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. Attempt any five of the following **20M**

a) Explain the interrupts 0f 80286 **4M**

Ans: **[Explanation of any four instructions -4 Marks]**

1. Instruction exceptions
2. Single step
3. NMI
4. Processor extension segment overrun
5. INTR
6. INT n

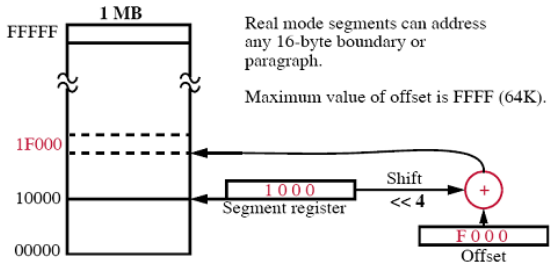
b) Explain the real addressing mode of 80286 **4M**

Ans: **[Diagram 2M, Description 2M]**

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.



c) List any four salient features of INTEL 80386 **4M**

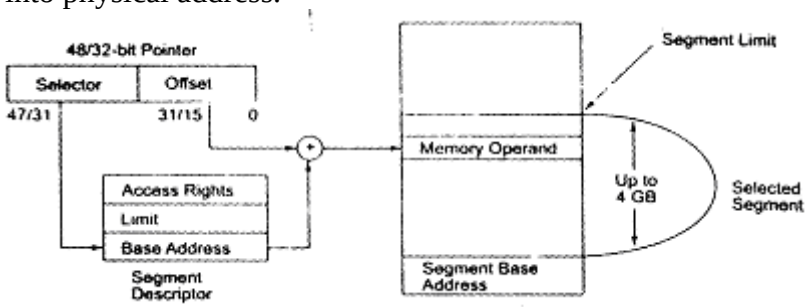
Ans: **[Any four features-4M]**

1. It is 32 bit processor.
2. It access 4 GB memory in real mode. (2^{32} = 4 GB)
3. It access 64 TB memories in virtual mode.
4. It contains inbuilt Memory Management Unit (MMU).
5. It works on frequency 5, 20 & 25 MHz.
6. There is 32 bit flag register but only 13 flags are active.
7. It supports DOS unique environment.
8. It is designed in fully pipelined architecture & has three queues for fetch, decode & execute.
9. It is available in 132 pins PGA (Pin Green Array) package.
10. It also offers a set of total 8 debug registers $DR_0 – DR_7$ for hardware debugging & control.
11. It has on chip address translation cache.
12. It is also available in 3 versions :
    i. 80386 SX → 16 bit up (286)
    ii. 80386 DX → full 32 bit
    iii. 80386 EX → full 32 bit + on chip DRAM Controller
13. Register size is 32 bit. It can access 8 bit, 16 bit & 32 bit data.

d) Explain protected virtual addressing mode of 80386 **4M**

Ans: **[Diagram 2M, Description 2M]**

In this mode 80386 can address 4 GB of physical memory. In this mode contents of segment register are used as selector address descriptor containing segment base address & access right bytes of a segment.
The offset is added within segment base address to calculate linear address & feature used as physical address. The paging unit works in this. The large segment of a memory is divided into 4 kb size pages, the paging unit converts linear address into physical address.



e) Draw and explain the interrupt vector table of 8086 microprocessor **4M**

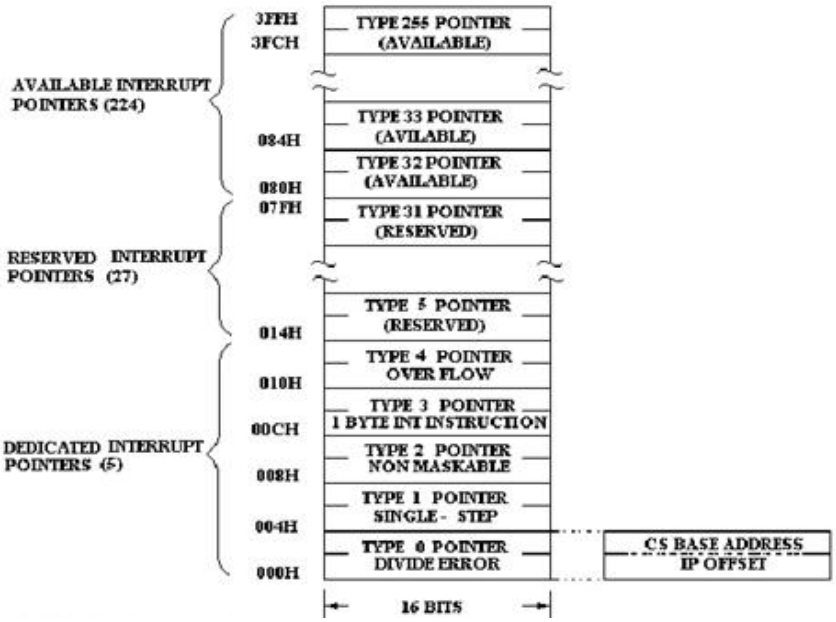
Ans: **[2M for diag, 2M for expl.]**

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. Type 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'.



f) List any four important features of Pentium processor. 4M

Ans: (Any four feature 1 Mark each)

- Compatible with Large Software Base, MS-DOS*, Windows*, OS/2*, UNIX
- Separate Code and Data Caches, 8-Kbyte Code, 8-Kbyte Write Back Data, MESI Cache Protocol
- Advanced Design Features, Branch Prediction
- Functional Redundancy Checking Support
- Internal Error Detection Features
- Multi-Processor Support, Multiprocessor Instructions, Support for Second Level Cache
- On-Chip Local APIC Controller
- Upgradable with a Pentium® OverDrive® Processor Power Management Features
  System Management Mode
  Clock Control
- 32 bit superscalar and super pipelined architecture CISC processor
- Communicates with the outside word via a 32 bit address bus can address upto 4G bytes of physical memory
- 64 bit data bus so arithmetic and logical operation can be perform on 64 bit operand.
- 8KB instruction cache is used to provide quick access to frequently used instructions
- Pentium uses technique called branch prediction to maintain a steady flow of instructions into the pipelines.
- Two integer pipeline U & V.
- Support five stage pipeline.
- Two ALUs
- Onchip floating point coprocessor
- Support power management feature.
- Onchip memory management unit.
- Support multiprogramming and multitasking.

g) Distinguish between 80486 and Pentium Processor **4M**

Ans: **[Any four Points-4M]**

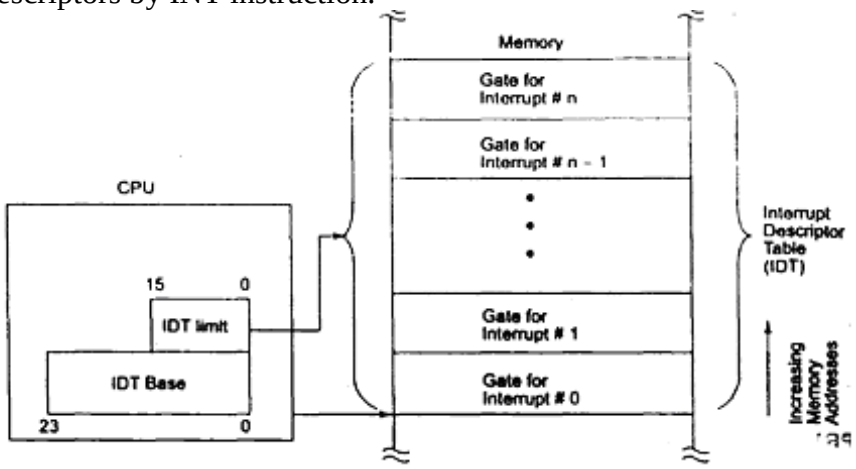
Sr. no.	Parameter	80486	Pentium
1.	Data Bus	It has 32 bit data bus.	It has 64 bit data bus.
2.	Architecture	RISC	Super scalar
3.	Operating Speed	100 MHz is the operating speed.	300 MHz is the operating speed.
4.	Cache	On board 8KB cache	Separate cache for data and code 8 KB
5.	Pipelining	5 stage pipe lining	8 stage Dual pipelining

Q2. Attempt any two of the following. **16M**

a) Explain IDT of 80286 **8M**

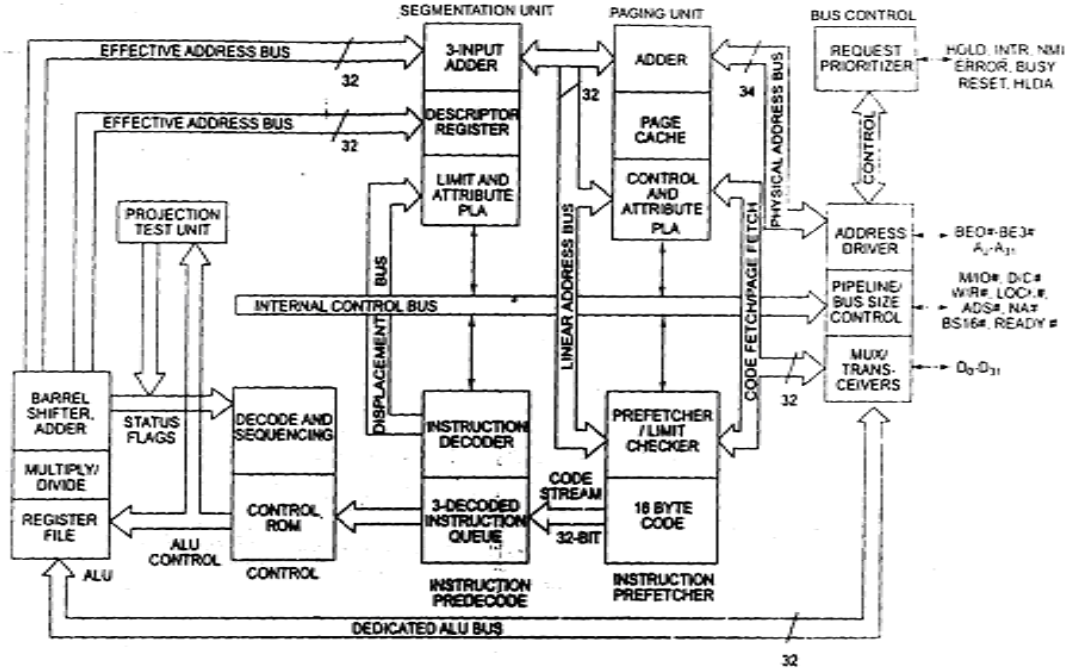
Ans: **[4M for diag, 4M for expln.]**

IDT are used to store task gates, interrupt gates & trap gates. The IDT has 24 bit base address & 16 bit limit register present in the CPU as shown in figure. Load interrupt Descriptor Table or LIDT is the instruction to load these internal register with 6 byte values & similar to Load Global Descriptor Table or LGDT. It handles 256 interrupt descriptors by INT instruction.



b) Draw and explain the internal architecture of 80386 8M

Ans: [4M for diag, 4M for expln.]



The figure shows the architecture of 80386. It is divided in 3 sections
1) MMU 2) CPU 3) BIU

1) Central Processing Unit :-

It consist of execution unit containing 8 General purpose register, 8 special purpose register used for data handling & calculation of offset address.

Central unit also consists of instruction unit of which decode & op code bytes received from 16 byte code queue & arrange them into 3 byte instruction decoded queue. It also consists of ALU containing barrel shifter.

2) MMU :-

It consists of segmentation & paging unit. Segmentation unit allows maximum 4 GB segments. Paging unit organizes physical memory in the form of pages of 4kb size. The segment unit & paging unit work together for MMU to access virtual memory to provide protection mechanism.

3) BIU :-

It contains request prioritizes, address driver, pipeline bus size control, MUX trans receiver to deal with memory of i/o device. This unit is responsible to access M/IO devices.

c) Explain the protection in 80386 in details 8M

Ans:

The purpose of the protection features of the 80386 is to help detect and identify bugs. The 80386 supports sophisticated applications that may consist of hundreds or thousands of program modules. In such applications, the question is how bugs can be found and eliminated as quickly as possible and how their damage can be tightly confined. To help debug applications faster and make them more robust in production, the 80386 contains mechanisms to verify memory accesses and instruction execution for conformance to protection criteria. These mechanisms may be used or ignored, according to system design objectives.

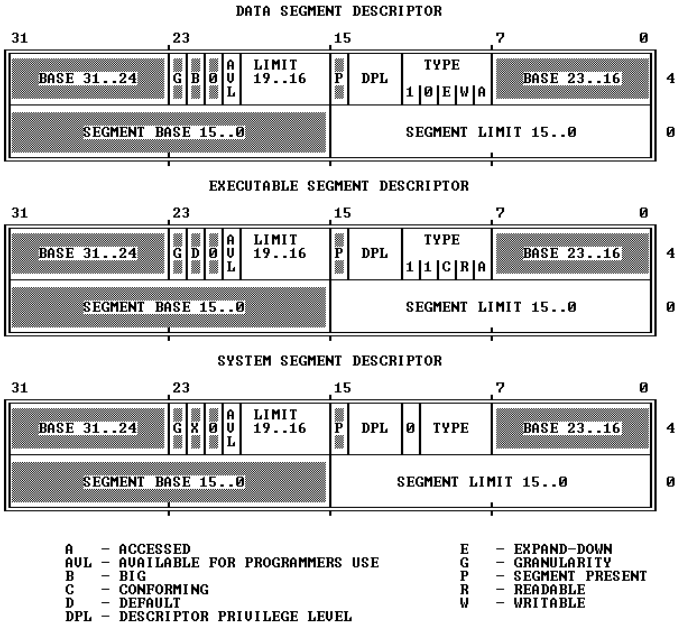
Protection in the 80386 has five aspects:

1. Type checking
2. Limit checking

3. Restriction of addressable domain
4. Restriction of procedure entry points
5. Restriction of instruction set

The protection hardware of the 80386 is an integral part of the memory management hardware. Protection applies both to segment translation and to page translation

Segment Level Protection



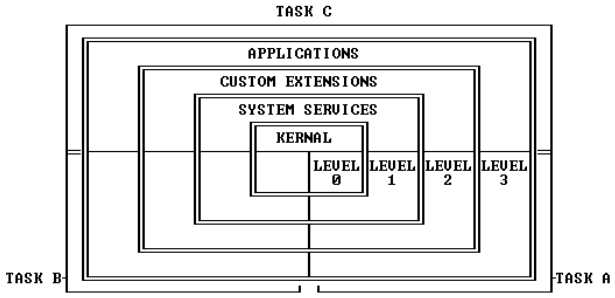
The TYPE field of a descriptor has two functions:

1. It distinguishes among different descriptor formats.
2. It specifies the intended usage of a segment.

The limit field of a segment descriptor is used by the processor to prevent programs from addressing outside the segment. The processor's interpretation of the limit depends on the setting of the G (granularity) bit. For data segments, the processor's interpretation of the limit depends also on the E-bit (expansion-direction bit) and the B-bit (big bit)

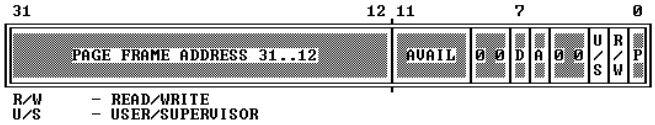
The concept of privilege is implemented by assigning a value from zero to three to key objects recognized by the processor. This value is called the privilege level. The value zero represents the greatest privilege, the value three represents the least privilege. The following processor-recognized objects contain privilege levels:

- Descriptors contain a field called the descriptor privilege level (DPL).
- Selectors contain a field called the requestor's privilege level (RPL). The RPL is intended to represent the privilege level of the procedure that originates a selector.
- An internal processor register records the current privilege level (CPL). Normally the CPL is equal to the DPL of the segment that the processor is currently executing. CPL changes as control is transferred to segments with differing DPLs



Page Level Protection
Two kinds of protection are related to pages:
Restriction of addressable domain.

1. Type checking.



Restricting Addressable Domain
The concept of privilege for pages is implemented by assigning each page to one of two levels:

1. Supervisor level (U/S=0) -- for the operating system and other systems software and related data.
2. User level (U/S=1) -- for applications procedures and data.

The current level (U or S) is related to CPL. If CPL is 0, 1, or 2, the processor is executing at supervisor level. If CPL is 3, the processor is executing at user level.
When the processor is executing at supervisor level, all pages are addressable, but, when the processor is executing at user level, only pages that belong to the user level are addressable.

Type Checking
At the level of page addressing, two types are defined:

1. Read-only access (R/W=0)
2. Read/write access (R/W=1)

When the processor is executing at supervisor level, all pages are both readable and writable. When the processor is executing at user level, only pages that belong to user level and are marked for read/write access are writable; pages that belong to supervisor level are neither readable nor writable from user level.

Q3. Attempt any four of the following. 16M

a) Draw the register structure of 80286 and explain function of each 4M

Ans:

(Note: There is no such diagram of register structure. Hence table showing list of register is the structure carries 1 mark, 3 marks for information)

General purpose Register
Segment register and its descriptor
Flag/MSW
Index and pointer
Instruction Pointer

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.
Segment Register:-

There are 4 segment register of 16 bit each. The code segment CS register is used for addressing a memory location in the CS of the memory, where the executable program is stared. The data segment DS register ports to the DS of memory where the data is stored. The ES also refers to a segment, which in another data segment of memory. Thus ES also contains the data. The stack segment is that segment of memory which is used to store the stock data. The CPU uses the stack for temporarily storing important data eg. The contents of the CPU register, which will be required in the last stage.
Index & Pointer:-

SP & BP are pointer register which hold 16 bit offset within particular segment. SI & DI are the index registers. The register SI if generally used to store the offset of source data or string in data segment while the

register DI is used to store the offset of destination in data or extra segment. The index register are particularly used for string manipulation.

Instruction Pointer :-

The instruction pointer register holds the address of the next code byte within the code segment. The value stored in IP is called as offset or displacement.

MSW/Flag Register :-It gives the status of operation

Descriptor Register: There are special types of descriptors that carry out additional function like transfer of control & task switching.

b) Explain pin function of 80286 4M

a) READY b) COD/INTA c) BUSY and ERROR d) BHE

Ans: [1M for each]

Symbol/Pin	I/O status	Description
READY#	Input	It is acknowledgement from i/o to memory when high, it indicates that peripheral device is ready to transfer data bus.
COD/INTA	Output	It distinguishes between instruction fetch, memory operand, reference & interrupt acknowledgement cycle.
BUSY#	Input	This signal is sent by co-processor. It is active low signal causing 80286 to wait or escape instruction.
ERROR#	Input	It is used by co-processor to cause 80286 to perform type 16 interrupt when wait or escape instruction is being executed.
BHE#	Input	It is used to indicate the transfer of data over data bus.

active low

c) Explain test register format of 80386 with suitable diagram. 4M

Ans: [2M for diag, 2M for expln.]

Test Register:- TR_6 & TR_7 → Used to test translation look aside buffer (TLB) used with paging.

TR_6 & TR_7 are used for translation look aside buffer (TLB). TLB holds page table address translation to reduce the no. of memory required for page table translation.

The test registers TR_6 & TR_7 are used to test translation look aside buffer used with paging unit.

	Control bit(11-0)											TR6
Linear address (31-12)	V	D	D#	U	U#	W	W#	0	0	0	0	C
	Control bit(11-0)											TR7
Physical address (31-12)	0	0	0	0	0	0	0	PL	REP	0	0	

V = '1' → Valid TLB entry

D = '1' → TLB entry invalid or dirty.

U = A bit for TLB.

W = Indicates that area addressed by TLB entry is writable.

C = '0' → Write or immediate look up (1) for TLB.

PL = hit of logic 1.

REP = Selects which block of TLB is written.

Hit rate : if the page is matched.

Miss rate : if the page is missed.

d) Describe any four DOS interrupt. 4M

Ans: **(1 marks each for writing entry and exit function of DOS)**

DOS interrupts are INT 21H contains 00H to 6CH. Hence student can write any four between these.
Its expected to write entry and exit

AH = 01h - READ CHARACTER FROM STANDARD INPUT, WITH ECHO

Return: AL = character read

Notes:

- ^C/^Break are checked
- ^P toggles the DOS-internal echo-to-printer flag
- ^Z is not interpreted, thus not causing an EOF if input is redirected character is echoed to standard output

AH = 02h -WRITE CHARACTER TO STANDARD OUTPUT

Entry: DL = character to write

Return: AL = last character output

Notes:

- ^C/^Break are checked
- the last character output will be the character in DL unless DL=09h on entry, in which case AL=20h as tabs are expanded to blanks
- if standard output is redirected to a file, no error checks (write- protected, full media, etc.) are performed

AH = 05h - WRITE CHARACTER TO PRINTER

Entry: DL = character to print

Notes:

- keyboard checked for ^C/^Break
- STDPRN is usually the first parallel port, but may be redirected under DOS 2+
- if the printer is busy, this function will wait

AH = 06h - DIRECT CONSOLE OUTPUT

Entry: DL = character (except FFh)

Return: AL = character output

Notes: does not check ^C/^Break

AH = 06h - DIRECT CONSOLE INPUT

Entry: AH = 06h DL = FFh

Return:

- ZF *set* if no character available and AL = 00h
- ZF *clear* if character available AL = character read

Notes:

- ^C/^Break are NOT checked
- if the returned character is 00h, the user pressed a key with an extended keycode, which will be returned by the next call of this function
- although the return of AL=00h when no characters are available is not documented, some programs rely on this behavior

AH=07h - DIRECT CHARACTER INPUT, WITHOUT ECHO

Return: AL = character read from standard input

Notes: does not check ^C/^Break

SeeAlso: AH=01h,AH=06h,AH=08h,AH=0Ah

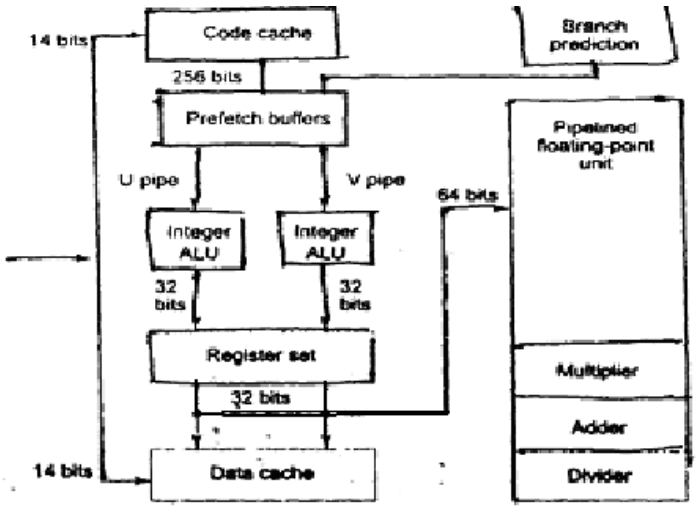
OR

DOS interrupts are INT 21H contains 00H to 6CH.
Hence student can write any four between these.
For example

1) Int 21
   Function 00H
   Terminate process: All memory belonging to the process is released
   Control is transferred to the termination handler.
   Call with: AH =00H
   Returns: Nothing
2) Int 21
   Function 01H
   Character input with echo: Inputs a character from the keyboard ,then echos it to the display. If no character is ready, waits until one is available..
   Call with: AH =01H
   Returns: AL = 8 bit input data
3) Int 21
   Function 02H
   Character Output: Outputs a character to the currently active video display
   Call with: AH =02H
   DL = 8 bit data for output
   Returns: Nothing
4) Int 21
   Function 03H
   Auxiliary input: Reads character from the first serial port
   Call with: AH =03H
   Returns: AL = 8 bit input data

e) Draw the architecture of Pentium processor.

Ans: (4 marks for any neat labeled diagram containing all blocks shown in diagram)



Q4 Attempt any four of the following. 16M

a) Explain the system segment descriptor of 80286. 4M

Ans: (Descriptor table 2 marks 2 marks for its field)

System Segment Descriptor

Intel Reserved					
P	DPL	S=0	TYPE	A	Base (23-16)
Base (15-0)					
Limit (15-0)					

1) P → Presence
'O' → Segment is not present, '1' – segment is present (mapped) in physical memory
2) DPL → Descriptor Privilege Level Priority

00	-High	highest
01	-	second
10	-	third
11	-low	lowest

3) S = 'O' → System segment descriptor or gate descriptor
4) Type S = 'O' → System segment descriptor or gate descriptor

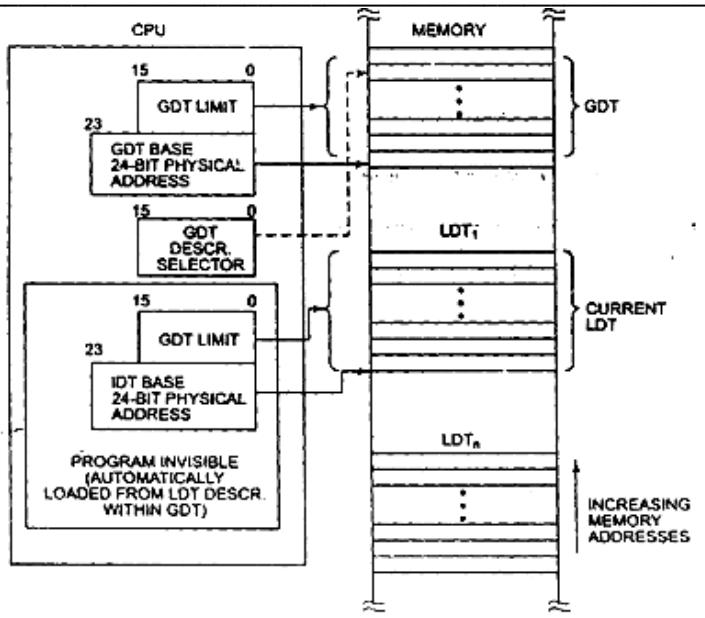
Type : 1 → Available Task State Segment (TSS) System
2 → Local Descriptor Table Descriptor
3 → Busy Task State Segment TSS

b) Explain LDT and GDT table of Intel 80286 with the diagram. 4M

Ans: (Diagram 2 marks 1 marks for each table)

Descriptor table is an array of 8k descriptor. The upper 13 bit of selector field (index) points in particular entry in a descriptor table.

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 task.
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



c) How debugging process is simplified in 80386 explain in details. 4M

Ans:

In 80386 processor on chip debugging features are available such as code execution break point, single capability provided by TF bit in flag register and code and data break point provided by debug register.
Break point instruction: INT 3 users can provide break point by this instruction
Single step Trap: TF in flag register is used for single step wise execution
Use of debug register Break points can be set using debug register.

d) Describe BIOS module and command processor. 4M

Ans: : description of BIOS module-2M; Description of Command processor-2M]

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)

Command Processor:-

Intrinsic command → internal command
Extrinsic command → external command

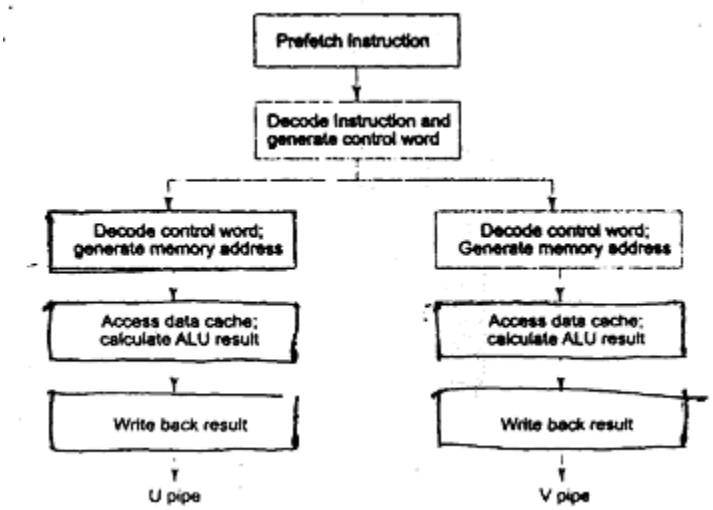
It is the user interface to the operating system. It is responsible for carrying out the commands i.e. loading or execution of program from the disk or mass storage device.

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

1) A resident portion – at low portion of memory.
2) Initialization section – at middle portion of the memory.
3) Transient Module – at the high end of the memory.

e) Explain the superscalar execution of Pentium processor. 4M

Ans: (1 marks for diagram 3 marks for stages)



OR

(Note: student may draw the pipeline instruction execution)

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.

Integer 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.

f) Explain the concept of separate code and data cache memory in 486 processor.

Ans: (2 marks for each)

(Note: In 486 only one 8K cache is available. Separate code and data cache is available in Pentium)

An instruction cache is used to provide quick access to frequently used instruction when an instruction is not in cache, it is read from external data bus & code cache is updated for it. The branch target butter & pre fetch buffer work together with the instruction cachet to fetch instructions as fast as possible.

A separate data cache stores a copy of most frequently accessed memory data Since memory access are significantly longer than processor clock cycles (longer time is required to access). The data & code cache is enabled or disabled by hardware or software.

Q5. Attempt any four of the following **16M**

a) With neat diagram explain flag register of 80286. **4M**

Ans: **[flag register format-2M; Explanation-2 M]**

D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
X	NT	IOPL		OF	D	I	T	S	Z	X	A	X	P	X	C

1) Sign flag :-

It acts according to the MSB of accumulator to indicate +ve or –ve number of a result in arithmetic or logical operation s = o, if the result is +ve, S = 1 if the result is –ve.

2) Zero flag :-

If result is O, Z = 1 otherwise Z = O. Zero flag is also affected in incremental, decremental operation of a register. If register become 0 then z = 1.

3) Auxiliary Carry flag :-

This bit is set for an overflow of a bit of accumulator as well as overflow of bit server is BCD operation.

4) Parity flag :-

P = 1 for even parity of result else zero.

5) Carry flag :-

C = 1 when addition results an overflows from higher order bit or in subtraction for a borrow otherwise C = 0.

6) Trap flag :-

If it is set, trap is executed after each instruction. It is used for step by step debugging of the program.

7) Interrupt flag :-

If this flag is set a certain type of interrupt is recognized by the CPU.

8) Direction flag :-

String instruction is used in, it decides the direction of IP jump [forward jump, backward jump]
DF = 0 when direction is forward i.e. from lower address to higher address.
DF = 1 when direction is backward i.e. from higher address to lower address.

9) Overflow flag :-

This flag is set whenever result of arithmetic or logical operation is more than destination bits. i.e. result is out of range.

10) IOPL:-

It is used in protected operation to select the privilege level of input device. **IOPL**

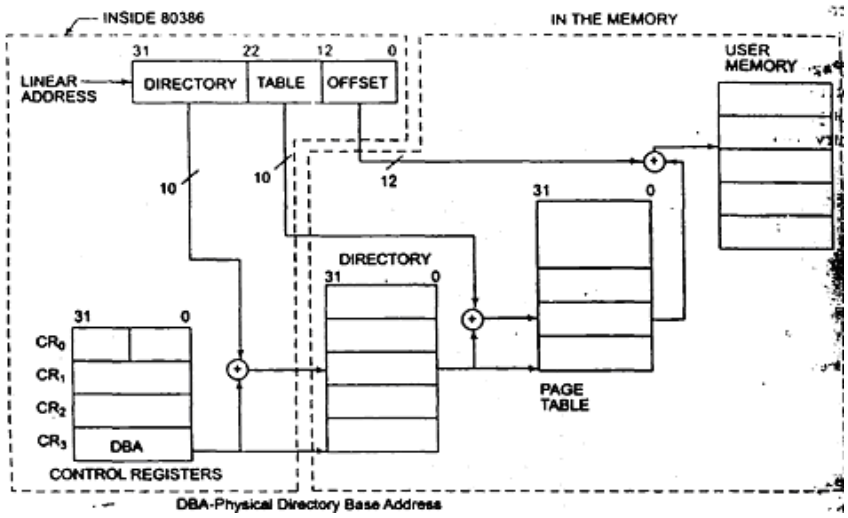
- 00 - Having highest priority
- 01 - Second highest priority
- 10 - Third highest priority
- 11 - Lowest priority

11) NT – Nested flag :-

It indicates that current task is nested within another task in protected mode operation.

b) Describe the functioning of 80386 processor in paged mode. 4M

Ans: (Any one diagram 1 marks 3 marks for description)



Paging Operation:-

Paging is a memory management technique used for virtual memory multitasking operation. The paging divides memory into fixed sizes such as 928 bytes, 256 bytes ------ 1 kb, 2 k & 4 k ---- & so on. They do not have any logical relation with a program as compared to segment.

Paging unit converts complete map of a task into pages each of size 4kb. The task is executed in terms of pages. Each task has three components :

1) Page Directory 2) Page Table 3) Page

Page Directory:-

The size of page directory is 4 kb & each directory entry has 4 bytes, therefore total 1024 entries are possible in directory. The table shows the entry format for page directory.

Page Table:-

Each page table is also of 4 kb & maximum 1024 entries are possible similar to page directory. It contains the starting address of page, the upper 20 bit $D_{31} – D_{12}$ use page frame address combined with lower 12 bits of linear address $A_{12} – A_{21}$ are used to select 1024 page table entries.

c) With labeled neat diagram explain memory organization of 80386. 4M

Ans: (1 marks for interfacing 3 marks for description)

The 80386 processor address 4GB memory with the help of BE 0 to BE3 and A2 to A31 address lines.

The A0 and A1 are enclosed with the bus enable BE signal BE 0 to BE3 .

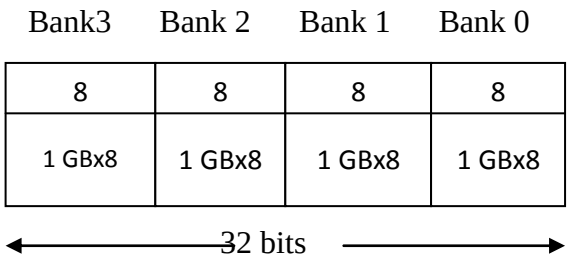
It selects the access of byte, word, double word of data. These signals are generated by A_0 & A_1 used to validate the data.

(Note: Interfacing only Data bus and address bus with demultiplexing is required no detail interface is expected)

Memory Organization:

- The memory is divided into four 8 bit wide memory banks, each containing upto 16 bytes of memory. This 32 bit wide memory organization allows bytes, words or double words of memory to be accessed directly.
- Memory location range from 00000000H to FFFFFFFFH.
- The four memory banks are accessed via bank Enable signals BE3#-BE0#.
- This arrangement allows:
  1) A byte to be accessed when one bank enable signal is activated by microprocessor.

2) A word can be accessed when two bank enable signals are activated.



d) Describe any two disk operational function of DOS in 21H. 4M

Ans: (2 marks for each)

0DH: Flush Disk Buffers ,entry AH= 0Dh : Erases all files names stored in disk buffers.
0EH: Select default disk drive, entry AH= 0E,DL=desired default disk drive number
Exit AL= the total number of drive present in the system.

e) Explain the structure of MS DOS with respective its layer. 4M

Ans: (1 mark for writing layers 1 marks for description of each layer)

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 :

6) Console display & keyboard (CON)
7) Line printer (PRN)
8) Auxiliary Device (AUX)
9) Date & Time (Clock)
10) 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

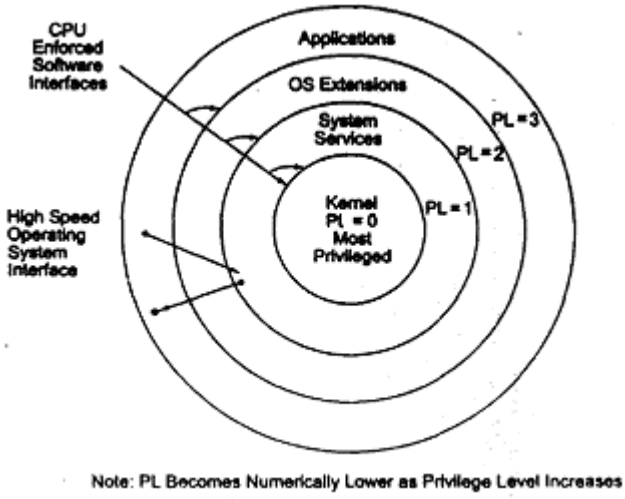
3) Command Processor :-

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.

f) Explain the privilege protection mechanism of 80286. 4M

Ans: (1mark Diagram 1 marks for each privilege level)



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.

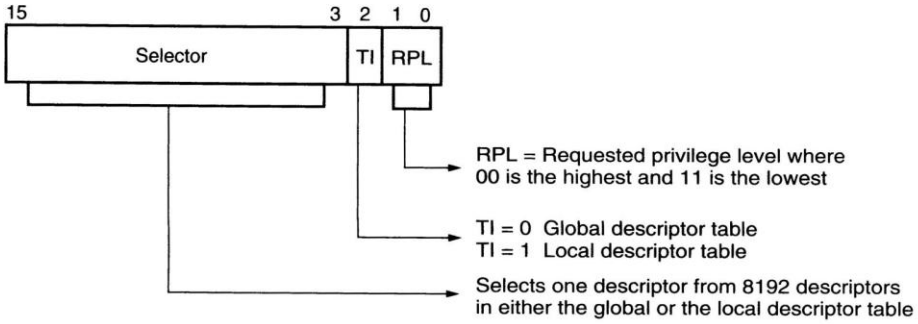
3) Selector Privilege :-

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.

Q6 Attempt any four **16M**

a) Explain the function of selector of 80286 **4M**

Ans: **(2 Marks for diagram 2 marks for description)**



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 enabling and disabling of paging in 80386. **4M**

Ans:

- The control register CR_0 to CR_3 controls the paging operation of 80386.
- CR_0 holds the MSW containing PG bit i.e. 31th bit. If PG = 1 page translation for linear address into physical address i.e. paging operation is enabled.
- Similarly if PG = 0 i.e. reset provides disabling or disable paging operation.
- After the PG bit is set, page translation takes effect on the text instruction and if the linear addresses from which you are executing are not at the same physical address they were before page translation took effect, you are going to wind up somewhere unexpected. For this reason take following steps:
  - Disable interrupt, including NMI.
  - Enable paging only from a page that is identified mapped
  - Flush the instruction prefetch queue immediate after the MOV instruction that enables paging.
- Disabling paging involves the same hazards that enabling it does.
- Be sure that your code will be mapped t the same physical space after the page translation is turned off and flush the prefetch queue afterward.

c) Which function is used to "delete file", Explain in detail with the example. 4M

Ans: (function 1M, explanation 3M)

Function 41H is used to delete the file.
Entry AH=41H, DS:DX= address of ASCII-Z string file name.
Example
Data segment
File name db 0Dh,0Ah, "C:\AMP\Prasad.txt",0Ah,0Dh,'$' (Provide the path of file)
Data ends
Code Segment

MOV Ah,41H
LEA dx, file name
INT 21h

---

d) Enlist file processing functions (any eight) 4M

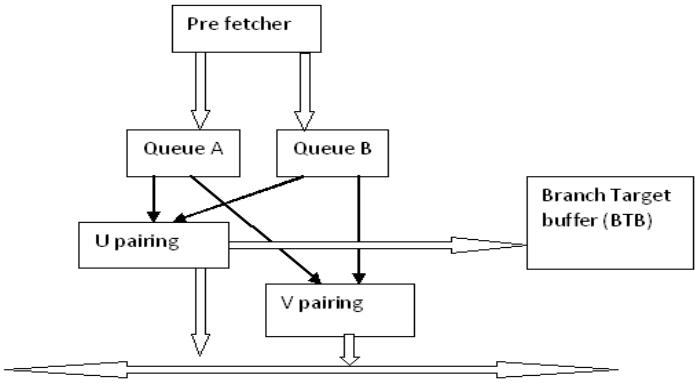
Ans: (each function ½ M)

39H: Create directory
3AH:Delete directory
3BH:Set Current Directory
3CH:Create file
3DH:open file
3EH:Close file
3FH:Read file
40H: Write file
41H: Delete file
43H:Get/Set file attribute
47H:Get Current directory
56H:Rename file
57H:Get/Set file date and time

e) Explain the branch prediction features of 80486. 4M

Ans:

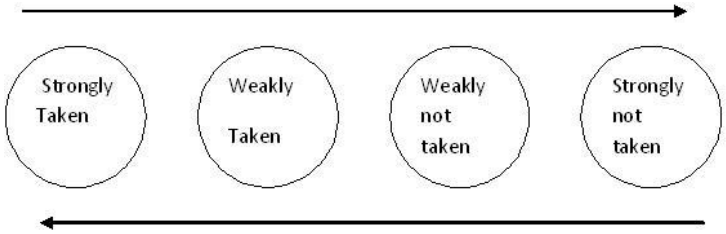
***(Note:- The branch prediction logic is not available in 80486. It is available in Pentium processor. Marks to be given for branch prediction logic*.)**



The processor includes branch prediction logic to avoid pipeline stalls, 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.

The prediction mechanism is implemented using 4 way set associative cache with 256 entries referred as branch target buffer. Whenever branch is taken CPU enters the branch instruction address & the destination address in BTB. When an instruction is decoded CPU searches the BTB to determine presence of entry. If its present CPU uses precious history to decide to take the branch. The history bits can indicate one of the four possible stages & updated as follows.

Movement when not taken



Movement when branch is taken.

f) State any four differences between .COM and .EXE. 4M

Ans:

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