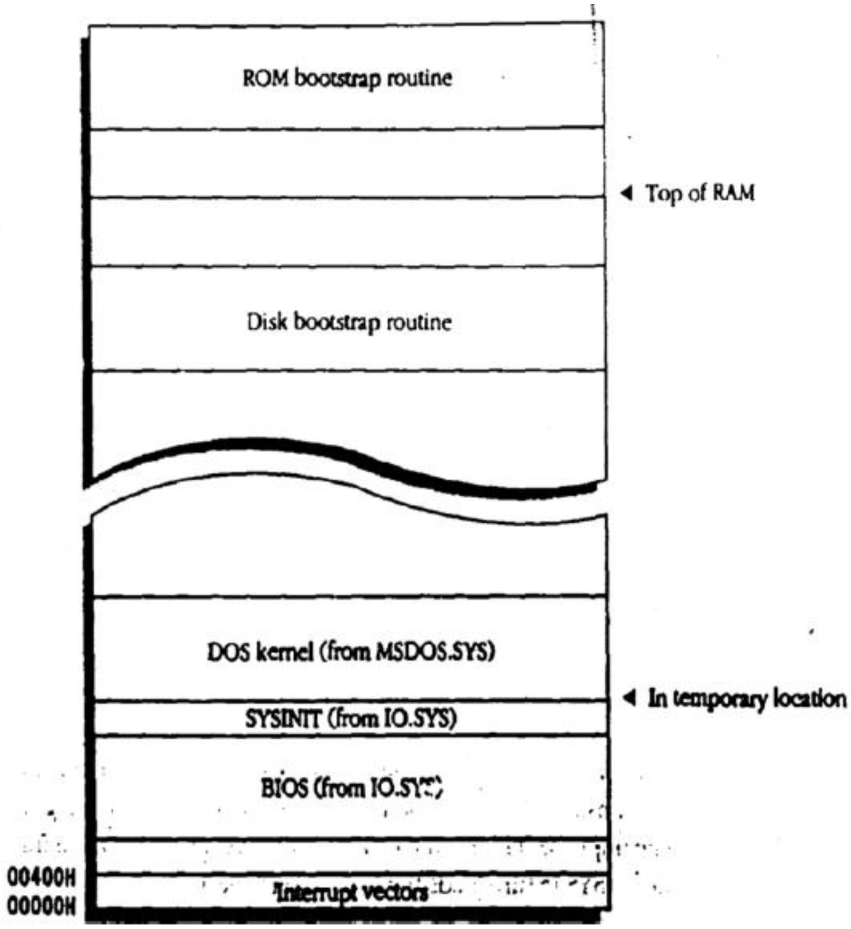
Sr. No	Software INT	Hardware INT
1	Interrupt generated by execution of instruction INT or some sort of software programs	External events occur on hardware pins on microprocessor dedicated pins and causes interrupt.
2	INT n, INTO, INT3 & BOUND are known as software interrupts.It provides the warning messages.	External signal applied on NMI or INTR pin is known as hardware interrupt.
3	Software Interrupts are Maskable	Hardware Interrupts can be maskable or Non Maskable

b) Attempt any one of the following. 6M

i) Illustrate loading of MSDOS into memory with graphical view.

Ans: **(2 Mark Diagram and 4 Mark Explanations)**

- Reset a start of the system points the IP = OFFFFFOH (feature X 86 family). The IP jumps to system test code at ROM boot strap routine.
- ROM Boot Strap reads Disk Boot Strap from the first sector of the system startup disk & control transfer to it.
- The disk boot strap routine checks the copies of MS – DOS by reading first sector of boot
- Directory & determining two files, IO.SYS & MS – DOS.SYS. (a IBMB10.com & IBMDOS.com)
- If files are not present user is prompted to change disk & strike any key to try again.
- If two files are present boot strap reads them into memory & transfers control to the initial entry point of IO.SYS
- IO.SYS consists of two separate modules.
- The first module is BIOS which contains of link set of resident device drivers.
- The second module is a SYS.INIT i.e. initialization code. It determines the amount of
- Contiguous memory presents in the system & then relocates itself to high memory & DOS kernel is loaded into the final memory.
- This SYS.INIT file calls the initialization code i.e. MSDOS.SYS for device driver & setup the vectors for any external h/w interrupt service.
- DOS kernel checks disk parameter block.
- The entire CONFIG.SYS file is loaded into memory for processing.
- After all installable device drivers have been loaded SYS.INIT closes all file handle & reopen to console printer & devices or standard i/o devices & calls MSDOS.EXE function to load command interpreter or shell



ii) **State function of GDT and LDT in 80286. What is the size of each descriptor? How many table each descriptor store.**

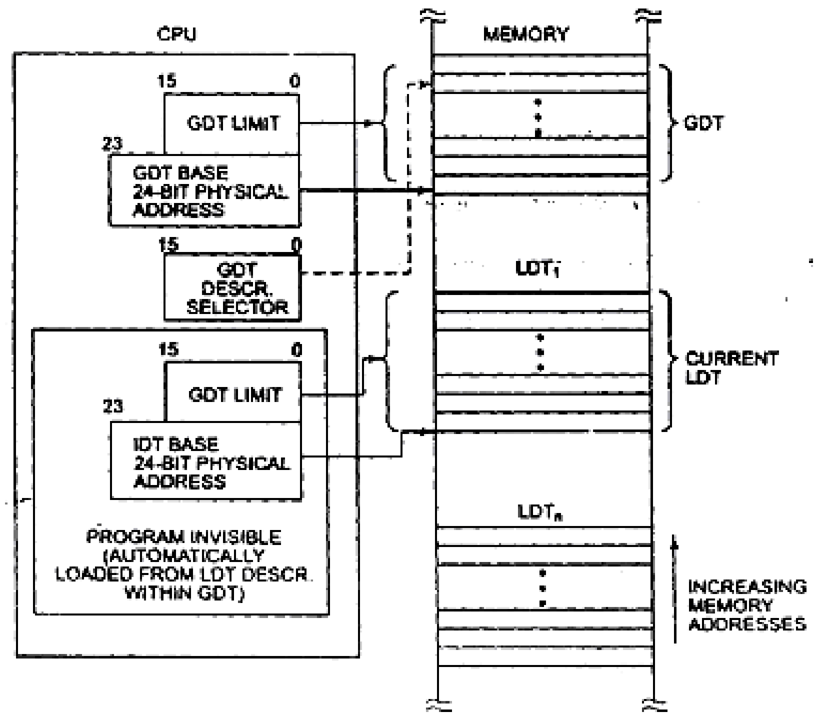
Ans: **(2 M Diagram and 4 M for Explanation)**

Two tables of descriptors, called descriptor tables contain all descriptors accessible by a task at any given time. A descriptor table is a linear array of up to 8192 descriptors. The upper 13 bits of the selector value are an index into a descriptor tale.

Each Table has 24 bit base register to locate the descriptor table in physical memory and a 16-bit limit register that confine descriptor access to the defined limits of the table as shown in fig. a restart able exception (13) will occur if attempt is made to reference a descriptor outside the table limits.

One table called as the global descriptor Table (GDT) contains descriptors available to all tasks. The other table, called the Local descriptor Table (LDP), contains descriptors that can be private to a task. Each task may have its own private LDT. The GDT may contain all descriptor types except interrupt and trap descriptors. The LDT may contain only segment, task gate, and call gate descriptors. A segment cannot be accessed by a task if its segment descriptor does not exist in either descriptor tale at the time of access.

Sr. No	LDT	GDT
1	It is associated with Local descriptor Table	It is associated with Global descriptor Table
2	It is available for single task.	It is available for all tasks.
3	It is mainly for Code, data, stack, task and call gate descriptor	It contains any type of segment except interrupt servicing.
4	It is used for isolation of task	It is used by operating system

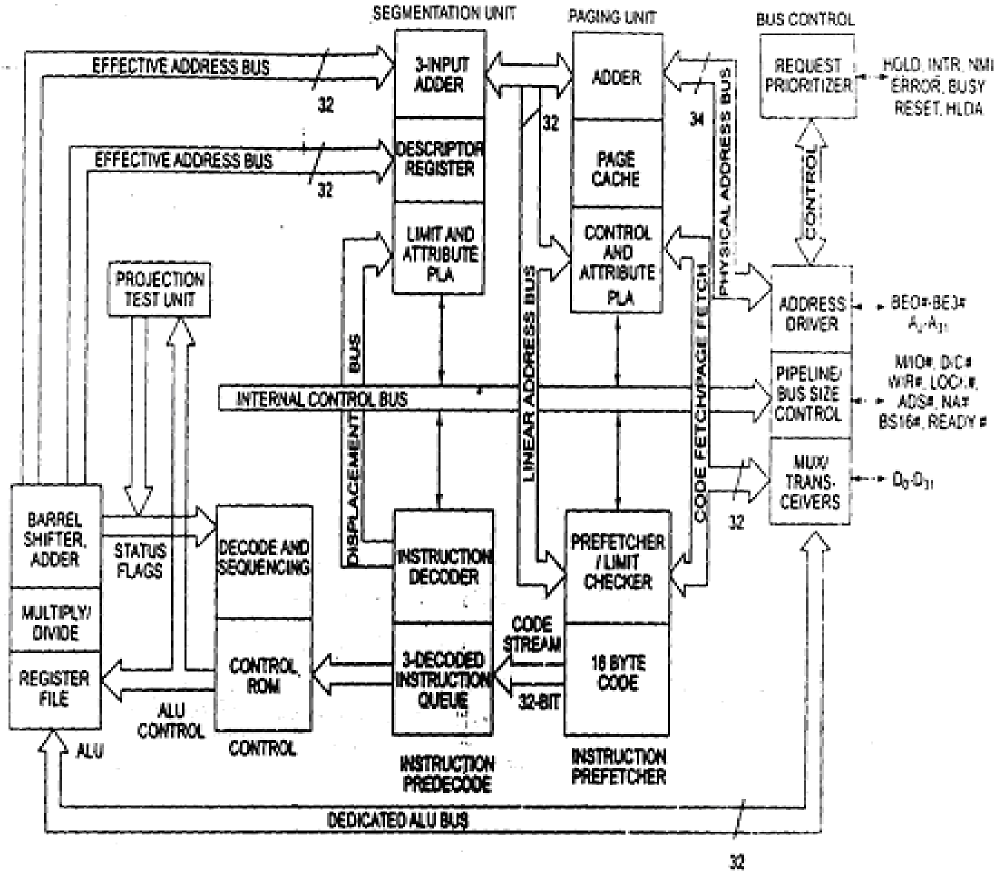


Q5 Attempt any two of the following: 16M

a) Draw the internal architecture of 80386. Also explain the following pin of 80386

- **BE0 to BE3**
- **D/$\overline{\text{C}}$**
- **$\overline{\text{LOCK}}$**
- **$\overline{\text{BUSY}}$**

Ans: (Architecture with label = 4 M each pin functionality 1M X= 4M)



- BE0 to BE3 – Output - It selects the access of byte, word, double word of data. These signals are generated by A_0 & A_1 used to validate the data
- D/C ¯ Output - Whenever it is 'I', it indicates the data bus contains data & when '0' up is in halt state or executes interrupt acknowledgement.
- Lock – Output - The o/p signal indicates that other system bus master will be prevented for gaining control of system bus.
- Busy- Input - This signal is sent by co – processor. It is active low signal causing 80286 to wait or escape instruction

b) Explain with the neat diagram structure of MS DOS. Draw and explain the format of 8086 hardware interrupt.

Ans**: (MS DOS structure 1 M, description 1 M for each layer Interrupt format 2 M, Description 2 M)**

The MS – DOS is partitioned into several layers to isolate the kernel logic of the OS, user perception & hardware.

The layers are as follows:

1) The BIOS (Basic Input Output System)

2) DOS kernel

3) The Command Processor (shell)

1) BIOS :-

It is residual program provided by the manufacturer. Specific BIOS is available for individual computer & provided by manufacturer. It contains default resident hardware driver for :

1) Console display & keyboard (CON)
2) Line printer (PRN)
3) Auxiliary Device (AUX)
4) Date & Time (Clock)
5) Boot Disk Device (block device)

2) DOS - Kernel :-

The Kernel is a proprietary program supplied by Microsoft corporation & provides collection of hardware independent services on system function such as :

1) File & Record Management

2) Memory Management

3) Character Device I/O

4) Spawning of other program

5) Access to the real time clock

3) Command Processor :-

Intrinsic command → internal command

Extrinsic command → external command

It is the user interface to the operating system. It is responsible for carrying out the commands i.e. loading or execution of program from the disk or mass storage device.

The default shall command is command.com. It is a program under the control of MS – DOS. It is fetcher divided into:

A resident portion – at low portion of memory.

Initialization section – at middle portion of the memory.

Transient Module – at the high end of the memory.

Command Processor
DOS Kernel
BIOS

Hardware Interrupt

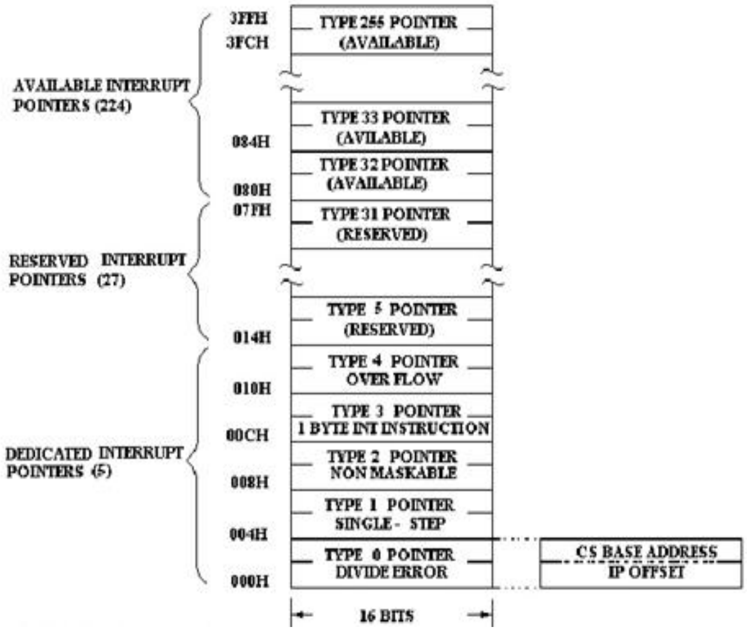
(Note: As such there is no specific format for interrupt but the default vector table is available containing address of Interrupt service routine . Hence student may draw this as given below.)

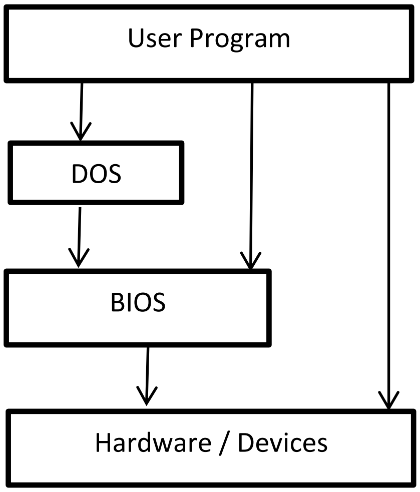
Hardware Interrupt:

There are two types of hardware interrupts are available for 8086.

None mask able Interrupt (NMI) Type 2. : This has the highest priority to provide the service . It responds run NMI pin receives low to high transition (+ve edge trigger). Processor enters in NMI routine & remains still NMI signal is disabled. IP = 00008 & CS = 00010.

Mask able Interrupt INTR: It is used for asynchronous external hardware events. Interrupt occurs when INTR is pulled high and interrupt flag is enabled. Processor reads the 8 bit vector supplied by hardware which identifies the source of interrupt from user define category.


AVAILABLE INTERRUPT POINTERS (224)
RESERVED INTERRUPT POINTERS (27)
DEDICATED INTERRUPT POINTERS (5)
3FFH
3FCH
084H
080H
07FH
014H
010H
00CH
008H
004H
000H
TYPE 255 POINTER (AVAILABLE)
TYPE 33 POINTER (AVILABLE)
TYPE 32 POINTER (AVAILABLE)
TYPE 31 POINTER (RESERVED)
TYPE 5 POINTER (RESERVED)
TYPE 4 POINTER OVER FLOW
TYPE 3 POINTER 1 BYTE INT INSTRUCTION
TYPE 2 POINTER NON MASKABLE
TYPE 1 POINTER SINGLE - STEP
TYPE 0 POINTER DIVIDE ERROR
CS BASE ADDRESS
IP OFFSET
16 BITS



c) What is five stage pipeline mechanism ?Draw Pentium system architecture

Ans: (Pipeline mechanism 4 m Pentium architecture 4 M)

Pentium microprocessor issues 2 instructions in parallel to the 2 independent integer pipelines u & v. Hence known as multiple instruction issue. It has 5 stage pipelines operated in parallel allowing integer instruction to execute in single clock in each pipelining.

Pipeline Stages:-

1) **Prefetch Stage (PF) :-**

There are two prefetch buffer or queue present in Pentium & at a time one of them is active to fetch instruction code from on chip cache or memory. In this stage CPU aligns the port appropriately because instructions are in variable length & given to decode stage DI.

2) **Decode 1 Stage (D1) :-**

In DI stage CPU decodes the instruction & generates a control word by following some rules for instruction pairing. The two instructions are parable only if they are simple & register independent.

3) **Decode 2 Stage (D2) :-**

In this stage control word from DI is decoded & memory addresses are generated for the data.

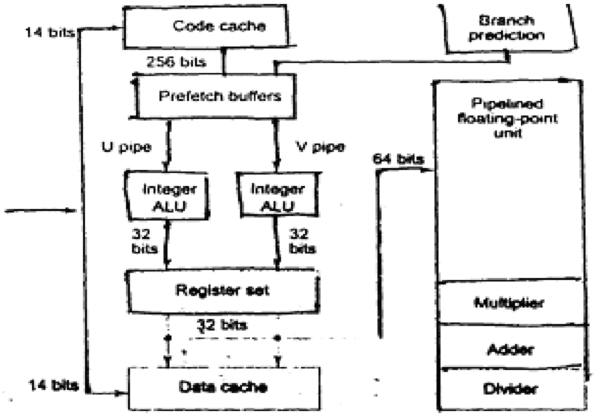
4) **Execution Stage (EX) :-**

In this stage CPU either access data cache for data operand or executes arithmetic logical function or floating point operation.

5) **Write Back Stage (WB) :-**

This is the final stage of integer instruction execution & CPU updates the target registers & E – flag register.

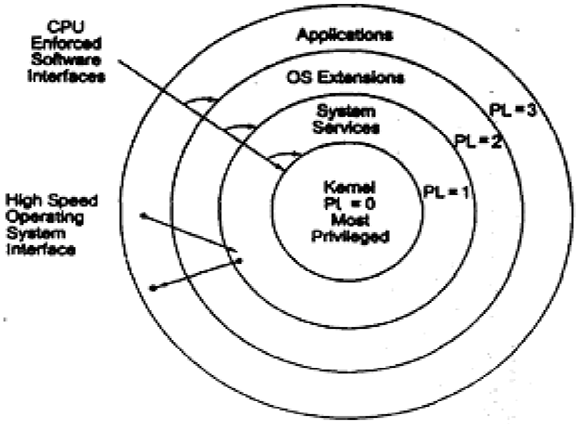
Pentium system architecture



Q6 Attempt any Four of the following 16M

a) **Illustrate with the diagram four level privilege mechanism in 80286 microprocessor**

Ans: (Diagram 1Marks Each Type 1 MarksX3 = 3M)


CPU Enforced Software Interfaces
Applications
OS Extensions
System Services
PL=3
PL=2
PL=1
Kernel PL = 0 Most Privileged
High Speed Operating System Interface

Note: PL Becomes Numerically Lower as Privilege Level Increases

It supports four level privilege mechanism levels to control the access of descriptor & hence corresponding segment of the task. This prevents the unwanted access to any code or data segment.

The operating system interrupt handlers or other system software can be protected from unauthorized access in virtual address space of each task. The fig. shows the mechanism. It has 3 types : 1) Task Privilege 2) Descriptor Privilege & 3) Selector Privilege.

1) **Task Privilege :-**

Each task has a privilege level to indicate its priority & known as current privilege level (CPL). It is defined by 2 LSB bytes of code segment register. The CPL is changed by transferring the control using gate descriptor to a new segment.

2) **Descriptor Privilege :-**

It is defined by DPL field of Access right bytes. The DPL specifies the list task privilege level that may be used to refer the descriptor.

3) **Selector Privilege :-**

This is defined by RPL of selector. RPL is less trusted privilege than CPL. This is also known as effective privilege level (EPL) of the task.

b) Differentiate between virtual mode and real mode of 80286

Ans: **(Any four each carry 1mark X 4= 4 marks)**

Sr No.	**Virtual Mode**	**Real Mode**
1	It Access only 1 GB memory	It Access only 1 MB memory
2	It uses Memory Management unit to calculate physical address.	It uses Address Generation unit to calculate physical address.
3	It supports the protection Mechanism	No Protection mechanism is available
4	It uses A23 to A0 address lines	It uses A19 to A0 address lines
5	It uses full segment of 64 K	It does not uses full segment of 64 K

c) What do you meant by TLB? How does it help the address calculation procedure?

Ans: **(2M diagram and 2M Explanation)**

Translation look aside buffer: It is a cache of most recently accessed pages. It is four way set associate 32 entry page table

TLB holds page table address translation to reduce the no. of memory required for page table translation.

Address Calculation procedure

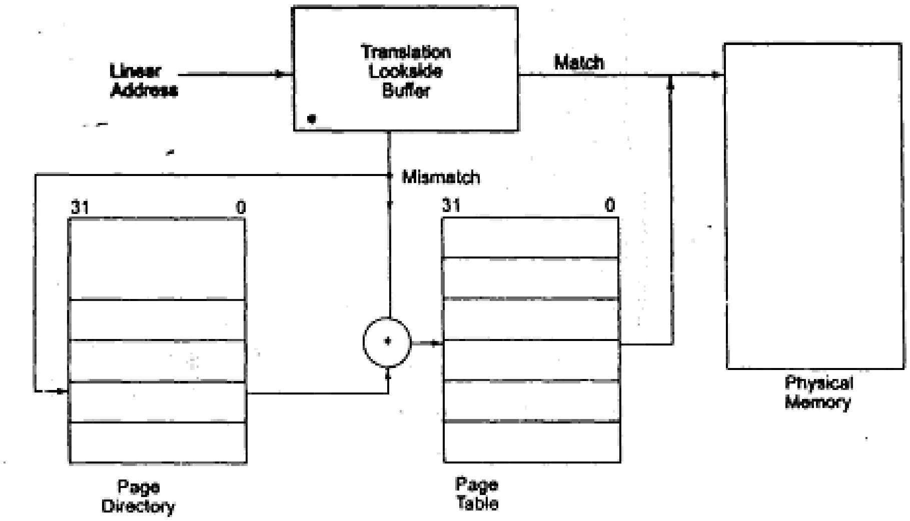


Figure for address calculation

The paging unit receives 32 bit linear address from segmentation unit. The upper 20 bit ($a_{31} – a_{12}$) gives page directory & path table address bits, they are compared with all 32 bit entries in TLB. If it matches physical address is calculated from match TLB entries & place on address bus.

To speed up the conversion process of linear address to physical address. Cache of 32 X 4 byte is provided to store a 32 bit page table entry that is known as TLB (Translation Loop aside Buffer).

If page table entry is not in TLB 80386 reads to appropriate page directory entry & set or reset 'P' & 'A' bit.

d) Differentiate between 80286 and 80386

Ans: (Any four each carry 1mark X 4= 4 marks)

	Parameter	**80286**	**80386**
1.	Address Bus	It has 24 bit address bus.	It has 32 bit address bus.
2.	Data Bus	It has 16 bit data bus.	It has 32 bit data bus.
3.	MMU	It support MMU	It contains inbuilt MMU
4.	Operating Speed	8, 10, 12 MHz is the operating speed.	5, 20, 25 MHz is the operating speed.
5.	Flags	It has 11 active flags.	It has 13 active flags.

e) State the function of branch prediction unit in Pentium processor. To what extent is the efficiency increased due to this unit?

Ans: (Function =2 M Efficiency =2 M)

Branch Prediction Unit: The Pentium processor includes branch prediction logic to avoid pipeline stalls. It is used minimize the effect of control hazards.

Efficiency:

If correctly, predict whether or not branch will be taken when branch instruction is executed. If branch prediction is not correct recycle penalty is applicable to u pipeline & 4 cycle penalty if branch is related to v pipeline. Hence efficiency depends on the pipeline and its prediction. The efficiency will increase if the prediction is correct.