Roll no:

CS321: Miterm Examination

Q1. In C, type `char` is a signed, 8-bit integer. With that in mind, what does this output?

```
char a = 100;
char b= 30;
char c= a + b;
printf( "%d\n", c); //
```

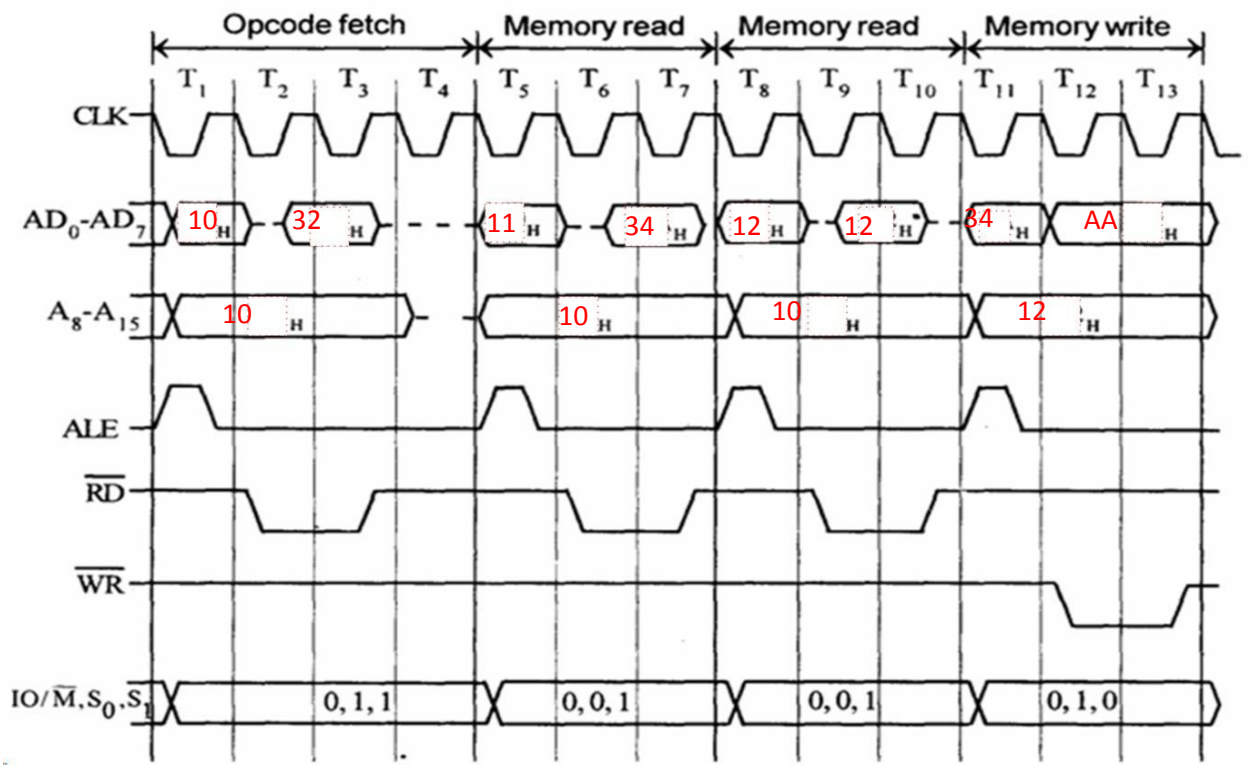
(5 points)

Ans:

The maximum signed 8-bit value is 011111112 = 1 + 2 + 4 + 8 + 16 + 32 + 64 = 127, so this sum overflows. On overflow, we wrap to the most negative value, -128 and the count up
from there,, so 127+3 results in -126.

Q2: Figure shows the timing diagram of STA 1234H instruction in 8085. Assuming that Accumulation contains AA_H and instruction is stored in memory starting 1010 H, which contains opcode 32H, fill the content of Address and Data bus in the timing diagram (places marked **H**)

(5 Points)



Memory:

1010: 32
1011: 34
1012: 12

Q3. State whether the following sentence is True or false. If it is false or True explain?. **Also fill the blanks if applicable.** **(20 points)**

a)Reading memory from the heap is slower than reading from a local variable allocated on the stack. This statement is False…… (explanation write in the main answer book)

b)An x86 program which uses **lea** instructions can be translated to a functionally equivalent version which does not use any lea instructions. The equivalent instruction is
……
Answer
.data
a db 1, 2,3,4
lea si,a is equivalent to
mov si, a

(c) Check A and B are equivalent declaration
A: line **DB** 5, 4, 3 DUP(2, 3 DUP(0), 1)
B: line **DB** 5, 4, 2, 0, 0, 0, 1, 2, 0, 0, 0, 1, 2, 0, 0, 0, 1

Ans: They are equivalent

d) The two address lines, along with CS signal, determine the selection of a particular port or control register in an 8255.

e) In an 8254, there are two 16-bit counter registers, each of which can be programmed as a timer or an event counter. This statement is False

f)In a x86-32 architecture pointers point to locations in memory that are multiples of 32 bits apart. This statement is …False……

g)Assume that 8155 programmed as counter, A high reset input resets the counter. To restart counting after resetting, a START command is not required through the control register. This statement is ……False……

h) 8259s can be cascaded in master-slave configuration and to have **64** levels of interrupts

i) 8155 has 8-bit …… **256**……word RAM memory

j) X86 assembly store the return value is always in ……eax…… when a function is finished

k)Function performed by CALL address instruction is equivalent to …… push ip …(instruction).. followed by jmp address……(instruction)..

l) A **ret** instruction is equivalent to **pop ip**

h)The INT 10H instruction (8086) calls the interrupt service procedure whose address is stored beginning at memory location ……

Ans: 40H

Q4:

Consider the following C code.

```
if (al > bl) AND (bl > cl)
X = 1;
```

The following assembly code was generated by one student. Check whether the implementation is correct and fill blanks the comments

```
        cmp al,bl     ;…….
        ja L1
        jmp next
L1:     cmp bl,cl     ;…….
        ja L2
        jmp next
L2:     mov X,1       ;……

next:
```

(b) Give an alternative assembly implementation (reduced number of instructions) of the above C statements

Ans: (one sample solution)

```
cmp al,bl ; first expression...
jbe next ; quit if false
cmp bl,cl ; second expression...
jbe next ; quit if false
mov X,1 ; both are true
```

next:

(c) Convert the following assembly in equivalent C statement code

```
cmp al,bl
ja L1
cmp bl,cl
jbe next
L1:mov X,1
next:
```

Ans:

if (al > bl) OR (bl > cl)

X = 1

Q5)

```
.data
array1 DB 10h,20h,30h
array2 DW 1000h,2000h,3000h
.code
mov esi,OFFSET array1
mov al,[esi]                ; AL = 10h
inc esi
mov al,[esi]                ; AL = 20h
inc esi
mov al,[esi]                ; AL = 30h
mov esi,OFFSET array2
mov ax,[esi]                ; AX = 1000h
```

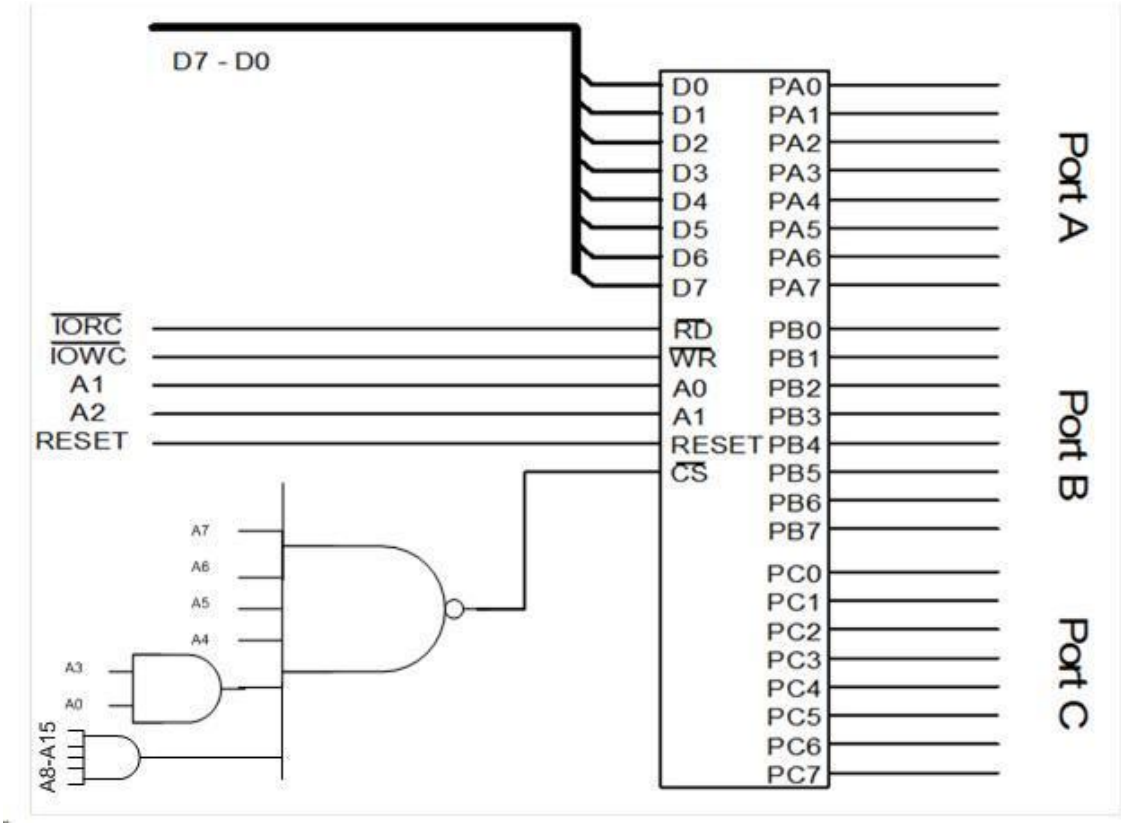
```
add esi,2
mov ax,[esi]                ; AX = 2000h
add esi,2
mov ax,[esi]                ; AX = 3000h
```

Assume that

j DW 10 ; i DW 20; Check the following instructions legal/ illegal Complete the table. Fill legal/ illegal in the respective row.:

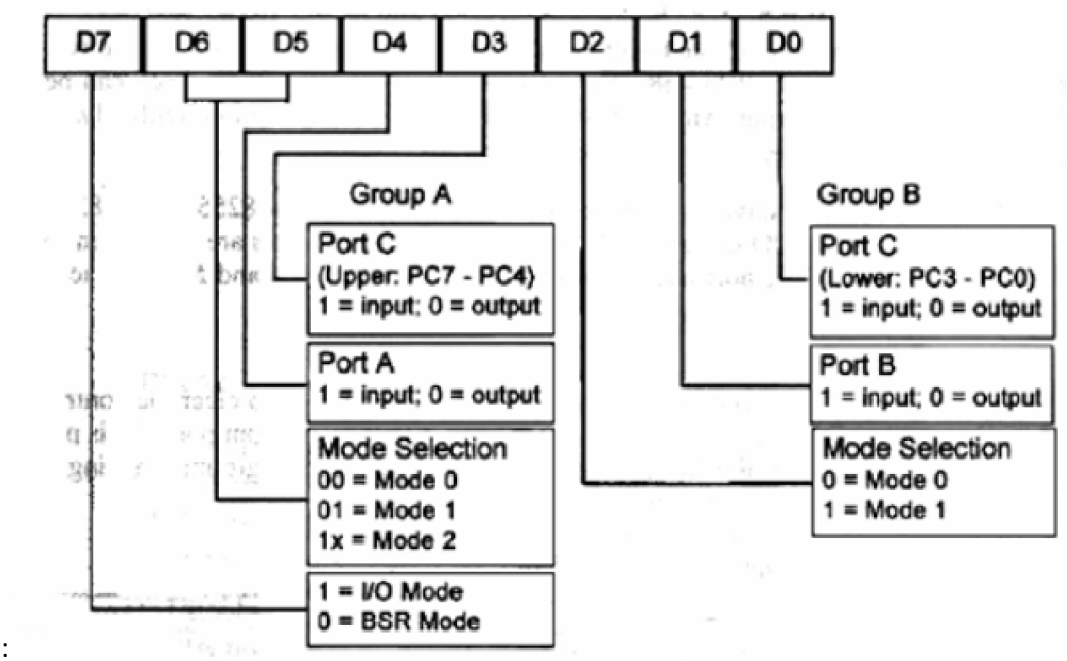
	instruction	comment
1	MOV AX, BL	illegal
2	MOV AL, BL	legal
3	MOV AH, BL	legal
4	MOV i, AL	legal
5	MOV AL, i	legal
6	MOV i, j	illegal
7	ADD 2, AX	illegal
8	ADD AX, 2	legal
9	MOV AL, j	illegal
10	MOV AL, Byte PTR j	legal

Q7: In a given 8086 -based system, 8255 interfaced as shown and its PORT A used for monitoring the temperature. Write Assembly language instructions to monitor that port A continuously for the temperature of 100 degrees. If it reaches 100, sends it to port address B (use appropriate control word) (10 Points)



Port Addresses are:

Port A: FFF9
PortB: FFFB
PortC: FFFD
Control: FFFF



:

Ans:

```
        Mov DX, FFFF
        MOV AL,90
        OUT  DX, AL
        MOV DX, FFF3
Wait:   IN, AL,DX
        CMP A, 64H
        JNZ wait
        MOV DX, FFF7
        OUT DX,AL
```

Q8 (a) **83h**

Q8 (b) Write an assembly language program that toggles the bits of Port A continuously in every 0.5. Assume that each instruction take one cycle. The processor operates at 1 MHz. Show delay calculations clearly **(5 Points)**

Ans:

```
Mov DX, FFFF
MOV AL,90
OUT  DX, AL
MOV DX, FFF3
back :  MOV AL, 55
        OUT DX,AL
        MOV AL,AA
        OUT DX,AL
        JMP back
```

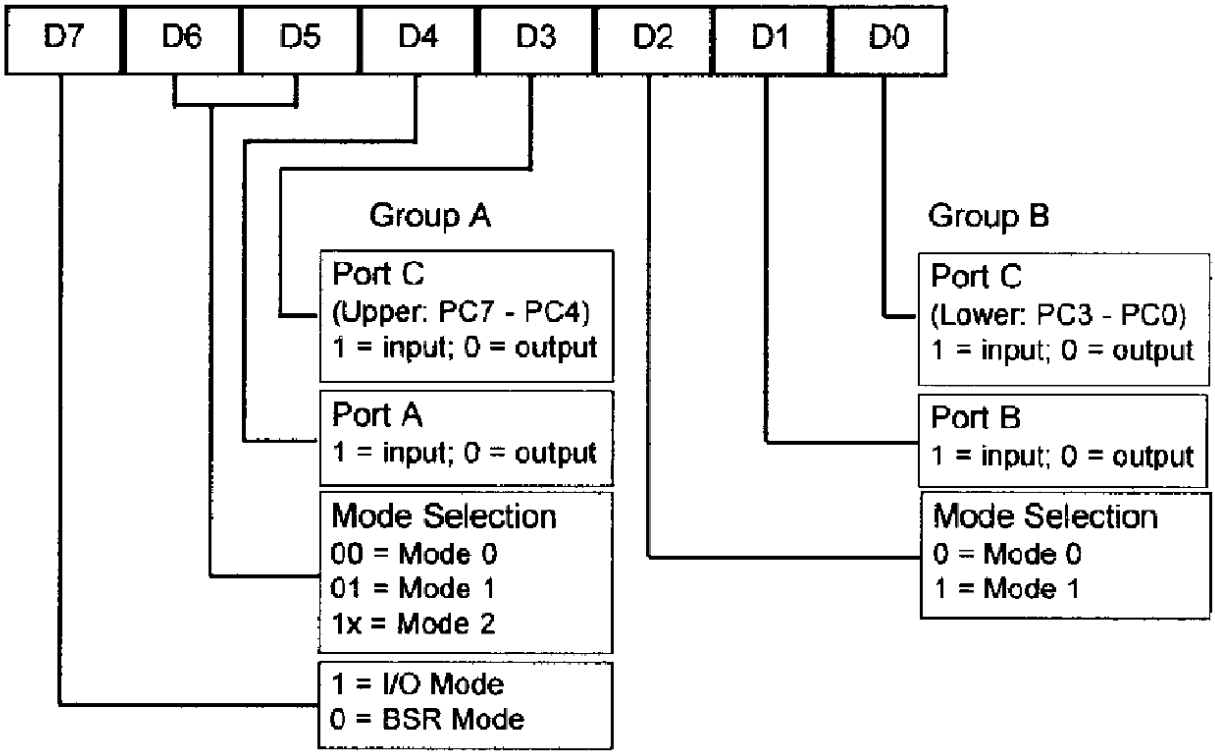
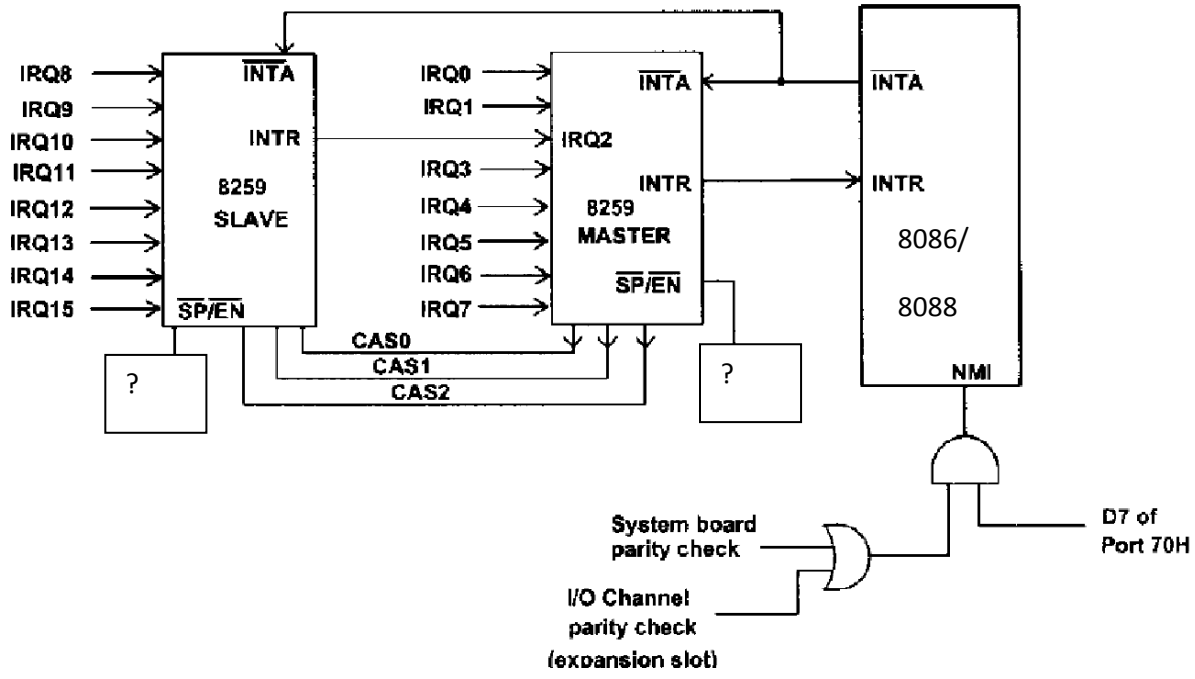
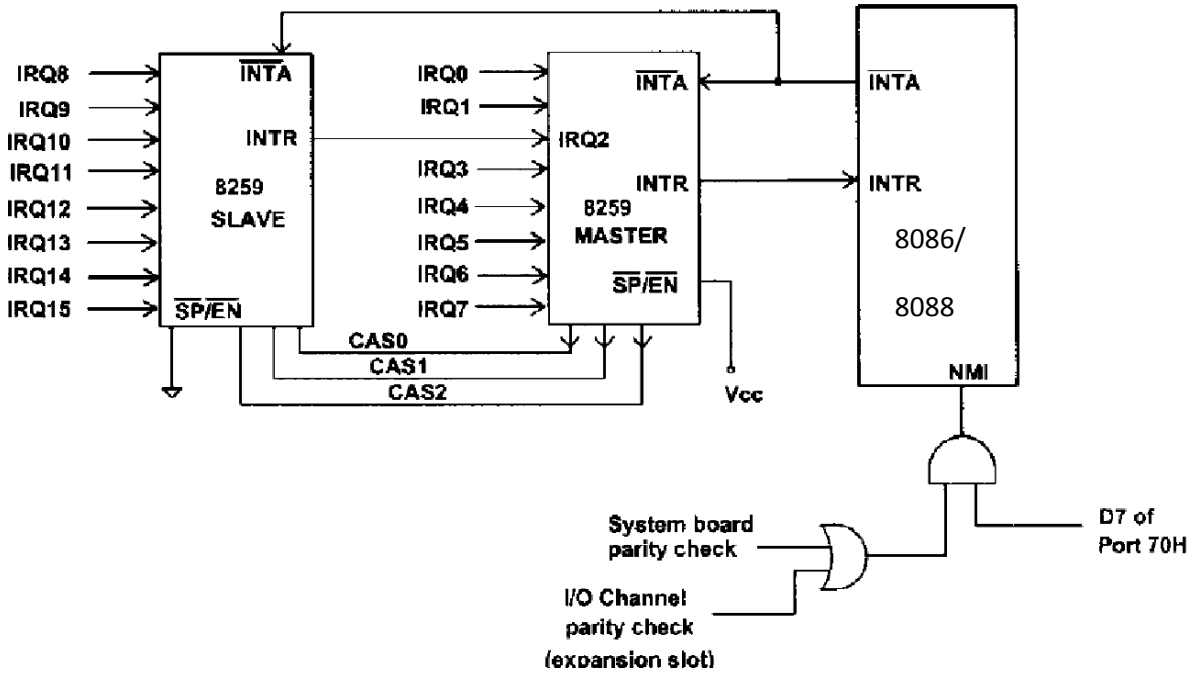


Figure shows an IBM PC, interrupt configuration. In figure mark the priority of the interrupts from (1 to 16). Also fill the box with ? what should be logic applied (logic 0 or logic 1) for this configuration.



ANS:


IRQ8
IRQ9
IRQ10
IRQ11
IRQ12
IRQ13
IRQ14
IRQ15
INTA
INTR
8259
SLAVE
SP/EN
IRQ0
IRQ1
IRQ2
IRQ3
IRQ4
IRQ5
IRQ6
IRQ7
INTA
INTR
8259
MASTER
SP/EN
CAS0
CAS1
CAS2
Vcc
INTA
INTR
8086/
8088
NMI
System board
parity check
I/O Channel
parity check
(expansion slot)
D7 of
Port 70H