SCHEME OF VALUATION

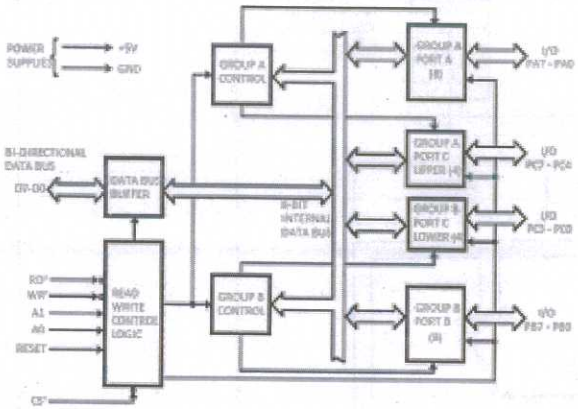
(Scoring Indicators)

Revision: 15 **Course Code:5131**

Course Title: MICROPROCESSORS AND INTERFACING

Qst. No.	Scoring Indicator	Split up score	Sub Total	Total
I	**PART A**			
1	RESB,RESW,RESD,RESQ Any two	2		
2	It points the next instruction to be fetch from memory	2		
3	CMPSB , MOVSB	2		
4	DIVIDE BY ZERO, SINGLE STEP, BREAKPOINT, OVERFLOW	2		
5	Superscalar architecture,	2		
II	**PART B**			
1	Features of 80386 • As it is a 32-bit microprocessor, It has 32-bit ALU. • 80386 has data bus of 32-bit. • It holds address bus of 32 bit. • It supports physical memory addressability of 4 GB and virtual memory addressability of 64 TB. • 80386 supports variety of operating clock frequency, which are 16 MHz, 20 MHz, 25 MHz and 33 MHz. • It offers 3 stage pipeline: fetch, decode and execute. As it supports simultaneous fetching, decoding and execution inside the system. • Operating modes of 80386 are real addressing mode ,PVAM, virtual 86 mode Any 6 points	6X1	6	6
2	Addressing modes Direct addressing Register addressing Immediate addressing Register indirect addressing Implicit addressing Any 4 with examples and desription	4X1.5	6	6
3	**SECTION .DATA** **MSG1 DB "ENTER A NUMBER",0AH,0DH** **LEN1 EQU $-MSG1** **MSG2 DB " NUMBER IS ODD" ,0AH,0DH** **LEN2 EQU $-MSG2** **MSG3 DB " NUMBER IS EVEN" ,0AH,0DH**			

	LEN3 EQU $-MSG3 **SECTION .BSS** **NUMBER RESB 2** **SECTION .TEXT** **GLOBAL _START** **_START:** **%MACRO DISPLAY 2** **MOV EAX, 4** **MOV EBX, 1** **MOV ECX, %1** **MOV EDX ,%2** **INT 80H** **%ENDMACRO** **;READ A NUMBER** **DISPLAY MSG1, LEN1** **MOV EAX, 3** **MOV EBX, 0** **MOV ECX,NUMBER** **MOV EDX ,2** **INT 80H** **MOV AL. [NUMBER]** **ROL AL,1** **JC L1** **DISPLAY MSG3, LEN3** **JMP EXIT1** **L1:** **DISPLAY MSG2, LEN2** **EXIT1:** **MOV EAX,1** **INT 80H**	6	6	6
4	The INTR is a maskable interrupt because the microprocessor will be interrupted only if interrupts are enabled using set interrupt flag instruction. It should not be enabled using clear interrupt Flag instruction. The INTR interrupt is activated by an I/O port. If the interrupt is enabled and NMI is disabled, then the microprocessor first completes the current execution and sends '0' on INTA pin twice. The first '0' means INTA informs the external device to get ready and during the second '0' the microprocessor receives the 8 bit, say X, from the programmable interrupt controller. These actions are taken by the microprocessor – • First completes the current instruction. • Activates INTA output and receives the interrupt type, say X. • Flag register value, CS value of the return address and IP value of the return address are pushed on to the stack.	6	6	6

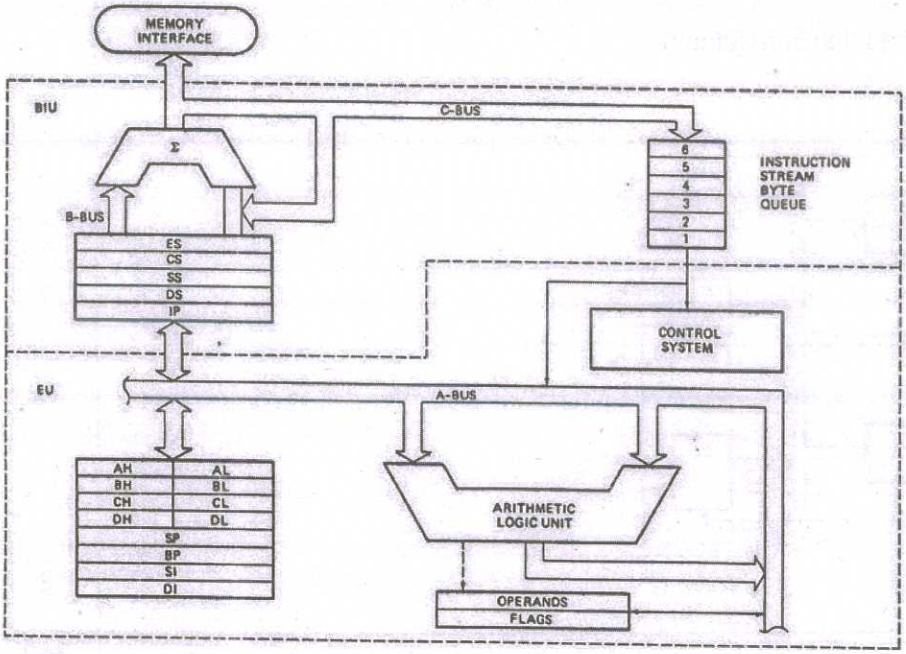
	• IP value is loaded from the contents of word location X × 4 • CS is loaded from the contents of the next word location. • Interrupt flag and trap flag is reset to 0			
5	 Explanation	4 2	6	6
6	• The instruction formats are: • **MUL reg8/mem8** • **MUL reg16/mem16** • The single operand is the multiplier. • The following table shows the default multiplicand and product, depending on the size of the multiplier: **multiplicand × multiplier = product**	3+3	6	6
7	A feature of certain Intel chips that makes one physical CPU appear as two logical CPUs.. 6 points based on these..	6	6	6

Multiplicand	Multiplier	Product
AL	*r/m8*	AX
AX	*r/m16*	DX:AX

PART C
III

Diagram
Explanation

a)



6
9

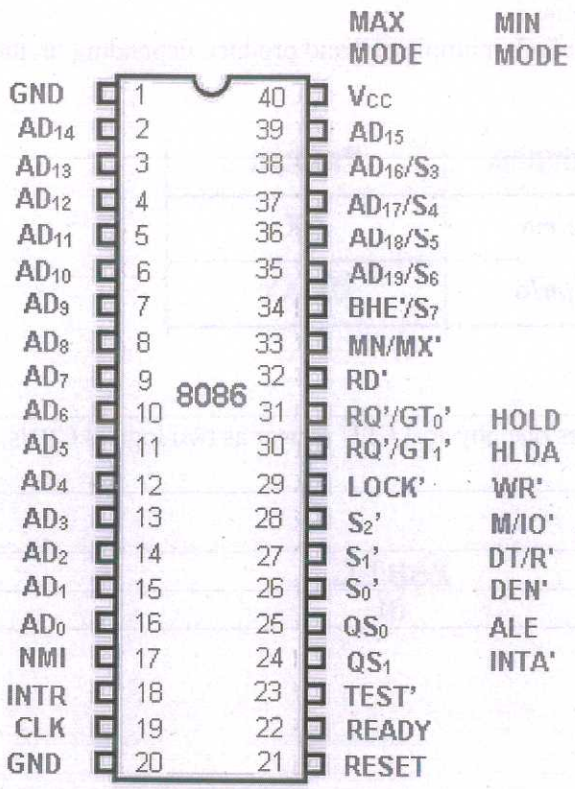
6

9

15

IV

a) Pin diagram



5

15

Minimum mode pins explanation 5

Maximum mode pins explanation 5

V				
a) These instructions are used to transfer the data from the source operand to the destination operand. Following are the list of instructions under this group – Instruction to transfer a word • **MOV** – Used to copy the byte or word from the provided source to the provided destination. • **PUSH** – Used to put a word at the top of the stack. • **POP** – Used to get a word from the top of the stack to the provided location. • **PUSHA** – Used to put all the registers into the stack. • **POPA** – Used to get words from the stack to all registers. • **XCHG** – Used to exchange the data from two locations. • **LEA** – Used to load effective address to the specified register • **PUSHF**- Pushes flag register on the stack • **POPF**- retrieve flag register from stack Any 7 instructions	7	7	15	15
b Algorithm 1. Load contents of variable declared in data section in register AL 2. Copy contents of register AL in register AH 3. Perform AND operation on register AL with 0F 4. Assign 04 to CL Register 5. Shift the contents of AH by executing SHR instruction using CL 6. Perform OR operation on register AX with 3030 7. Display the content of AX **Program –** **SECTION .DATA** **BCD_1 DB 45H** **SECTION .BSS** **ASCII_1 RESB 2** **ASCII_2 RESB 2** **SECTION .TEXT** **GLOBAL _START** **_START:** **%MACRO DISPLAY 2** **MOV EAX , 4** **MOV EBX , 1** **MOV ECX , %1** **MOV EDX ,%2** **INT 80H** **%ENDMACRO** **MOV AL. BCD_1** **MOV AH, AL** **AND AL,0FH** **SHR AH,4**	2 6	8		

```
0R AX,3030H
MOV ASCII_1, AH
MOV ASCII_2, AL
DISPLAY ASCII_1,2
DISPLAY ASCII_2,2
```

VI

a)

Program –

```
        SECTION .DATA
         MSG1 DB "ENTER A STRING",0AH,0DH
         LEN1 EQU $-MSG1
         MSG2 DB " ENTER A CHARACTER" ,0AH,0DH
         LEN2 EQU $-MSG2
         MSG3 DB "CHARACTER IS NOT PRESENT" ,0AH,0DH
         LEN3 EQU $-MSG3
         MSG4 DB "  CHARACTER IS PRESENT" ,0AH,0DH
         LEN4 EQU $-MSG4

        SECTION .BSS
        STR1 RESB 50
        A RESB 1
        COUNT RESB 2
        SECTION .TEXT
        GLOBAL _START
_START:
%MACRO DISPLAY 2
MOV EAX, 4
MOV EBX, 1
MOV ECX, %1
MOV EDX ,%2
INT 80H
%ENDMACRO
DISPLAY MSG1, LEN1                    ; READ STRING
MOV EAX, 3
MOV EBX, 0
MOV ECX, STR1
MOV EDX ,50
INT 80H
MOV [COUNT], EAX
DISPLAY MSG2, LEN2                    ;  READ A CHARACTER
MOV EAX, 3
MOV EBX, 0
MOV ECX, A
MOV EDX ,1
INT 80H

MOV EDX,00H
MOV ECX, COUNT
MOV AL, A
L1:
  CMP BYTE [STR1+EDX], AL             ; COMPARE EACH CHAR FOR EQUITY
  JE EXIT1                            ; IF EQUAL EXIT FROM LOOP
```

9 9 15

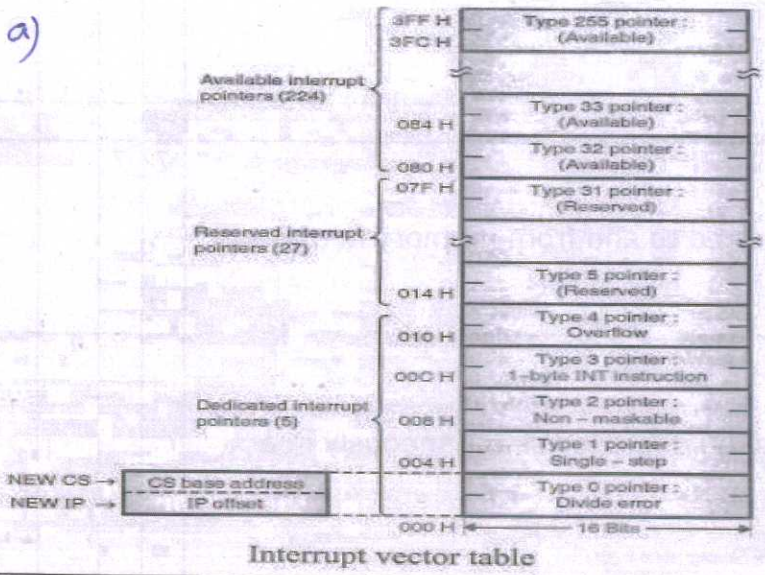
```
  INC EDX
  LOOP L1                          ; CONTINUES UNTIL COUNT=0
  DISPLAY MSG3, LEN3
  JMP L2
EXIT1:
  DISPLAY MSG4, LEN4
L2:
  MOV EAX,1
  INT 80H
```

JC	address	Used to jump if carry flag (CY = 1)	**6**	6		
JNC	address	Used to jump if no carry flag (CY = 0)				
JE/JZ	address	Used to jump if equal/zero flag ZF = 1				
JNE/JNZ	address	Used to jump if not equal/zero flag ZF = 0				
JGE/ JNL	address	Used to jump if carry flag CY = 1				
JL/ JNGE	address	Used to jump if less than/not greater than/equal instruction satisfies.				
JLE/ JNG	address	Used to jump if less than/equal/if not greater than instruction satisfies				
Any 6						

a) Branch instructions

VII

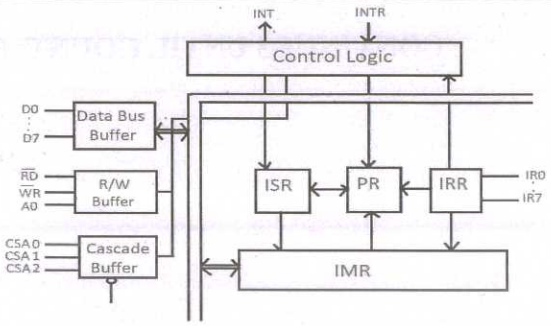
a)



Interrupt vector table

6	6		

b)



8259 block diagram
Explain each block

4
9
5

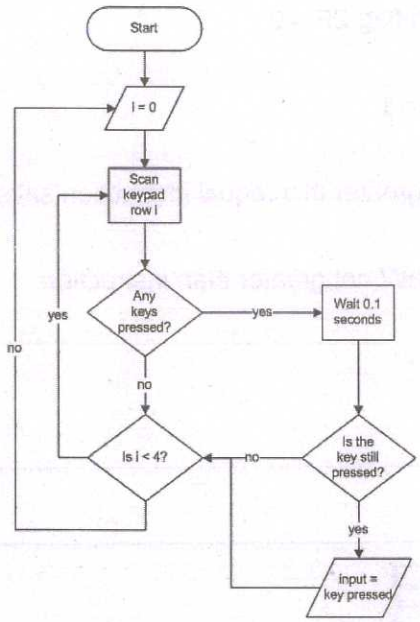
VIII

a)
Input output mode
Bit set reset (BSR) mode
Explain two modes

4
3

7

b)



8

15 15

IX

a) Features of Pentium Processor are as follows:

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 pipe lining
4. Instruction cache
5. Two parallel integer execution units
6. Allows the execution of two instructions to be executed simultaneously in a single processor clock
7. Floating point unit
8. Super scalar architecture
9. Three execution units
10. One execution unit executes floating point instructions

write any 7

7

7

15

b)

i)Superscalar architecture is a method of parallel computing used in many processors. In a **superscalar** computer, the central processing unit (CPU) manages multiple instruction pipelines to execute several instructions concurrently during a clock cycle.

ii)**MMX** is a Pentium microprocessor from Intel that is designed to run faster when playing multimedia applications. According to Intel, a PC with an **MMX** microprocessor runs a multimedia application up to 60% faster than one with a microprocessor having the same clock speed but without **MMX**.

4 4 8

Operating modes of 80386

80386 supports 3 operating modes: real, protected and virtual real mode.

Of the two modes of 80286 microprocessor, initially the 80286 was booted in real mode. However, to have better operating performance, separate software command is used to switch from the real mode to the protected mode.

But it requires the resetting of microprocessor in order to switch to real mode from protected mode. This drawback was eliminated in 80386 that allows the switching between the modes using software commands.

In the **protected mode**, 80386 microprocessor operates in similar way like 80286, but offers higher memory addressing ability.

In **virtual mode**, the overall memory of 80386 can be divided into various virtual machines. And all of them acts as a separate computer with 8086 microprocessor. This mode is also called virtual 8086 mode or V86 mode

3 3 9 3 15 15

In technologies,**multi-core** is usually the term used to describe two or more CPUs working together on the same chip. Also called **multicore** technology, it is a type of architecture where a single physical **processor** contains the **core** logic of two or more **processors**.

6 6