SOLVED CURRENT FINAL PAPERS CS401

(Solved by AQSA & AIRBORNE)

(yellow highlighted are the unsolved questions and red highlighted are the ones I got in my exam 28-2-2013 from this file)

Q3 names of registers that cannot perform arithmetical operation?

 Direction, interrupt, trap

which instruction a Trap flag can be set ?

(ans : no instruction we use pushf and pop f

wat are the register (i.e not scratch register) chapter 16 ?

EAX, ECX, EDX, FS, GS, EFLAGS, and any other
registers.
…………………..
0% MCQs were from Past Papers
2 No Questions
How many types of interrupt are used for debugging? Write down the names.
INT 1, Trap, Single step Interrupt
This interrupt is used in debugging with the trap flag.
INT 3, Debug Interrupt

Q.   Give an example of Base+Index+Offset addressing mode.

Ans. The values of the base register, the index register, and the constant offset are all added together to get the effective address. For example "mov [bx+si+300], ax" moves the word contents of the AX register to the word in memory starting at offset attained by adding BX, SI, and 300 in the current data segment. Default segment association is again based on the base register. It might be used with the array base of a two dimensional array as the constant offset, one dimension in the base register and the other in the index register. This way all calculation of location of the desired element has been delegated to the processor.

Q.   Which DOS service of INT 21 is used to read a charecter from keyboard?
Ans.
INT 21 - READ CHARACTER FROM STANDARD INPUT, WITH ECHO

AH = 01h
Return: AL = character read

Q3No Question
For which purpose INT 3 is used and describe the funnctionality of this interrupt.

Ans. INT 3, Debug Interrupt .Apart from single stepping debugger has the breakpoint feature. INT 3 is used for this feature. INT 3 has a single byte opcode so it can replace any instruction. To put a breakpoint the instruction is replaced with INT 3 opcode and restored in the INT 3 handler. The INT 3 opcode is placed again by a single step interrupt that is set up for this purpose after the replaced instruction has been executed.

How data movement instructions writtn by SUN Spark processor? Defend your answer by providing two exampls.

Ans. The Sun SPARC is a very popular processing belonging to the RISC (reduced instruction set computer) family of processors. SPARC introduces a concept of register window. SPARC instructions have two sources and a distinct destination. This allows more flexibility in writing programs .e.g

Data Movement
 LDSB [rn], rn(load signed
byte) LDUW [rn], rn (load
unsigned word)
STH [rn], rn (store half
word)
Arithmetic
source1 = rn
source2 = rn or simm13
dest = rn
ADD r2, r3, r4
SUB r2, 4000, r5
SLL, SRA, SRL (shifting)
AND, OR, XOR (logical)

Q   How Nomenclature can be observed in computer architecture? Explain with examples.

Ans. Registers are more than one in number, so we have to precisely identify or name them. Some manufacturers number their registers like r0, r1, r2, others name them like A, B, C, D etc. Naming is useful since the registers are few in number. This is called the nomenclature of the particular architecture. Still other manufacturers name their registers according to their function like

X stands for an index register. This also informs us that there are special functions of registers as well, some of which are closely associated to the particular architecture. For example index registers do not hold data instead they are used to hold the address of data.

Q.   Why are device drivers necessary given that BIOS already has code that communicate with the hardware?

Ans. We will be using BIOS disk services to directly see the data stored in the directory entries by DOS. For this purpose we will be using the BIOS disk services.
INT 13 - DISK - RESET DISK SYSTEM
AH = 00h
DL = drive
Return:
CF = error flag
AH = error code

Q No 5 Questions
How many pins are used for serial port communication and which type of communication occur through serial port?

Ans. Serial port is a way of communication among two devices just like the parallel port. the bits are sent one by one on the serial port in a specially formatted fashion. The serial port connection is a 9pin DB-9 connector with pins assigned

Q.   At which location Interrupt Gate Discriptor Table located? Describe the structure of the table.
Handling interrupts in protected mode is different. Instead of the IVT at physical address 0 there is the IDT (interrupt descriptor table) located at physical address stored in IDTR, a special purpose register. The IDTR is also a 48bit register similar in structure to the GDTR and loaded with another special instruction LGDT. The format of the interrupt descriptor is as shown Below. Table from pg 182

The P and DPL have the same meaning as in data and code descriptors.
The S bit tells that this is a system descriptor while the 1110 following it tells that it is a 386 interrupt gate.

INT10 of video services BH, AL, CX, DX indicate what?
INT 10 - VIDEO - WRITE GRAPHICS PIXEL
AH = 0Ch
BH = page number
AL = pixel color
CX = column
DX = row

Q   In IAPX88 processor what limitations are there for multiplying a 16-bit number by 2 and which operation can be used to resolve multiplication?

Ans.   There are no specific limitations IAPX88 processor only provides us the basic building blocks. We build a plaza out of these blocks, or a building, or a classic piece of architecture is only dependent upon our imagination. With our logic we can extend these algorithms as much as we want.
SHL operation can be used to resolve the multiplication.

Q.   Differences b/w Motorola 68k and x86?

Ans. Motorolla 68K processors are very similar to Intel x86 series in their architecture and instruction set. 68K processors have 16 23bit general purpose registers named from A0-A7 and D0-D7. A0-A7 can hold addresses in indirect memory accesses. These can also be used as software stack pointers. Stack in 68K is not as rigit a structure as it is in x86. Motrolla processors allow bit addressing, that is a specific bit in a byte or a bit field, i.e. a number of bits can be directly accessed. 68K allows indirect memory access using any A register. 68K has no segmentation; it however has a paged memory model.

Q2. how interrupt pointer register work when call instruction is executed ?

Q3.given instruction is wrong write this in correct format mov DX, 0X123?

mov dx, 0x0123

Q4. Data movement instruction of sun spark?5marks

Data Movement
LDSB [rn], rn (load signed byte)
LDUW [rn], rn (load unsigned word)
STH [rn], rn (store half word)

Q5.write instruction to set frequency at 1 ms?5marks

mov ax, 1100

out 0x40, al
mov al, ah
out 0x40, al

Q6.   Code of c language is given and asking for write the given code in assembly language?5mark

Q7.which pin of db 9 connector assign to ring indicator?2 marks

9 pins

Write the instruction of following 5marks?

Copy BL into CL
Answer: mov cl, bl
 Copy DX into AX
Answer: mov ax, dx
Store 0x12 into AL
Answer: mov al, 0x12
Store 0x1234 into AX
Answer: mov ax, 0x1234
Store 0xFFFF into AX
Answer: mov ax, 0xFFFF

How to create or Truncate File using INT 21 Service?5marks

INT 21 - CREATE OR TRUNCATE FILE
AH = 3Ch
CX = file attributes
DS:DX -> ASCIZ filename
Return:
CF = error flag
AX = file handle or error code

Q What is Triple fault?

A **triple fault** is a special kind of exception generated by the CPU when an exception occurs while the CPU is trying to invoke the double fault exception

handler, which itself handles exceptions occurring while trying to invoke a regular exception handler

2 instructions were given output btani thi.

<mark>int for receiving character from keyboard</mark>

mode for high resolution.VESA VBE 2.0

3 marks questions:

Q short jump.

Ans.   If the offset is stored in a single byte as in 75F2 with the opcode 75 and operand F2, the jump is called a short jump. F2 is added to IP as a signed byte. If the byte is negative the complement is negated from IP otherwise the byte is added. Unconditional jumps can be short, near, and far. The far type is yet to be discussed. Conditional jumps can only be short. A short jump can go +127 bytes ahead in code and -128 bytes backwards and no more. This is the limitation of a byte in singed representation.

<mark>int 16 services</mark>

5 marks questions:

Bios services

BIOS services are very low level. A level further lower is only directly controlling the hardware. BIOS services provide a hardware independent layer above the hardware and OS services provide another higher level layer over the BIOS services. The layer of BIOS provides services like display a character, clear the screen, etc. All these layers are optional in that we can skip to whatever lower layer we want. BIOS exports its various services through different interrupts. Keyboard services are exported through INT 16, parallel port services through INT 17 and similarly others through different interrupts. The BIOS INT 14 provides serial port services. We will use a mix of BIOS services and direct port access. Important BIOS services regarding the serial port are discussed below.
INT 14 - SERIAL - INITIALIZE PORT
AH = 00h
AL = port parameters
DX = port number (00h-03h)
Return:
AH = line status
AL = modem status

Q program to draw half line

INT 10 - VIDEO - WRITE GRAPHICS PIXEL

AH = 0Ch
BH = page number
AL = pixel color
CX = column
DX = row
Example 12.3
; draw line in graphics mode
[org 0x0100]
mov ax, 0x000D ; set 320x200 graphics mode
int 0x10 ; bios video services
mov ax, 0x0C07 ; put pixel in white color
xor bx, bx ; page number 0
mov cx, 200 ; x position 200
mov dx, 100 ; y position 200
l1: int 0x10 ; bios video services
dec dx ; decrease y position
loop l1 ; decrease x position and repeat
mov ah, 0 ; service 0 – get keystroke
int 0x16 ; bios keyboard services
mov ax, 0x0003 ; 80x25 text mode
int 0x10 ; bios video services
mov ax, 0x4c00 ; terminate program
int 0x21

Q1: Suppose we want to execute multiple process at the same time for this purpose we need to store and restore states of processing running immediately operating system which process is used to store and restore states process in a multiple operating system

Q2: which register is used by GDT to store its base and limits
The base and limit of GDT is stored in a 48bit register called the GDTR. This register is loaded with a special instruction LGDT and is given a memory address from where the 48bits are fetched.

Q3: which interrupt is used to extend read disk and have can we use extended read service also mention all the attributes which are used in this service
INT 13 - INT 13 Extensions - EXTENDED READ
AH = 42h
DL = drive number
DS:SI -> disk address packet
Return:
CF = error flag
AH = error code
disk address packet's block count field set to number of blocks
successfully transferred

The format of the disk address packet used above is as follows.
Offset Size Description

00h BYTE size of packet = 10h
01h BYTE reserved (0)
02h WORD number of blocks to transfer
04h DWORD -> transfer buffer
08h QWORD starting absolute block number

Q4: write an assembly program for drawing a line in graphics mode of video service

ANS: ; draw line in graphics mode
[org 0x0100]
mov ax, 0x000D ; set 320x200 graphics mode
int 0x10 ; bios video services
mov ax, 0x0C07 ; put pixel in white color
xor bx, bx ; page number 0
mov cx, 200 ; x position 200
mov dx, 200 ; y position 200
l1: int 0x10 ; bios video services
dec dx ; decrease y position
loop l1 ; decrease x position and repeat
mov ah, 0 ; service 0 – get keystroke
int 0x16 ; bios keyboard services
mov ax, 0x0003 ; 80x25 text mode
int 0x10 ; bios video services
mov ax, 0x4c00 ; terminate program
int 0x21

Q5: at which locations interrupts gate descriptor table is stored and describe the structure of the table with an example

Ans.   The IDT (interrupt descriptor table) located at physical address stored in IDTR, a special purpose register. The IDTR is also a 48bit register similar in structure to the GDTR and loaded with another special instruction LGDT. The format of the interrupt descriptor is as shown below.
Pg 182 table

Q6: for what purpose INT 1 is reserved

INT 1, Trap, Single step Interrupt
This interrupt is used in debugging with the trap flag. If the trap flag is set the Single Step Interrupt is generated after every instruction. By hooking this interrupt a debugger can get control after every instruction and display the registers etc. 8088 was the first processor that has this ability to support debugging.

Q7: In c and Pascal calling conventions which register are used as scratch whenever we call a functions

In C and Pascal calling conventions, both standards do not preserve or guarantee the value of EAX, ECX, EDX, FS, GS, EFLAGS, and any other registers.

Q8: how to identify maximum possible speed of multitasking

When the program is executed the threads display the numbers independently. However as keys are pressed and new threads are registered, there is an obvious slowdown in the speed of multitasking. To improve that, we can change the timer interrupt frequency. The following can be used to set to an approximately 1ms interval.

mov ax, 1100
out 0x40, al
mov al, ah
out 0x40, al

This makes the threads look faster. However the only real change is that the timer interrupt is now coming more frequently.

2 Marks Questions

Q1 - what is the function naming convention in language C ?

Two prevalent calling conventions are the C calling convention and the Pascal calling convention. C prepends an underscore to every function or variable name while Pascal translates the name to all uppercase. C++ has a weird name mangling scheme that is compiler dependent. To avoid it C++ can be forced to use C style naming with extern "C" directive

Q2 - write instruction to allocate space for 32Bit PCB?

pcb: times 32*16 dw 0 ; space for 32
PCBs

Q3 - what are the roles of segment-selector and segment-descriptor?

Role of selector is to select on descriptor from the table of descriptors and the role of descriptor is to define the actual base address.

Q4 - Segment and Offset address in case of FAR Jump?

Far jump is not position relative but is absolute. Both segment and offset must be given to a far jump. Sometimes we may need to go from one code segment to another, and near and short jumps cannot take us there. Far jump must be used and a two byte segment and a two byte offset are given to it. It loads CS with the segment part and IP with the offset part. Execution therefore resumes from that location in physical memory.

3MarksQuestions
Q1 - Information through descriptor's first 16 Bits used to represent what ?

In descriptor table D bit defines that this segment is to execute code is 16bit mode or 32bit mode.

Q 2 - can we increase the speed of multitasking. if yes then how?
Yes, When the program is executed the threads display the numbers independently. However as keys are pressed and new threads are registered, there is an obvious slowdown in the speed of multitasking. To improve that, we can change the timer interrupt frequency. The following can be used to set to an approximately 1ms interval.
mov ax, 1100
out 0x40, al
mov al, ah
out 0x40, al
This makes the threads look faster. However the only real change is that the timer interrupt is now coming more frequently.

Q3 - Purpose of Global Descriptor Table (GDT) ?

The Global Descriptor Table or GDT is a data structure used by Intel x86-family processors starting with the 80286 in order to define the characteristics of the various memory areas used during program execution, including the base address, the size and access privileges like executability and writability. These memory areas are called segments in Intel terminology.

The GDT can hold things other than segment descriptors as well. Every 8-byte entry in the GDT is a descriptor, but these can be Task State Segment (or TSS) descriptors, Local Descriptor Table (LDT) descriptors, or Call Gate descriptors. The last one, Call Gates, are particularly important for transferring control between x86 privilege levels although this mechanism is not used on most modern operating systems.

4 - How many bytes will be moved?
- Mov cx, 384
REP movsb
CX register is set to 384, so counter will run 384 times each time moving single byte, so 384 bytes will be moved.
- Mov cs, 384
REP movsw
CX register is set to 384, so counter will again run 384 times, but this time complete word will be moved as REP movsw; therefore, 384 X 2 = 768 bytes will be moved

5 Marks Questions
1 - Define Coordination between data bus, control bus and address bus?
The group of bits that the processor uses to inform the memory about which element to read or write is collectively known as the *address bus*. Another important bus called the *data bus* is used to move the data from the memory to the processor in a read operation and from the processor to the memory in a write operation. The third group consists of miscellaneous independent lines used for control purposes. For example, one line of the bus is used to inform the memory about whether to do the read operation or the write operation. These lines are collectively known as the *control bus*. These three buses are the eyes, nose, and ears of the processor. It uses them in a

synchronized manner to perform a meaningful operation. Although the programmer specifies the meaningful operation, but to fulfill it the processor needs the collaboration of other units and peripherals. And that collaboration is made available using the three buses.

2 - Values required for the following registers to write a character on serial port using INT 21?
- AH =
- AL =
- DX =
INT 14 - SERIAL - WRITE CHARACTER TO PORT
 AH = 01h
 AL = character to write
 DX = port number (00h-03h) (04h-43h for Digiboard

3 - Data movement and arithmetic operation in Motorola 68k processor?
Data Movement
EXG D0, D2
MOVE.B (A1), (A2)
MOVEA (2222).L, A4
MOVEQ #12, D7
Arithmetic
ADD D7, (A4)
CLR (A3) (set to zero)
CMP (A2), D1
ASL, ASR, LSL, LSR, ROR, ROL, ROXL, ROXR (shift operations)

Q What is the format of interrupt gate descriptor?

pg 128

(1)   Which instruction is used to determine zero flag. 2 marks

JNZ instruction is used to determine the zero flag. Jump is taken is zero flag is set

(2)   Which memory space is needed to store all the task of the register. 2 marks

PCB

(3)   How many types of descriptors also mention their names. 3 marks

IDT can have 3 types of descriptors; interrupt gate, trap gate and task gate
3 types of descriptor tables.
GDT(global), LDT(local), IDT(interrupt)

(4) in read from file int21 service the error flag and file handling goes in which register and flag. 3marks

Bx = file handle
Cf = error flag

(6) how many flags are not ues in arithmetic operation and also write their names.3 marks

Direction, interrupt, trap

(7)n INT 10 of video services while writing graphics pixel, what do the values of registers BH, AL, CX, and DX corresponds to? 5mrks

INT 10 - VIDEO - WRITE GRAPHICS PIXEL

AH = 0Ch
BH = page number
AL = pixel color
CX = column
DX = row

Q1:   Describe the structure of "Device Driver File" which is used for device management?   2 Marks

ANS: Device driver file starts with a header containing a link to the next driver in the first four bytes followed by a device attribute word. The most important bit in the device attribute word is bit 15 which dictates if it is a character device or a block device. If the bit is zero the device is a character device and otherwise a block device. Page 158

Q2:   Why trap flag has a special relation with interrupt and debugger?   2 Marks

ANS: INT 1 is used in debugging with the trap flag. If the trap flag is set the Single Step Interrupt is generated after every instruction. So INT 1 relates to trap flag.

Q3:   Whether the relation b/w RET and CALL is dependent or independent?   2 Marks

ANS: Both the instructions are commonly used as a pair, however technically they are independent in their operation. The RET works regardless of the CALL and the CALL works

regardless of the RET. If you CALL a subroutine it will not complain if there is no RET present and similarly if you RET without being called it won't complain. Page 56

Q4:   Who is responsible for clear the stack in C and Pascal calling conventions? 2 Marks

ANS: In C the caller removes the parameter while in Pascal the callee removes them.   Page 179

Q5: What are the important "command code" in request header of device driver? 3Marks

ANS: Important command codes include 0 to initialize, 1 to check media, 2 to build a BIOS parameter block, 4 and 8 for read and write respectively. Page 158

Q6:   Write command to accomplish following tasks in AFD:

   a)   Command to view stack from an offset address 0x20 in "Memory Window 1"?

M1 SS:0020

b)   Command to change DX register content to 0x45?   3 Marks

 mov dx, 0x45

Q7:   Which attribute of INT10 being used for getting information about Super VGA?   3 Marks

ANS: page 172

INT 10 – VESA – Get SuperVGA Infromation

AX = 4F00h

ES:DI -> buffer for SuperVGA information

Return:

AL = 4Fh if function supported

AH = status

Q8:   Which pins are used in Serial Port? Which type of communication is in it? 5 Marks

ANS: page 163

1. Carrier Detect
2. Received Data

3. Transmitted Data

4. Data Terminal Ready

5. Signal Ground

6. Data Set Ready

7. Request to send

8. Clear to Send

9. Ring Indicator

Serial port is a way of communication among two devices just like the parallel port. The basic difference is that whole bytes are sent from one place to another in case of parallel port while the bits are sent one by one on the serial port in a specially formatted fashion.

Q9:   What is the function of "segment override prefix" and what changes it brings to opcode?   5 Marks

ANS: To override the association for one instruction of one of the registers BX, BP, SI or DI, we use the segment override prefix. For example "mov ax, [cs:bx]" associates BX with CS for this one instruction. For the next instruction the default association will come back to act. The processor places a special byte before the instruction called a prefix. Page 26.

Q10:   Draw a half straight line in graphics mode middle of the screen from extreme left to right?   5 Marks

ANS: ; draw line in graphics mode

[org 0x0100]

mov ax, 0x000D ; set 320x200 graphics mode

int 0x10 ; bios video services

mov ax, 0x0C07 ; put pixel in white color

xor bx, bx ; page number 0

mov cx, 200 ; x position 200

mov dx, 100 ; y position 200

l1: int 0x10 ; bios video services

dec dx ; decrease y position

loop l1 ; decrease x position and repeat

mov ah, 0 ; service 0 – get keystroke

int 0x16 ; bios keyboard services

mov ax, 0x0003 ; 80x25 text mode

int 0x10 ; bios video services

mov ax, 0x4c00 ; terminate program

int 0x21

Q11:   Using the function "int multiply (int x, int y)" from C and convert it into assembly code?   5 Marks

ANS: page 179

push dword [myy]

push dword [myx]

call _ multiply

add esp, 8

; EAX holds the answer

Q.   Give an example instruction to show based register indirect addressing mode. 2mrks

Ans.   A base register is used in brackets and the actual address accessed depends on the value contained in that register. For example "mov [bx], ax" moves the two byte contents of the AX register to the address contained in the BX register in the current data segment.

Q.   How many broad categories of video services are classified? And also write the name of these video services.

Ans. The video services are exported via INT 10. We will discuss some very simple services. Video services are classified into two broad categories;

(a)   Graphics mode services and
(b)   Text mode services.

In graphics mode a location in video memory corresponds to a dot on the screen. In text mode this relation is not straightforward. The video memory holds the ASCII of the character to be shown and the actual shape is read from a font definition stored elsewhere in memory.

Q. What is the purpose of bios parameter block?

Ans.   The BIOS parameter block is a structure that provides parameters about the storage medium. It is stored in the first sector or the boot sector of the device.

Q.   What is the purpose of Abort?

An 'abort' is caused by an attempt to access a memory or I/O location which is out of bounds to the program that is currently executing. An abort is signalled by some device external to the ARM asserting a signal on the CPU's ABORT line.

Q.   How to reset disk file system using INT 13 disk rest service? 3mrks

INT 13 - DISK - RESET DISK SYSTEM
        AH = 00h
        DL = drive (if bit 7 is set both hard disks and floppy disks reset)
Return: AH = status (see #00234)
        CF clear if successful (returned AH=00h)
        CF set on error

Q. Which two instructions are used to check whether particular bits are set or not? How these two instructions differ from each other?

Ans. AND can be used to check whether particular bits of a number are set or not. Previously we used shifting and JC to test bits one by one. Now we introduce another way to test bits, which is more powerful in the sense that any bit can be tested anytime and not necessarily in order. AND can be applied on a destination with a 1-bit in the desired position and a source, which is to be checked. If the destination is zero as a result, which can be checked with a JZ instruction, the bit at the desired position in the source was clear.
However the AND operation destroys the destination mask, which might be needed later as well. Therefore Intel provided us with another instruction analogous to CMP, which is non-destructive subtraction. This is the TEST instruction and is a non-destructive AND operation. It doesn't change the destination and only sets the flags according to the AND operation. By checking the flags, we can see if the desired bit was set or cleared.

Q. What is multithreading?

Ans.   Multi-threading is a widespread programming and execution model that allows multiple threads to exist within the context of a single process. These threads share the process' resources, but are able to execute independently. The threaded programming model provides developers with a useful abstraction of concurrent execution.

Q. In assembly language programming which attributes of INT 10 service are being used for getting information about super VGA.

Ans.

We use service 4F of INT 10 for getting information about super VGA, details of INT-104F are given below:-

INT 10 - VESA SuperVGA BIOS (VBE) - GET SuperVGA INFORMATION
        AX = 4F00h
        ES:DI -> buffer for SuperVGA information (see #00077)
        Return: AL = 4Fh if function supported
        AH = status
         00h successful
        ES:DI buffer filled
        01h failed

Q. In INT 10 of video services while writing graphics pixel, what do the values of registers BH, AL, CX, and DX corresponds to? 5mrks

Ans.

        INT 10 - VIDEO - WRITE GRAPHICS PIXEL
        AH = 0Ch
        BH = page number
        AL = pixel color
        CX = column
        DX = row

Q. What is divide overflow error? Complete the following code such that it will cause divide overflow error. mov BL,_____ mov AX,_____ DIV BL

Ans.

Mov BL, 2   ;   use a very small divisor
Mov AX, 0x8432   ;   load a big number in ax
DIV BL   ;   Interrupt 0 will be generated i.e divide overflow

Q. What are the content at the starting position of the file designated as device driver?

Ans. Device driver file starts with a header containing a link to the next driver in the first four bytes followed by a device attribute word. The most important bit in the device attribute word is bit 15 which dictates if it is a character device or a block device. If the bit is zero the device is a character device and otherwise a block device. Next word in the header is the offset of a strategy routine, and then is the offset of the interrupt routine and then in one byte, the number of units supported is stored. This information is padded with seven zeroes.

Q.   In which technique user feels the programs are running at the same time and how does this technique works?

Ans.   Multitasking is the technique in which the user feels the programs are running at the same time. In multitasking scheduler is responsible to schedule the threads. Time division technique is used to give each thread limited time to execute, then second thread and so on. Due to fast processing speed of the processor user feel that multiple programs are running at same time.

Q.   In interrupt gate descriptor what does bit s and bit p represent and what is the purpose of these bits?

Ans. The P bit must be set to signal that this segment is present in memory. The P and DPL have the same meaning as in data and code descriptors. The S bit tells that this is a system descriptor while the 1110 following it tells that it is a 386 interrupt gate

Q.   What is difference between these instructions?

> **MOV AX, 10**
> **MOV AX, 0X10**
> **MOV AX, [0X10]**
>
> **Ans.**
>
> **MOV AX, 10   ; Loads decimal 10 to AX register**
> **MOV AX, 0X10   ; Loads Hexadecimal number 0x10 into AX register**
> **MOV AX, [0X10]   ; Loads the contents of the memory pointed by 0x10 in current code segment.**

Q. Which processor's instruction set allows the following instructions and describe what task will be performed by each of these instructions:

- **CLR(A3)**

- **EXG D0, D2**

- **Move.B (A1),(A2)**

- **MoveA (2222).L,A4**

Ans.   Motorola 68K processor allows following instruction set:-

- **CLR(A3)**   **; Clears or set value 0 in A3 register**
- **EXG D0, D2**   **; Moves value in from D0 to D2 register**

- **Move.B (A1),(A2)**   **; Moves single Byte from A1 to A2 register**

- **MoveA (2222).L,A4**   **; Moves address 0x2222 to A4 register**

Q.11: Consider the following pseudo-code and write the assembly code for it. Note: There is more credit for a shorter code.

If(a1>b1) AND (b1>c1)

{

cx=1

}

Ans.

[org 0x0100]

        cmp a1, b1
        jb exit
        cmp b1, c1
        jb exit
        mov cx, 1

exit: mov ax, 0x4c00 ; terminate program
        int 0x21

Q.12: Explain the purpose of assembly language instructions from 1 to 4 lines as given below:

Mov si,12

Mov al,[cs:si]

Mov [opcode],al

Mov byte [cs:si], 0xcc

Ans.

Reference Pg. 138 ex 10.2

```
mov si, l2              ; load breakpoint position in si
mov al, [cs:si]         ; read opcode at that position
mov [opcode], al        ; save opcode for later use

mov byte [cs:si], 0xCC  ; change opcode to INT3
```

Q.1: What are different registers settings for file closing using INT21 service?

Ref: Pg 161

```
INT 21 - CLOSE FILE
AH = 3Eh
BX = file handle
Return:
CF = error flag
AX = error code
```

Q.2: Which bit of port 37A enables the IRQ7 triggering by the ACK pin?

Ref. Pg. 125 Ex 9.9

Ans. Port 37A is the control port. Our interest is with bit 4 of this port which enables the IRQ 7 triggering by the ACK pin. We have attached a circuit that connects 8 LEDs with the parallel port pins.

Q.3: How many types of user descriptors are used in 8088 architecture? Write their names only.

Ans. In 8088 architecture, user descriptors are not used. The concept of GDT, LDT and IDT was first introduced in Intel 80386 microprocessor.

Q.4: What is the purpose of direction flag (DF) and what will happen if DF= 1?

Ans. The direction of movement is controlled with the Direction Flag (DF) in the flags register. If this flag is cleared the direction is from lower addresses towards higher addresses and if this flag is set the direction is from higher addresses to lower addresses. If DF is cleared, this is called the auto increment mode of string instruction, and if DF is set, this is called the auto decrement mode. There are two instructions to set and clear the direction flag.

cld ; clear direction flag

std ; set direction flag

Q.5: In case of a typical central processing unit, how program counter governs the complete execution of an instruction?

Ans.   Instructions run one after another, first, second, third and so on. Instructions have a positional relationship. The whole logic depends on this positioning. If the computer executes the fifth instructions after the first and not the second, all our logic is gone. The processor should ensure this ordering of instructions. A special register exists in every processor called the program counter or the instruction pointer that ensures this ordering. "The program counter holds the address of the next instruction to be executed." A number is placed in the memory cell pointed to by this register and that number tells the processor which instruction to execute; for example 0xEA, 255, or 152. For the processor 152 might be the add instruction. Just this one number tells it that it has to add, where its operands are, and where to store the result. This number is called the *opcode*. The instruction pointer moves from one opcode to the next. This is how our program executes and progresses. One instruction is picked, its operands are read and the instruction is executed, then the next instruction is picked from the new address in instruction pointer and so on.

Q.6: Following are three common video services for INT 10 used in text mode. Identify the use of each video service which is listed below:

- **AH=00h**
- **AH=01h**

- **AH=02h**

Ans.

INT 10 - VIDEO - SET VIDEO MODE
AH = 00h
AL = desired video mode

Some common video modes include 40x25 text mode (mode 0), 80x25 text mode (mode 2), 80x50 text mode (mode 3), and 320x200 graphics mode (mode D).

INT 10 - VIDEO - SET TEXT-MODE CURSOR SHAPE
AH = 01h
CH = cursor start and options
CL = bottom scan line containing cursor (bits 0-4)

INT 10 - VIDEO - SET CURSOR POSITION
AH = 02h
BH = page number
0-3 in modes 2&3
0-7 in modes 0&1
0 in graphics modes
DH = row (00h is top)

DL = column (00h is left)

Q6. Calculate first & last physical values by followings.
a. 0001
b. E000

Ans.

   (a) 0001
   (b) First Physical Address:-
       00010 + 00000 = 00010

   Last Physical Address:-
       00010 + 0FFFF = 10019

   (c) E000

First Physical Address:-
   0E000 + 00000 = 0E000

Last Physical Address:-
   0E000 + 0FFFF = 1DFFF

Q7. What value is needed to be put in the table index in order to access the
GDT
LDT

Ans.
The table index (TI) is set to 0 to access the global table of descriptors called the GDT (Global Descriptor Table). It is set to 1 to access another table, the local descriptor table (LDT) that we will not be using

Q8. What will happen when the execution of this code
Mov dx, msg
Move ah, 9
Int 0x21

Ans.
The above code writes the $ terminated string pointed by 'msg' on the screen, the details of the interrupt are given below:-

INT 21 - DOS 1+ - WRITE STRING TO STANDARD OUTPUT
AH = 09h
DS:DX -> '$'-terminated string
Return: AL = 24h (the '$' terminating the string, despite official docs which state that nothing is returned) (at least DOS 2.1-7.0 and NWDOS)

Q10. Calculate Physical address

1DDD;0436
1234;7920
74F0;2023
0000:6727
FFFF:4336

Ans.

   (a) 1DDD: 0436
        1DDD0 + 00436

= 1E206

(b) 1234: 7920

12340 + 07920
= 19C60

(c) 74F0;2023
= 74F00 + 02023
= 76F23

(d) 0000: 6727
= 00000 + 0627
= 00627

(e) FFFF : 4336
= FFFF0 + 04336
= 04326

The thread registration code initializes the PCB and adds it to the linked list so that the _____ will give it a turn. ► assembler ► scheduler (Page 141) ► linker ► debugger

Question No: 5 (Marks: 1) - Please choose one VESA VBE 2.0 is a standard for ► High resolution Mode (Page 180)

► Low resolution Mode

► Medium resolution Mode

► Very High resolution Mode

Question No: 6 (Marks: 1) - Please choose one Which of the following gives the more logical view of the storage medium

► BIOS ► DOS (Page 155) ► Both ► None

Question No: 9 (Marks: 1) - Please choose one Which of the following pins of a parallel port connector are grounded?

► 10-18 ► 18-25 (Page 125) ► 25-32 ► 32-39

Question No: 10 (Marks: 1) - Please choose one The physical address of IDT(Interrupt Descriptor Table) is stored in _____

► GDTR ► IDTR (Page 182) ► IVT ► IDTT

In NASM an imported symbol is declared with the .... while and exported symbol is declared with the...........................

► **Global directive, External directive**

► **External directive, Global directive (Page 189)**

► **Home Directive, Foreign Directive**

► **Foreign Directive, Home Directive**

For the execution of the instruction "DIV BL", the implied dividend will be stored in ► **AX (Page 85)** ► **BX** ► **CX** ► **DX**

Question No: 18 (Marks: 1) - Please choose one INT 21 service 01H is used to read character from standard input with echo. It returns the result in _____ register.

► **AL (Page 152)** ► **BL** ► **CL** ► **BH**

Question No: 19 (Marks: 1) - Please choose one BIOS sees the disks as

► **logical storage**

► **raw storage (Page 155)**

► **in the form of sectors only**

► **in the form of tracks only**

Question No: 23 (Marks: 1) - Please choose one Motorola 68K processors have....................... 23bit general purpose registers.

► **4** ► **8** ► **16 (Page 191)** ► **32**

Value of AH in the write Graphics pixel service is ► **0Ch (Page** ► **0Bh** ► **1Ch** ► **2Ch**

IRQ is referred to

► **Eight input signals**

► **One output signal**

► **One input signals**

► **Eight output signals**