

NAME :- Abdullah Tahiv

ID :- 13789

Dept :- BSCS

Subject :- Microprocessor &  
Assembly Language

(ii) Discuss virtual machine concept using example.

Ans) An effective way to explain how a computer hardware and software are related is called the virtual machine concept. Hardware such as CPU or monitor which is related through window as this is made using virtual machine with the help of ~~the~~ drives and also kernel files.

Example

Following are the examples

- VM ware workstation
- Xen
- KVM
- Hyper V

c) Features of P965 express chipset.

The chipset P965 express is mostly used in desktop computers the old ones having core 2 duo technique.

The main feature was the fast memory access uses an updated memory controller Hub. They could work dual channel. It also support multiple serials.

6) Explain different registers used in x86 32-bit processor.

Ans) Four 32-bit data registers are used for the operations. These 32-bit registers can be used in 3 ways.

• EAX, EBX, ECX, EDX

• EAX is the primary accumulator, it is used in I/O and most arithmetic instructions.

• EBX is the base register, as it is used in indexed addressing.

• ECX is the count register as the ECX, CX register store loop count.

• EDI is the data register and is also used in I/O operations and also used with AX register.

Q10:- Elaborate ----- character?

Ans) When using the I/O functions high level I/O is slower than the low level I/O, Low level

I/O provides direct access to files and devices

Performing quick and efficiently I/O operation

The low level I/O system in C provides functions that can be used to access files

Opt. Open()

Close

Read

Write

(22)a) Real address mode and protected mode.

Ans) The main difference is the mode of CPU is in protected mode the OS can use features like paging and virtual memory. Also real mode code is never in 32bit where as protected mode code can be 16 bit or 32bit every x86 CPU starts in real mode and OS must switch to protected mode.

b) Differentiate between instruction & directive.

\* A directive is mainly an order, usually issued by an authority. A directive may establish policy, assign responsibilities, define objectives and delegate authority to those working in and with the authoritative figure. Instruction on the other hand act as guideline, they often appear as a series of steps or stages one must complete one after the other.

Q) Difference between Data label & Code label.

Ans) Data label is the label that we use to define data as we defined memory locations num1, num2. Code label is the label that we have on code as seen in condition jumps and used for loop control statement.

## e) Difference between Status flag & Control flag

- Status flag or condition flag is a collection of status flag bits for processor
- Carry flag, parity flag, Auxiliary flag, zero flag, sign flag, overflow flag.

The control flag indicate how a successful attempt or failed attempt through each module is handled

- Trap flag, interrupt flag, direction flag

## c) Difference between equ directive and Equal-sign directive.

The EQU directive gives a symbolic name to a numeric constant, a register relative value or a PC relative value

But the equal sign directive associates a symbol name with an integer expression  
the syntax is name = expression.

Q3) a)  $S = 11101100$ ,  $X = 00010011$ ,  $Y = 00111100$   
 find  $Z = WVX \wedge \neg Y$

w	x	y	$\neg y$	wvx	wvx $\wedge$ $\neg y$
1	0	0	1	1	1
1	0	0	1	1	1
1	0	1	0	1	0
0	1	1	0	1	0
1	0	1	0	1	0
1	0	1	0	1	0
0	1	0	1	1	1
0	1	0	1	1	1

b) creat  $\neg A \wedge \neg B$

\*  $(A)_{16} \rightarrow (10)_{10} \rightarrow (1010)_2$   
 $(B)_{16} \rightarrow (11)_{10} \rightarrow (1011)_2$

$\neg A$	$\neg B$	$(\neg A \wedge \neg B)$
0101	0100	T
0100	0100	F
0100	0100	F
0100	0100	F

3)c) Decimal format =

$$-1 = (-1)_{10}$$

ii) Binary format

$$1 = 0001$$

taking 2's complement

$$= 1110$$

$$\begin{array}{r} + 1 \\ \hline 1111 \end{array}$$

$$-1 = (1111)_2 \text{ or } -1_{16} (1111)_{16}$$

iii) Hexadecimal format

$$1 = 0001$$

taking 2's complement

$$-1 = 1110$$

$$\begin{array}{r} + 1 \\ \hline 1111 \end{array}$$

$$1111 = (0F)_{16} \text{ OR } (0FH)_{16}$$

iv) Octal format

$$= -1 (-1)_8$$

Q4)a) Show - - - endian order

\* Order of double word in memory is as following

0000	21
0001	43
0002	65
0004	87

val

DWORD 12345678h =>

Q4)b) Write - - - size-string.

\* Assemble statement to calculate the number of bytes in given string is as following

my string Byte = I am student of INU  
my string - len = ( \$ - my string )

size-string = ( \$ - my string )



4)c) write program - - - register

- \* model gloat, stdcall
- \* stack 4096
- \* Exit process proto, dw Exit code: dword  
.code

Main PROC

; assign the integer value to register

mov Eax, 150 ; A = EAX = 150

mov Ebx, 100 ; B = EBX = 100

mov Ecx, 50 ; C = ECX = 50

mov edx, 40 ; D = EDX = 40

; Calculate the expression

add ecx, ebx ; ECX = (C+B)

sub ecx, eax ; ECX = (C+B) - A

mov edx, ecx ; D = (C+B) - A

invoke exit process, 0

Main ENDP

END main.

#### 4)d) Assign - - - statement

- A simple assembly program that perform the following arithmetic operations.

Program  $A = (A + B) - (C / D)$

- model gloat stdcall
- stack 4096

Exit process proto, dwExit, code, dword  
.code

main Proc

; assign the integer values to register  
mov eax, 150 ; A = EAX = 150  
mov ebx, 100 ; B = EBX = 100  
mov ecx, 50 ; C = ECX = 50  
mov edx, 40 ; D = EDX = 40  
; calculate the expression

add eax, ebx ; EAX = (A+B)  
div ecx, edx ; ECX = (C/D)  
sub eax, ecx ; EAX = (A+B) - (C/D)

invoke Exitprocess, 0

main ENDP

EDN main