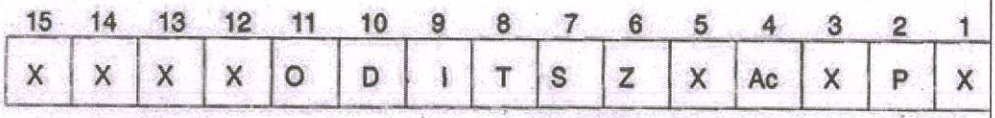
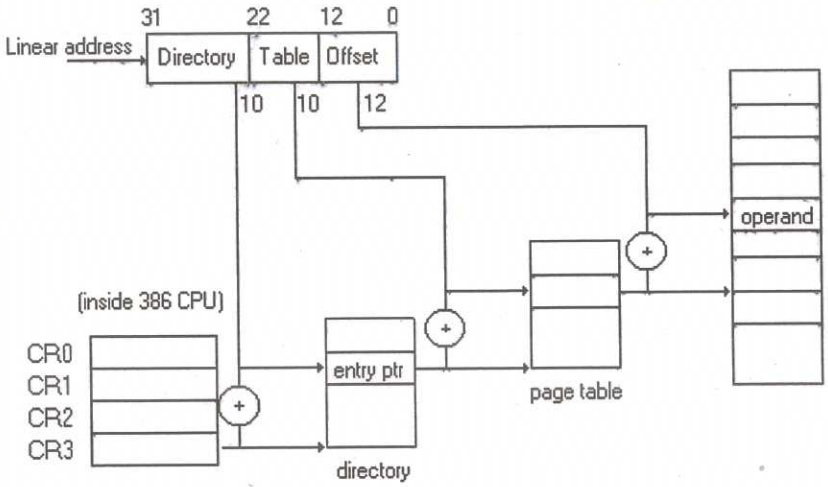
Scoring Indicators

Code :6041 Version:A

Qn. No.	Scoring Indicators	Split score	Total score
	PART-A		
I 1	There are four segment register in 8086 • Code segment register (CS) • Data segment register (DS) • Extra segment register (ES) • Stack segment register (SS)	2	
2	The way in which a source operand is denoted in an instruction is known as **addressing mode.**	2	5x2=10
3	Assembler directive is a message to the assembler that tells it how to translate instructions into machine code.	2	
4	Real address mode ,protected virtual address mode and virtual 8086 mode.	2	
5	Core is the processor within the CPU.	2	
	PART-B		
II 1	1. Single +5V power supply 2. Clock speed range of 5-10MHz 3. capable of executing about 0.33 MIPS (Millions instructions per second) 4. It is 16-bit processor having 16-bit ALU, 16-bit registers, internal data bus, and 16-bit external data bus resulting in faster processing. 5. It uses two stages of pipelining, i.e. Fetch Stage and Execute Stage, which improves performance. 6. Fetch stage can prefetch up to 6 bytes of instructions and stores them in the queue. 7. It has 256 interrupts. (any five)	6	6
2	 O – Overflow flag D – Direction flag I – Interrupt flag T – Trap flag S – Sign flag Z – Zero flag Ac – Auxiliary carry flag P – Parity flag Cy – Carry flag X – Not used	6	6

3	The INTEL predefined interrupts for 8086 are; 1. Division by zero(Type-0) interrupt. 2. Single step(Type-1) interrupt. 3. Nonmaskable interrupt, NMI(Type-2) interrupt. 4. Breakpoint(Type-3) interrupt. 5. Interrupt on overflow(Type-4) interrupt. The 8086 will automatically do a type-0 interrupt if the result of a division operation is too large to fit in the destination register and this interrupt is nonmaskable. The user can write an ISS for type-1 interrupt to halt the processor temporarily and return the control to the user so that after execution of each instruction, the processor status (content of register/memory) can be verified. If they are correct then we can proceed to execute the next instruction. Execution of one instruction by one instruction is known as single step and this feature will be useful to debug a program. The 8086 processor will automatically generate a type-2 interrupt when it receives a low-to-high transition on its NMI input pin. This interrupt cannot be disabled or masked. Usually, the type-2 interrupt is used to save program data or processor status in case of system ac power failure. Type-3 interrupt is used to implement a breakpoint function, which executes a program partly or up to the desired point and then return the control to the user. In the 8086 processor, the Overflow Flag (OF) will be set if the signed arithmetic operation generates such a result whose size is larger than the size of the destination register/memory. During conditions, the type-4 interrupt can be used to indicate an error condition. The type-4 interrupt is initiated by "INTO" instruction.	6	6
4	1. 64 bit data bus 2. 8 bytes of data information can be transferred to and from memory in a single bus cycle 3. Supports burst read and burst write back cycles 4. Supports pipelining 5. 8 KB of dedicated instruction cache 6. Two Integer execution units, one Floating point execution unit 7. Dual instruction pipeline 8. 8 KB dedicate data cache (Any five)	6	6
5	The **paging** unit of **80386** uses a two level table **mechanism** to convert a linear address provided by segmentation unit into physical addresses. ... **Paging** Descriptor Base Register: The control register CR2 is used to store the 32-bit linear address at which the previous page fault was detected.	Fig-4 Expl-2	6



6.

1. More processing power
2. Faster
3. Smaller energy consumption
4. Less heat production

4x1.5=6 | 6

7.

1. Data Transfer instruction

All the instructions which perform data movement come under this category. The source data may be a register, memory location, port etc. the destination may be a register. memory location or port

2: Arithmetic Instructions

Instructions of this group perform addition, subtraction, multiplication, division, increment, decrement, comparison, ASCII and decimal adjustment etc.

3. Logical Instructions

Instruction of this group perform logical AND, OR, XOR, NOT and TEST operations.

4. Rotate Instructions
5. Shift Instructions
6. Branch Instructions

It is also called program execution transfer instruction. Instructions of this group transfer program execution from the normal sequence of instructions to the specified destination or target.

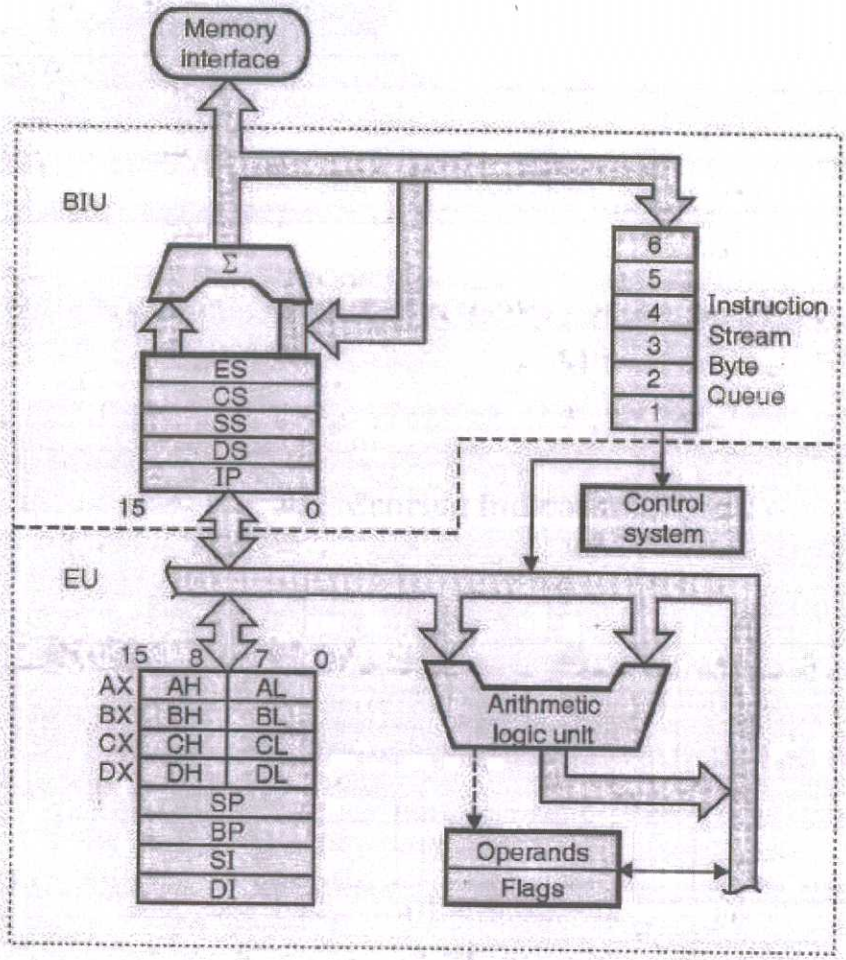
7. Flag Manipulation and Processor Control Instructions

Instructions of this instruction set are related to flag manipulation and machine control.

8. String Instructions

String is series of bytes or series of words stored in sequential memory locations.

6 | 6

Qn. No.	Scoring Indicators	Split score	Total score
III a)	 8086 internal architecture	Fig-6 Expl-4	10
b)	In a **minimum mode 8086** system, the microprocessor **8086** is operated in **minimum mode** by strapping its MN/MX pin to logic 1. In this **mode**, all the control signals are given out by the microprocessor chip itself..	3+2	5

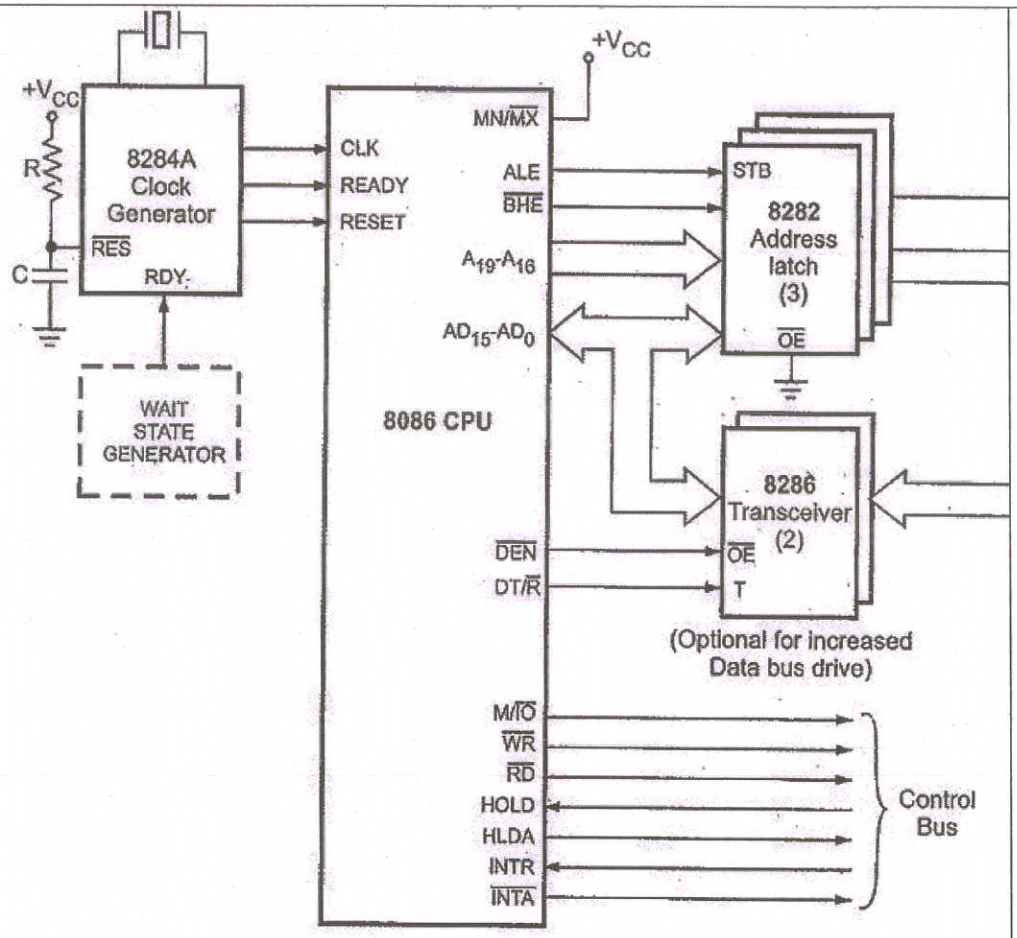


Fig. 10.2 Typical minimum mode configuration

IV a)

a) General registers
b) Index registers
c) Segment registers
d) Pointer registers
e) Status Register

General Registers

All general registers of the 8086 microprocessor can be used for arithmetic and logic operations. These all general registers can be used as either 8-bit or 16-bit registers. The general registers are:

i. **AX (Accumulator):**
AX is used as 16-bit accumulator. The lower 8-bits of AX are designated to use as AL and higher 8-bits as AH. AL can be used as an 8-bit accumulator for 8-bit operation.

This Accumulator used in arithmetic, logic and data transfer operations. For manipulation and division operations, one of the numbers must be placed in AX or AL.

Fig.6+4

10

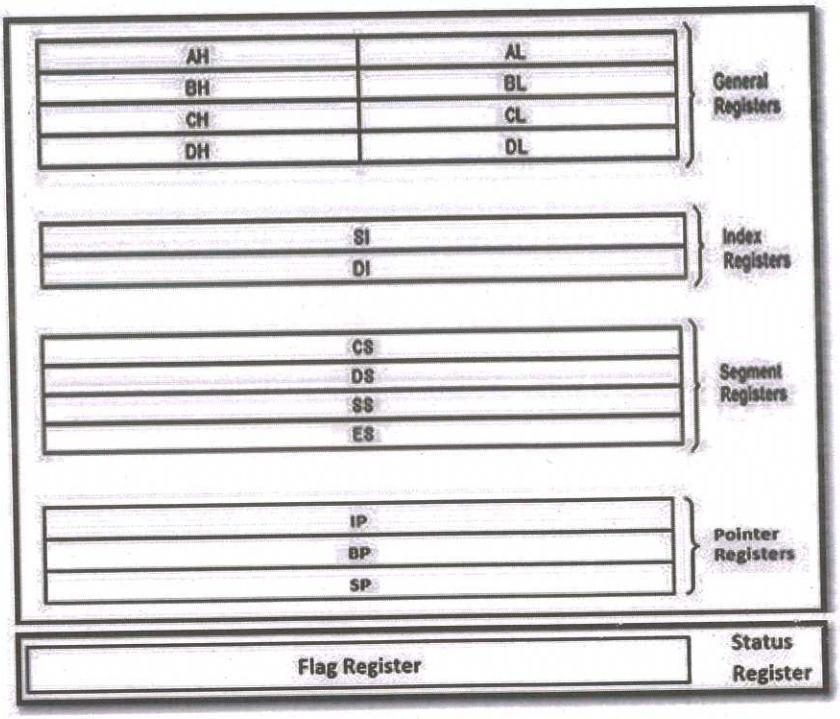


Figure: Register groups of 8086 micro-processor

BX (Base Register):

BX is a 16 bit register, but BL indicates the lower 8-bits of BX and BH indicates the higher 8-bits of BX. The register BX is used as address register to form physical address in case of certain addressing modes

ii. **CX (Count Register):**
The register CX is used default counter in case of string and loop instructions. Count register can also be used as a counter in string manipulation and shift/rotate instruction.

iii. **DX (Data Register):**
DX register is a general purpose register which may be used as an implicit operand or destination in case of a few instructions. Data register can also be used as a port number in I/O operations.

e) Segment Register: 8086 can able to access a memory capacity of up to 1 megabyte. This 1 megabyte of memory is divided into 16 logical segments. Each segment contains 64 Kbytes of memory.
There are four segment registers to access this 1 megabyte of memory. The segment registers of 8086 are:

i. **CS (Code Segment):**
Code segment (CS) is a 16-bit register that is used for addressing memory

location in the code segment of the memory (64Kb

ii. **Stack segment (SS)**
Stack Segment (SS) is a 16-bit register that used for addressing stack segment of the memory (64kb) where stack data is stored..

iii. **Data segment (DS)**
Data Segment (DS) is a 16-bit register that points the data segment of the memory (64kb) where the program data is stored.

iv. **Extra segment (ES):**
Extra Segment (ES) is a 16-bit register that also points the data segment of the memory (64kb) where the program data is stored.

f) Index Registers

The index registers are particularly useful for string manipulation.

i. **SI (Source Index):**
SI is a 16-bit register. This register is used to store the offset of source data in data segment**DI (Destination Index):**
DI is a 16-bit register. This is destination index register performs the same function as SI.

g) Pointer Registers:

Pointer Registers contains the offset of data(variables, labels) and instructions from their base segments (

i. **SP (Stack Pointer):**
Stack Pointer register points the program stack

ii. **BP (Base Pointer):**
Base Pointer register also points the same stack segment. Unlike SP, we can use BP to access data in the other segments also.

iii. **IP (Instruction Pointer):**
The Instruction Pointer is a register that holds the address of the next instruction to be fetched from memory.It contains the offset of the next word of instruction code instead of its actual address

h) Status Register:

The status register also called as flag register. The 8086 flag register contents indicate the results of computation in the ALU. It also contains some flag bits to control the CPU operations.

IV b)

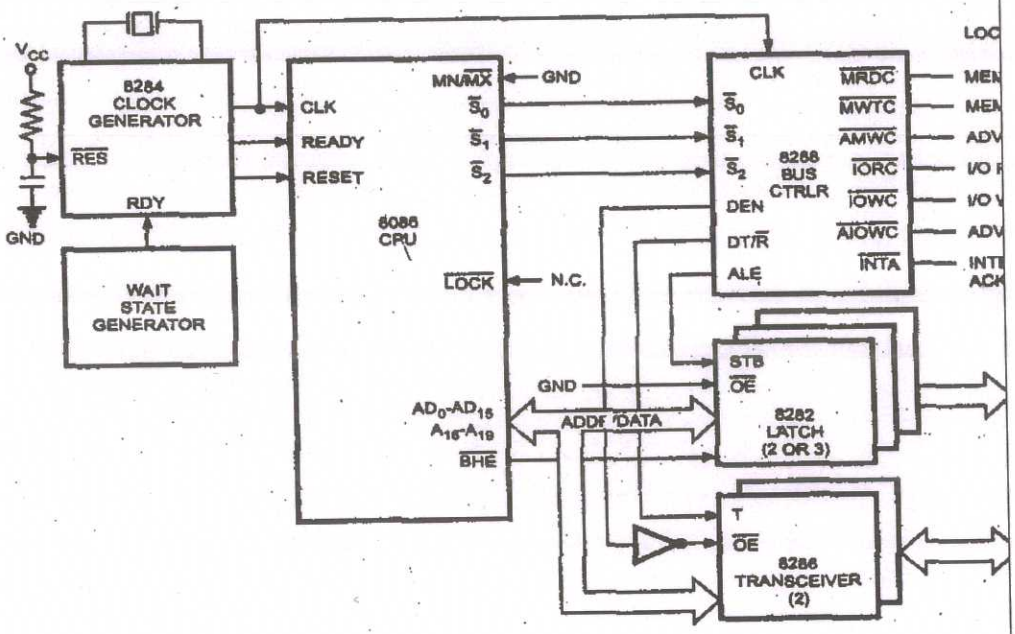


Fig3+2

5

A **processor** is in the Maximum Mode Configuration of 8086 when its MN/MX pin is grounded. In this mode, the Bus controller (8288) chip used to generate control signals I/O W, I/O R, RD., WR (Active low), etc., by receiving the active low status signals (S2, S1 & S0) from the microprocessor. Types of addressing modes:

V

Register mode – In this type of addressing mode both the operands are registers.

Example:

MOV AX, BX

Immediate mode – In this type of addressing mode the source operand is a 8 bit or 16 bit data. Destination operand can never be immediate data.

Example:

MOV AX, 2000

Displacement or direct mode – In this type of addressing mode the effective address is directly given in the instruction as displacement.

Example:

MOV AX, [0500]

Register indirect mode – In this addressing mode the effective address is in SI, DI or BX.

15

15

Example:

MOV AX, [DI]

Based indexed mode – In this the effective address is sum of base register and index register.

The physical memory address is calculated according to the base register.

Example:

MOV AL, [BP+SI]

Indexed mode – In this type of addressing mode the effective address is sum of index register and displacement.

Example:

MOV AX, [SI+2000]

Based mode – In this the effective address is the sum of base register and displacement.

Example:

MOV AL, [BP+ 0100]

Based indexed displacement mode – In this type of addressing mode the effective address is the sum of index register, base register and displacement.

Example:

MOV AL, [SI+BP+2000]

Input/Output mode – This addressing mode is related with input output operations.

Example:

IN A, 45

OUT A, 50

VI a) In an 8086 Interrupt system the first 1 Kbyte of memory from 00000H to 003FFH is reserved for storing the starting addresses of interrupt service routines. This block of memory is often called the interrupt vector table or the interrupt pointer table. Since 4 bytes are required to store the CS and IP values for each interrupt service procedure, the table can hold the starting addresses for 256 interrupt service routines.

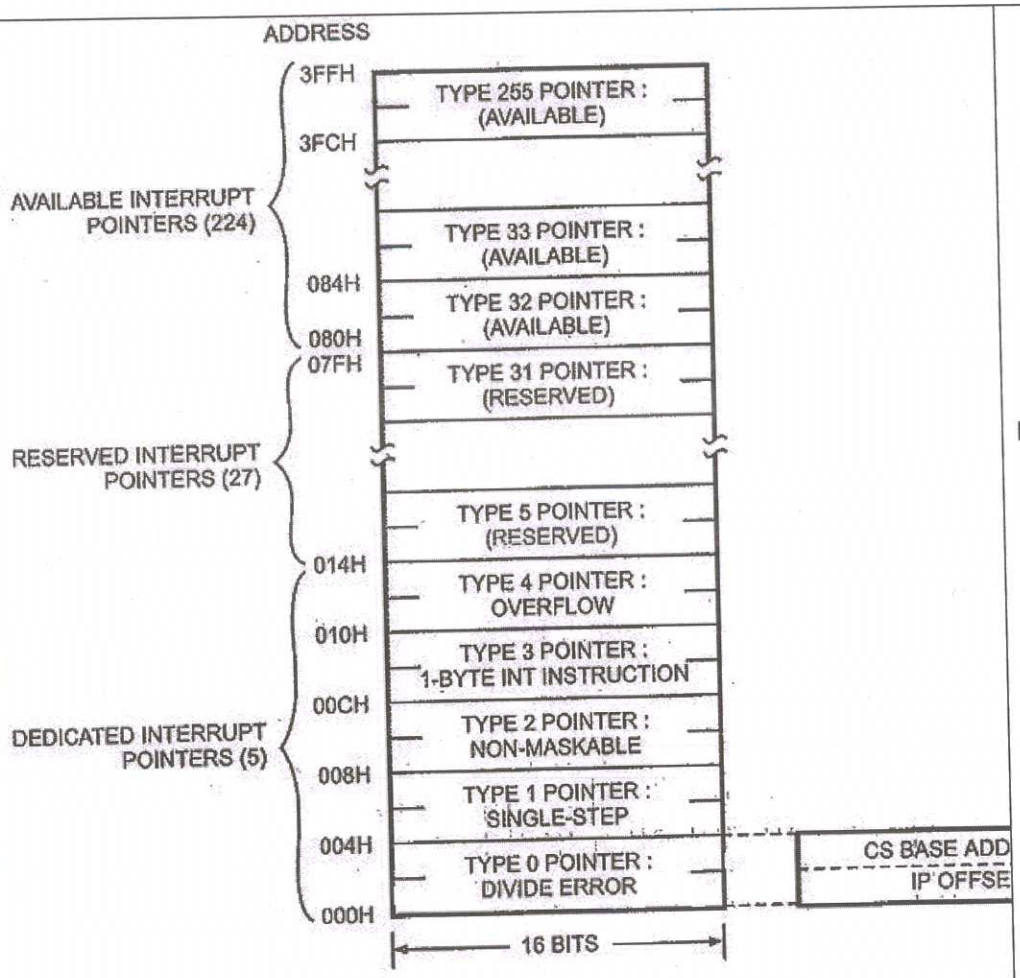


Fig. 9.2 8086 interrupt vector table

VI b)

1.Hardware Interrupts (External Interrupts)

ex: NMI, INTR

2.Software Interrupts (Internal Interrupts and Instructions)

ex: INT n (Software Instructions)

3.Some conditions produced by the execution of an instruction.

VII

Pentium processor uses Superscalar **architecture** and hence can issue multiple instructions per cycle. **Pentium** processor executes instructions in five stages. This staging, or pipelining, allows the processor to overlap multiple instructions so that it takes less time to execute two instructions in a row.

It has data bus of 64 bit and address bus of 32- bit • There are two separate 8kB caches – one for code and one for data.

Fig 7+3

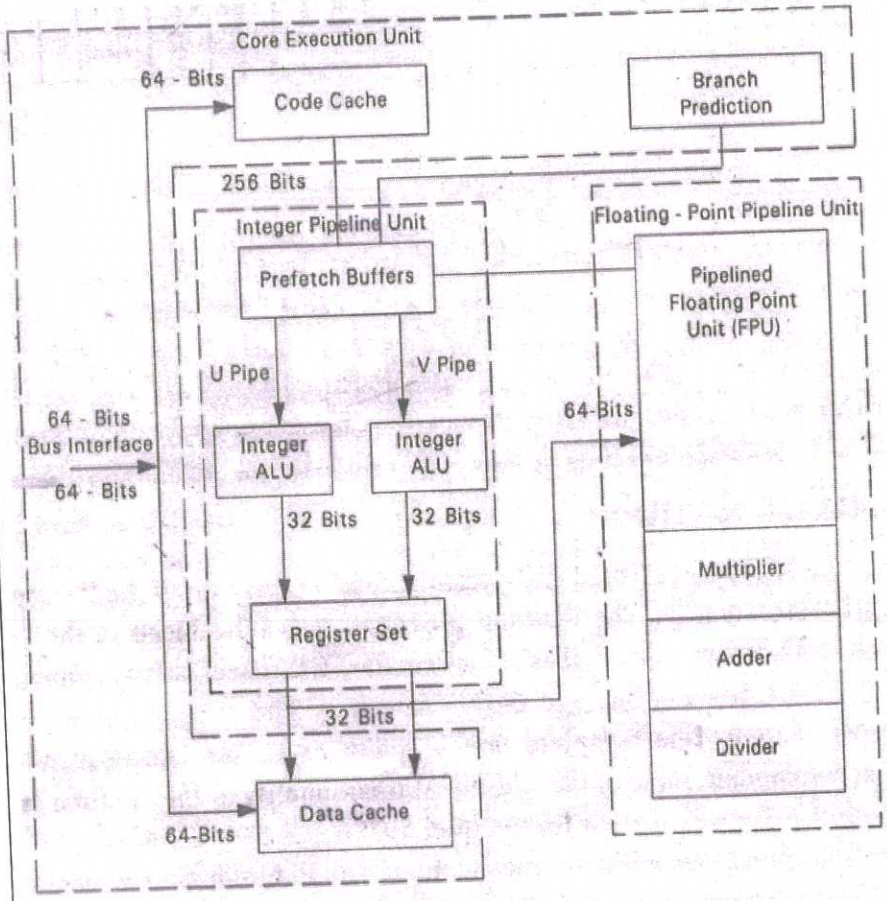
10

5

5

Fig 9+6

15



Pentium Architecture

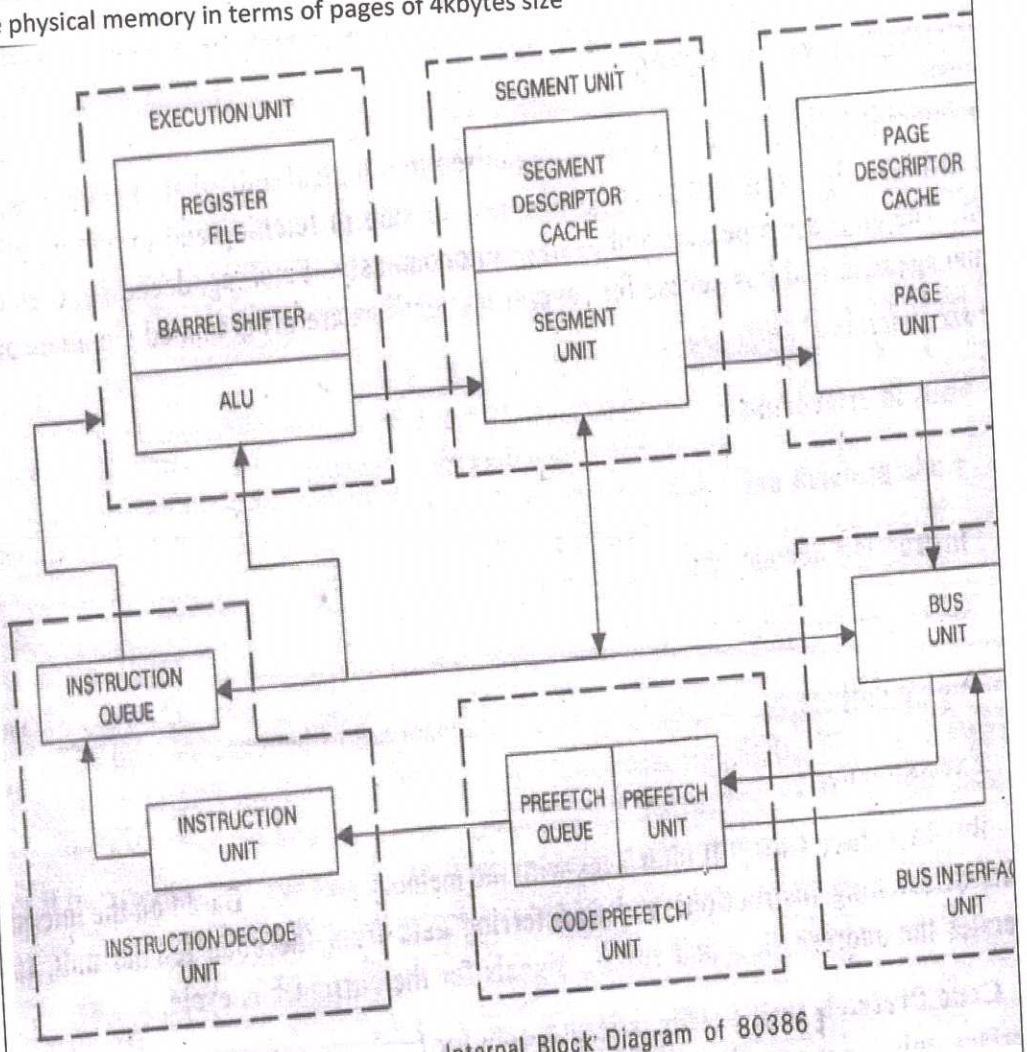
VIII a) The Internal Architecture of 80386 is divided into 3 sections. • Central processing unit • Memory management unit • Bus interface unit • Central processing unit is further divided into Execution unit and Instruction unit • Execution unit has 8 General purpose and 8 Special purpose registers which are either used for handling data or calculating offset addresses.

The Instruction unit decodes the opcode bytes received from the 16-byte instruction code queue and arranges them in a 3- instruction decoded instruction queue. • After decoding them pass it to the control section for deriving the necessary control signals. The barrel shifter increases the speed of all shift and rotate operations. • The multiply / divide logic implements the bit-shift-rotate algorithms to complete the

Fig 6+4

10

operations in minimum time. • The Memory management unit consists of a Segmentation unit and a Paging unit. • Segmentation unit allows the use of two address components, viz. segment and offset for relocability and sharing of code and data. • Segmentation unit allows segments of size 4Gbytes at max. • The Paging unit organizes the physical memory in terms of pages of 4kbytes size



Internal Block Diagram of 80386

VIII b) The **Intel** Pentium Pro processor implements**dynamic** execution using an out-of-order, speculative execution engine, with register renaming of integer, floating point and flags variables, multiprocessing bus support, and carefully controlled **memory** access reordering. 5

IX a) The Intel Core 2 Duo (also known as Core2 Duo) processor is a 64 bit dual core processor. This means two processor cores work inside a Core 2 Duo in parallel. The Core 2 Duo, which was introduced on July 27 2006, is the direct successor of the Core Duo. Each core is based on the **Pentium M** micro architecture. Fig 7+3 5 10

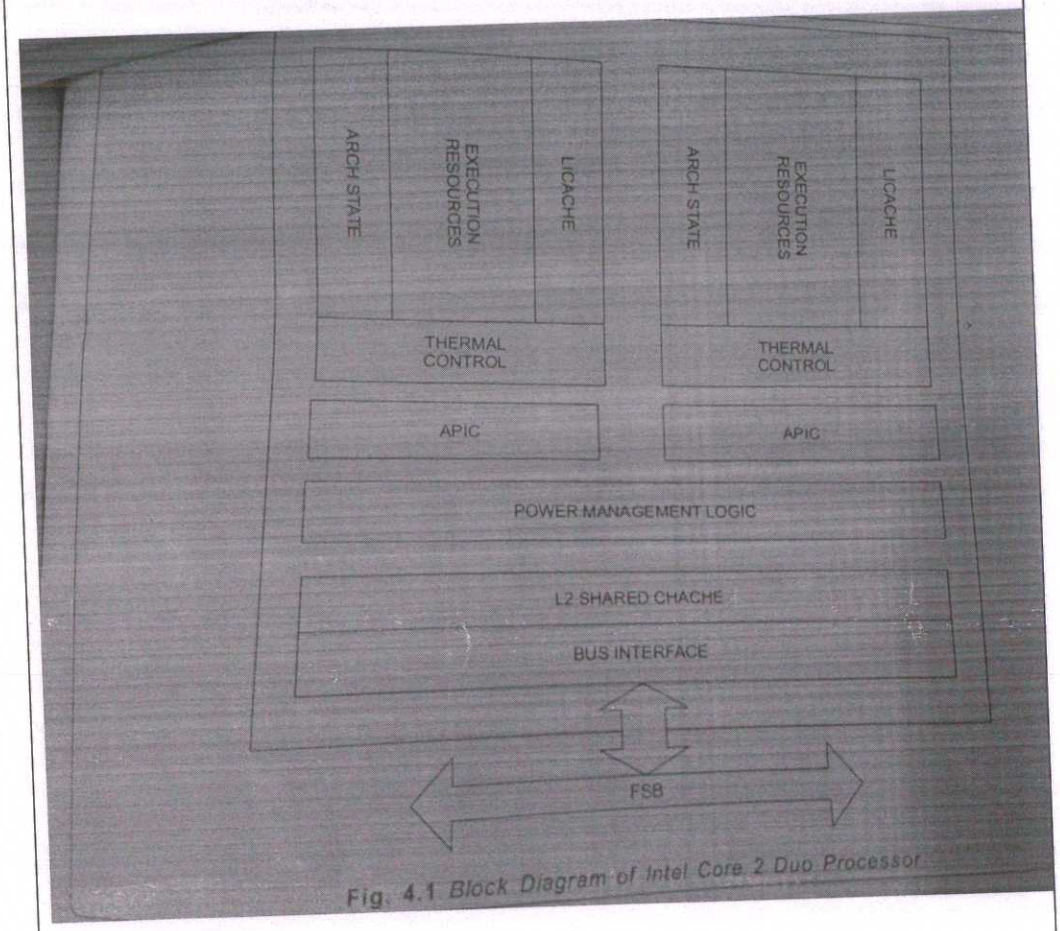


Fig. 4.1 *Block Diagram of Intel Core 2 Duo Processor*

IX b)

- IA-64 is a 64-bit processor architecture developed at Intel that is based on Explicitly Parallel Instruction Computing (EPIC).
- Sophisticated Branch Architecture.
- Loop Control Hardware.
- Control & Data Speculation.
- Cache Control.
- Powerful Integer Architecture.
- Advanced Floating Point Architecture.
- Multimedia Support (MMX™**Technology**) Memory. (Any five)

5

5

X a)

4.10 COMPARISON OF CORE I3, I5 AND I7 PROCESSORS :

Model	Core i3	Core i5	Core i7
Number of cores	2	4	4
Clock Speed Range (Several Models)	3.4GHz – 4.2GHz	2.4GHz – 3.8GHz	2.9GHz – 4.2GHz
Cache Memory	3 – 4MB	4 – 6MB	8MB
Hyper-threading	Yes	No	Yes
Turbo boost	No	Yes	Yes
K model	No	Yes	Yes

10 | 10

X b) Intel Hyper-Threading Technology uses processor resources more efficiently, enabling multiple threads to run on each core. As a performance feature, it also increases processor throughput, improving overall performance on threaded software.

5 | 5

By combining Intel processors and chipsets with an operating system and BIOS supporting Intel HT Technology, we can:

- Run demanding applications simultaneously while maintaining system responsiveness
- Keep systems protected, efficient, and manageable while minimizing impact on productivity
- Provide headroom for future business growth and new solution capabilities