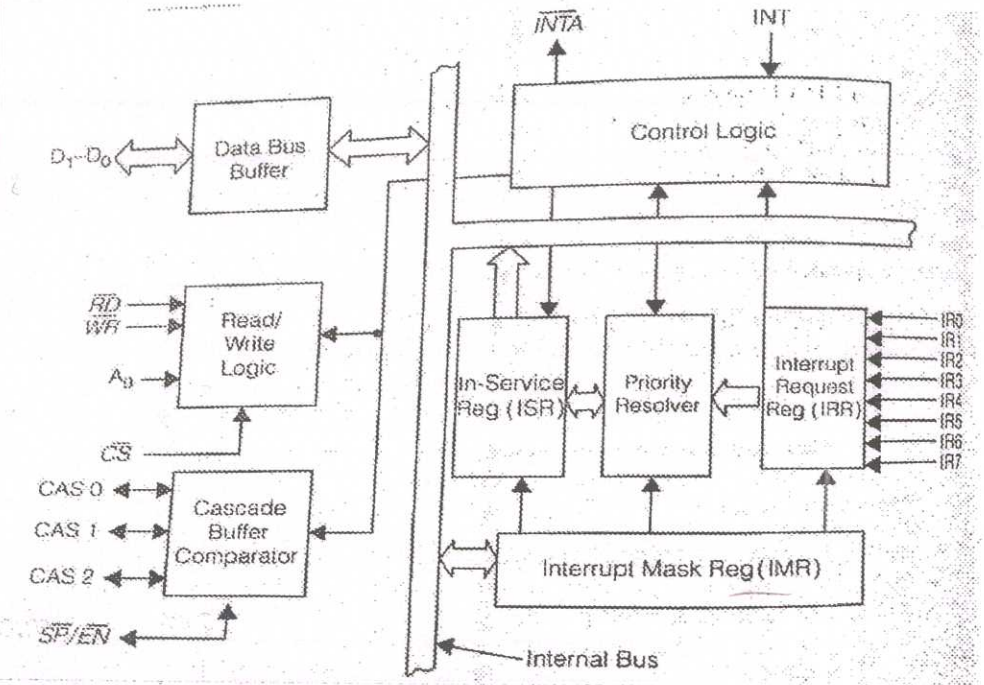
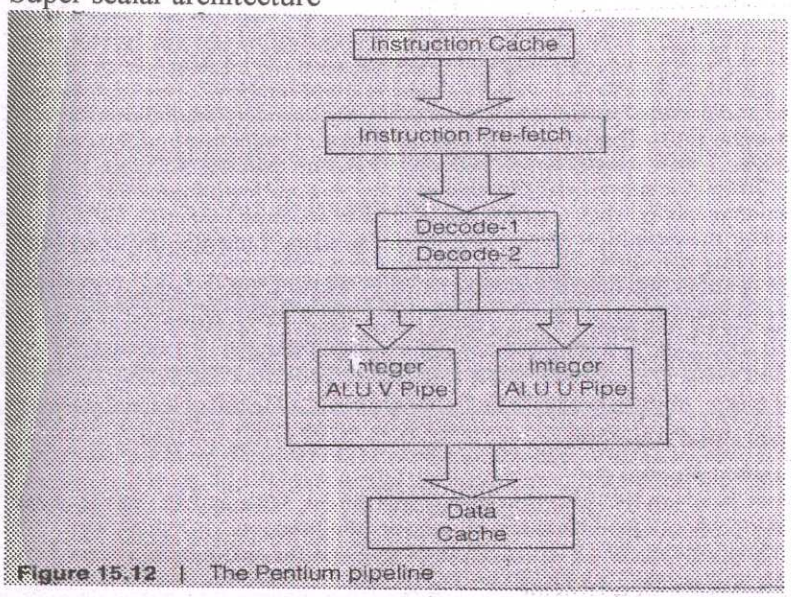
Scoring Indicators

Code: 5131

Qn: No		Split score	Total score
I 1	Data bus- 16 bits. Address bus- 20bits	1+1	2
I 2	Opcode - MOV operand - AL & 02h	1+1	2
I 3	JC,JNC,JZ,JNZ,JE,JA,JB,JO,JP etc	1+1	2
I 4	Keyboard inteface, display interface	1X2	2
I 5	Simultaneous execution of multiple threads using multiple cores.	1X2	2
II 1	Conditional flags CF- carry flag (CF=1 if carry generates) PF- Parity flag (to make result odd/even parity) AF- auxiliary carry flag (AF=1 if carry generates from the lower nibble, 4th bit position) ZF-Zero flag (ZF=1 if result is zero) SF- sign flag (SF=1 if result is negative) OF- overflow flag (OF=1 if result doesn't fit into the register)	6x1	6
II 2	addressing modes 1. Register →operands in register 2. Immediate →source constant data 3. Direct → source / destination memory address 4. Register indirect → address of data in register 5. Register relative →operand address = address in register + displacement 6. Based indexd → operand address= [BX]+[SI / DI] 7. Relative based index → operand address= [BX]+[SI / DI] + displacement	Any 4 4x 1.5	6
II 3	MOV , LEA, XCHG, PUSH, POP, LDS/LES, LAHF, SAHF, PUSHF, POPF XLAT	Any 4 4X 1.5	6
II 4	INT 0 – DIVISION BY ZERO INT 1- SINGLE STEPPING INT 2 – NMI INT 3 - BREAKEPOINT INT 4- OVERFLOW INTERRUPT	6	6

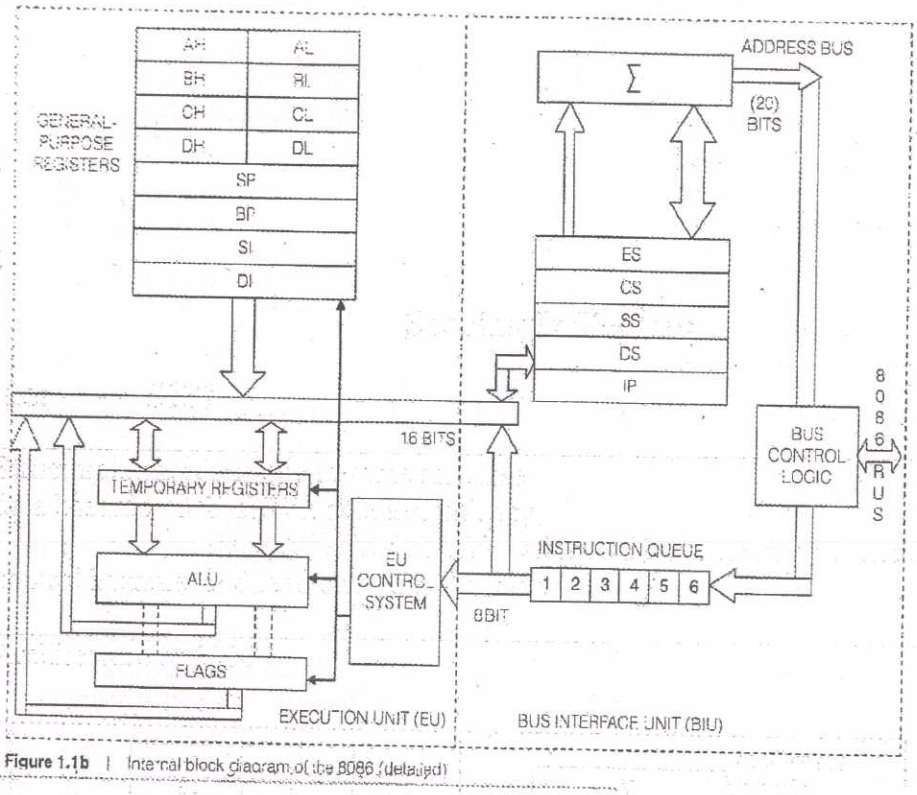
Scoring Indicators

Code: 5131

II 5	8259  re 10.23 | Internal block diagram of a programmable interrupt controller (PIC)	6	6
II 6	Super scalar architecture  Figure 15.12 | The Pentium pipeline Explanation • Two instructions can be simultaneously fetched, decoded and then executed using two pipelines U & V (Superscalar of degree 2). • The result is updated in the data cache. • Instructions are pre-fetched from instruction Cache. • At a time two instructions can be decoded using two decoder units . • Based on the operation code, instructions can be executed using the two integer units (U and V pipeline). • Data is received from the data cache.	3+3	6

Scoring Indicators

Code: 5131

II 7	Structural hazards→ due to same resourse data hazards → based on data dependency, Read After Write, Write after Read and Write after write control hazards → due to branch instructions	2 X 3	6
III a	Architecture of 8086  Figure 1.1b | Internal block diagram of the 8086 (detailed) Block diagram Explanation Registers(general purpose, segment , index , pointer etc) Adder Instruction queue Flags ALU EU Control unit Bus control logic	8 +7	15
IVa	Segment register → starting address of segment (16 bits) IP → offset (16 bits) 20bit physical address = left shift segment register 4 bits +offset Eg:	4+4	15

Scoring Indicators

Code: 5131

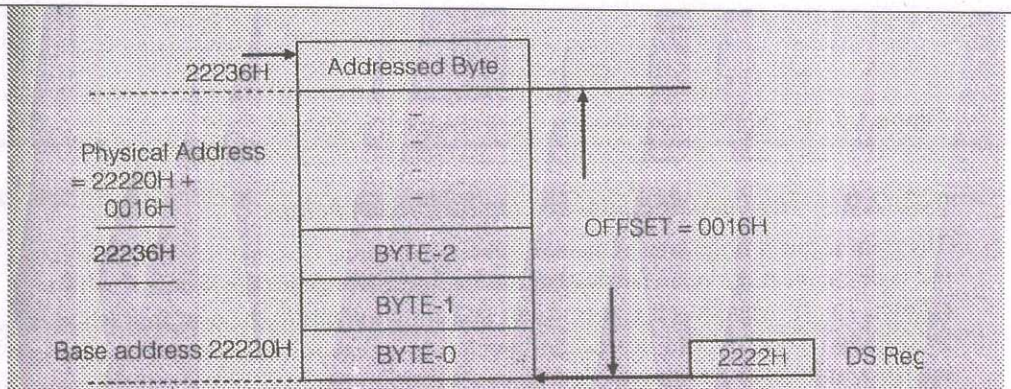
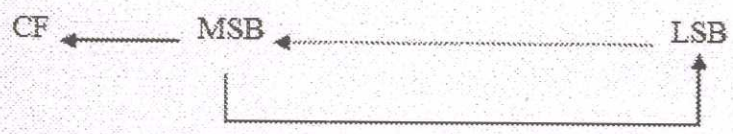
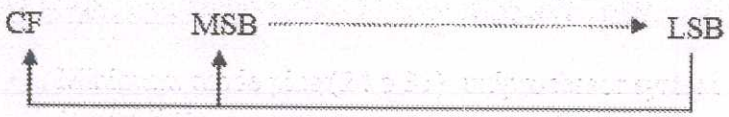
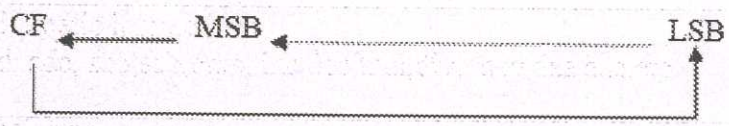
<table>
<tr><td></td><td> 

Figure 1.4 | Calculation of a physical address from the logical address for a data segment</td><td></td><td></td></tr>
<tr><td>IV b</td><td>Minimum mode pins (24 o 31)- uniprocessor system
MN/MX =1
Pins -INTA , ALE, DEN, DT/R, M/IO, WR, HLDA, HOLD</td><td>7</td><td></td></tr>
<tr><td>V a</td><td>Arithmetic instructions
add, adc, sub,sbb,cmp, inc,dec,mul,div, neg,daa,aaa,etc</td><td>2 x 4 =8</td><td></td></tr>
<tr><td>V b</td><td>Shift- SHL, SHR, SAR, SAL
Rotate- ROL, ROR, RCL, RCR
ROL
 

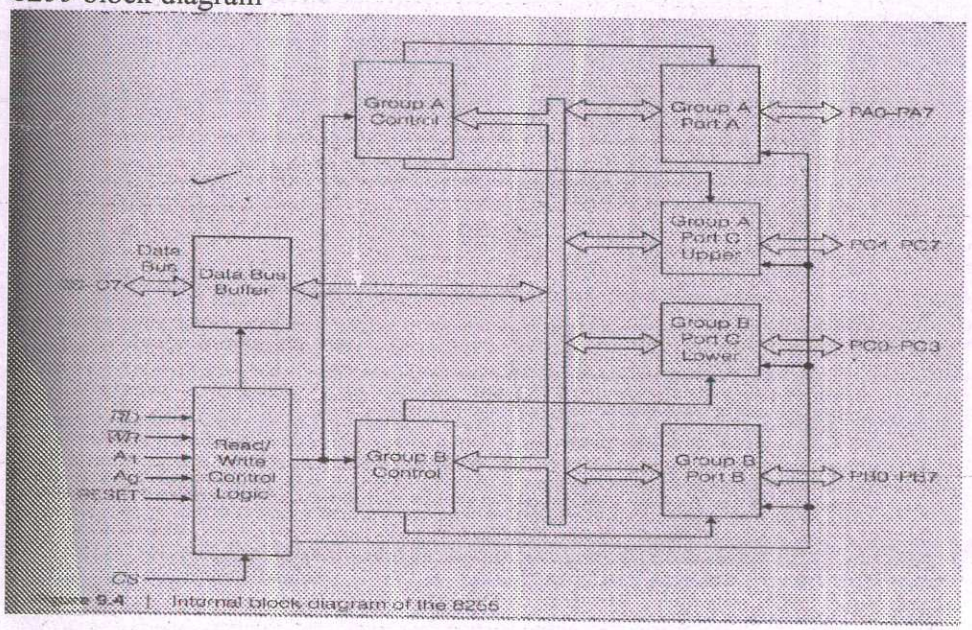
ROR
 

RCL
 

RCR
 
</td><td>List (1)
+
1.5 x 4 = 6</td><td>15</td></tr>
</table>

Scoring Indicators

Code: 5131

VI a	String instructions MOVSB/MOVSW → mov from SI TO DI STOSB/STOSW →store content in al/ax to di LODSB/LODSW → load al/ah with value in si SCASB/SCAW → scan the string to search the content in al/ax is present or not CMPSB/CMPSW → compare content in si and di REP/REPZ → repeat string instruction	List (2) + Any 3X2= 6 Total =8	15
VI b	procedure definition: syntax→ procedurename proc near/far (near/far optional) statement1 --- ret procedurename endp Procedure Invoke/call: syntax→ call procedurename	5+2	
VII a	Hardware interrupts INTR , NMI	3+3	
	Software interrupts Specified using instructions Int 21h	2 Total =8	15
VII b	Interrupt execution steps Explanation While calling INT → push flag , clear IF, clear TF , push CS, push IP Then executes interrupt On executing IRET→ POP IP, POP CS, POP flag, Then continue main routine.	5+2	
VIII a	8255 block diagram 	5+3	15

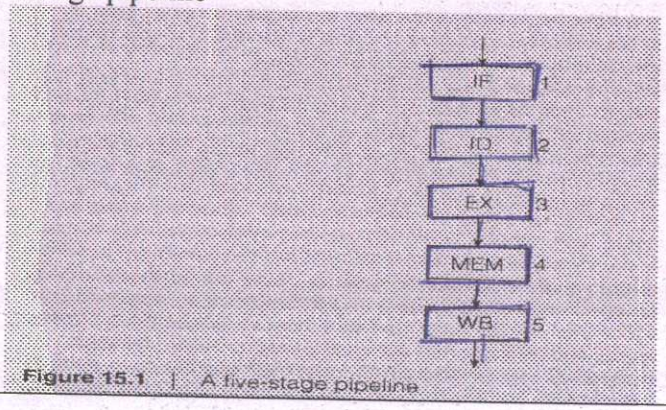
Scoring Indicators

Code: 5131

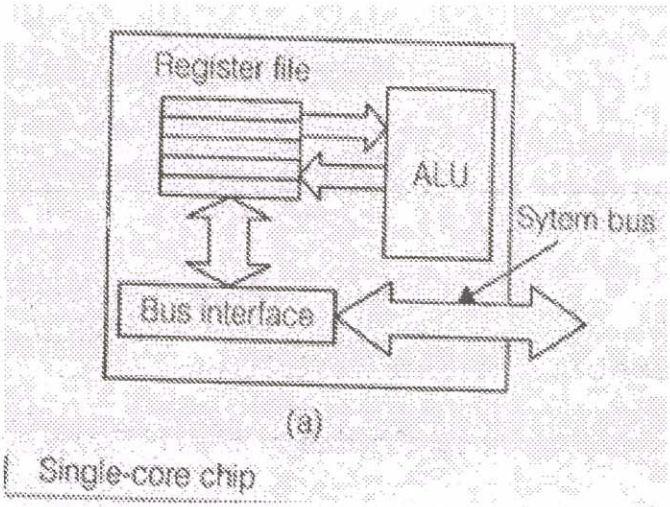
	8255 has ➢ Two 8bit PORTS - PORT A & PORT B (PB 0 TO PB 7) ➢ Two 4bit PORTS ➢ PORT C UPPER , PORT C LOWER ➢ Group A control - To control port A and upper of Port C ➢ Group B control - To control port B & lower of port C) ➢ Data buffer (bidirectional, to interface the 8255 chip to databus of 8086) ➢ Read write logic (to manage data transfer between 8086 and 8255 for read/write) ➢ Two address pins 0f 8086 are connected to A0 and A1 pins of 8255 to select port and control register.		
VIII b	Modes of operation of 8255 (Mode 0, Mode 1, Mode 3) Mode 0 → • basic input/output mode • (8 bit ports A & B, 4bit ports port C upper & port C lower • Any port can be input / output • Outputs are latched, inputs are not Mode 1→ • strobed input/output or handshaking mode • (ports A & B can be input / output, • Port C used for hadshaking signals • Input/output are latched • Interrupt logic is supported) Mode 2 → • strobed bi-directional bus • Bidirectional port –port A • Control port –port C • Input/output are latched • Handshaking are provided	List (1) + 3*2 =6 Total = 7	

Scoring Indicators

Code: 5131

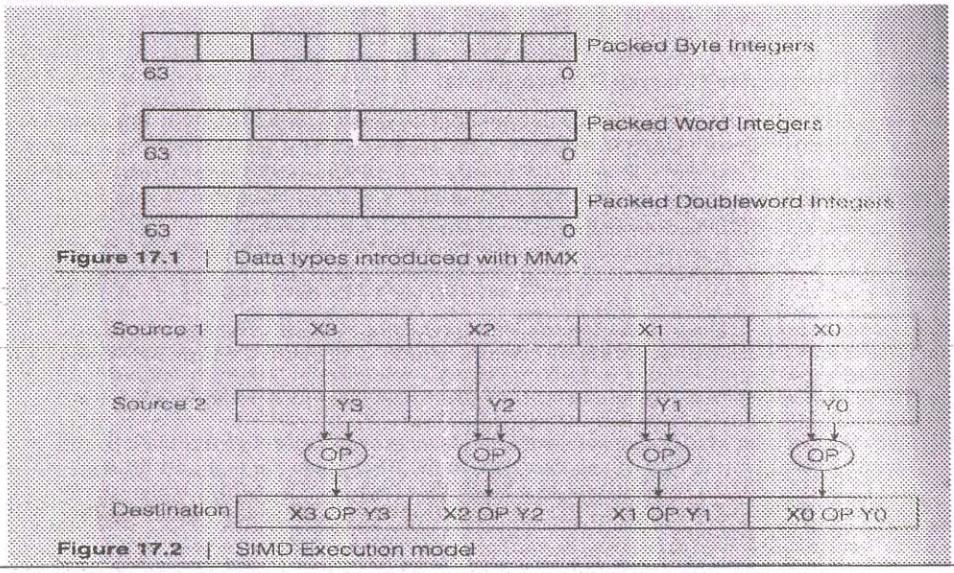
IX a	**80386** There are 2 modes of operation in 8086 **Real mode**(addressable memory **protected mode** (Protected virtual addressing mode)of operation **In real mode** – 80386 act as 8086 – On start up, the system is in real mode. – Only 2^20 =1MB physical memory can be accessed. So it acts as fast 8086. – Addres calculation =segment register+offset (20 bits) **In protected virtual addressing mode** – Addressable physical memory = 2^32 = 4gb – Features in this mode • Memory management ❖ Address calculation in memory • Protection ❖ (data and code of one user is isolated from others) • Multitasking→multiple tasks can be executed at the same time • Interrupt handling→ same as 8086. Instead of IVT(Interrupt Vector table) ,Interrupt Description Table is used	4+4	15
IX b	PENTIUM FEATURES • 32 bit processor (32 bit address bus, 64 bit data bus) • Internal calculation 32 bit, external transfer 64 bit • It is based on Superscalar architecture • It has 2 integer pipelines and 1 floating point unit • Has Separate code cash(8 K), data cash (8K) • Less than one clock cycles per instructions • Branch prediction → during pipelined execution the processor can predict the branch address based on previous history. It improves performance. 5 stage pipeline  Figure 15.1 | A five-stage pipeline Instruction fetch, Instruction decode, Instruction execute, Memory load/store, Write back output	4+3	

Code: 5131

X a	**Multi-core processing**
	• Concept of multi-core processing
	  • Replication /duplication of processing engine
	  • Multiple processing units(computing cores) on the same chip.
	  • Known as Chip multiprocessing(CMP)
	• **Multicore processor**
	  – Are **chips** in which **two or more cores are packed together**. Eg: dual-core pentiumD
	• **Core**
	  – Computing engine includes **ALU and register set**
	 
	
	• **Multicore processor can be homogeneous**
	  – Identical core on same chip
	or heterogeneous
	  – Non-identical cores on same chip
	Heterogeneous multicore processor
	Multicore processor can improve Performance
	• It Allows hyperthreading.
	  – Simultaneous multithreading→hyperthreading
	  – With multiple cores multiple threads can be executed simultaneously
	  – Thread/process → running program
	  – Eg: in dual core (two threads simultaneously one in each core)
		Any 4 points
		4X2
			15

Scoring Indicators

Code: 5131

X b	MMX Multimedia extension in Pentium , Add 8 64bit registers to Pentium Perform SIMD with in a register(SIMR) for single instruction multiple data (explanation 2, eg: 2) Datatypes in MMX  Figure 17.1 | Data types introduced with MMX Figure 17.2 | SIMD Execution model	4 3	

PICTURE QUALITY LOW