Assignment no.7

Name:- Gaurav Patil

Roll No. - 21259

Batch – G2

Title - Write X86/64 ALP to detect protected mode and display the values of GDTR, LDTR, IDTR, TR and MSW Registers also identify CPU type using CPUID instruction.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* C O D E \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

section .data

title :

db 10, "=====values of GDTR,LDTR,IDTR====="

title\_len equ $-title

gdtmsg db 10, "2) GDT value : "

gmsg\_len equ $-gdtmsg

ldtmsg db 10, "3) LDT value : "

lmsg\_len equ $-ldtmsg

idtmsg db 10, "4) IDT value : "

imsg\_len equ $-idtmsg

trmsg db 10, "5) TR value : "

trmsg\_len equ $-trmsg

mswmsg db 10, "6) MSW value : "

mswmsg\_len equ $-mswmsg

colmsg db ":"

rmodemsg db 10, "1) Processor In : Real Mode"

rmsg\_len equ $-rmodemsg

pmodemsg db 10, "1) Processor In : Protected Mode"

pmsg\_len equ $-pmodemsg

end db 10, "======================================",10

end\_len equ $-end

section .bss

gdt resd 1

resw 1

ldt resw 1

idt resd 1

resw 1

tr resw 1

msw resw 1

dnum\_buff resb 04

cr0\_data resd 1

%macro display 2

mov eax,4

mov ebx,1

mov ecx,%1

mov edx,%2

int 80h

%endmacro

section .text

global \_start

\_start:

display title,title\_len

smsw eax

mov [cr0\_data],eax

ror eax,1

jc prmode

display rmodemsg,rmsg\_len

jmp nxt1

prmode:

display pmodemsg,pmsg\_len

nxt1:

sgdt [gdt]

sldt [ldt]

sidt [idt]

str [tr]

smsw [msw]

display gdtmsg,gmsg\_len

mov bx,[gdt+4]

call display\_num

display colmsg,1

mov bx,[gdt+2]

call display\_num

display colmsg,1

mov bx,[gdt]

call display\_num

display ldtmsg,lmsg\_len

mov bx,[ldt]

call display\_num

display idtmsg,imsg\_len

mov bx,[idt+4]

call display\_num

display colmsg,1

mov bx,[idt+2]

call display\_num

display colmsg,1

mov bx,[idt]

call display\_num

display trmsg,trmsg\_len

mov bx,[tr]

call display\_num

display mswmsg,mswmsg\_len

mov bx,[msw]

call display\_num

display end,end\_len

exit:

mov eax,01

mov ebx,00

int 80h

display\_num:

mov esi,dnum\_buff

mov ch,04

mov cl,04

up1:

rol bx,cl

mov dl,bl

and dl,0fh

add dl,30h

cmp dl,39h

jbe skip1

add dl,07h

skip1:

mov [esi],dl

inc esi

dec ch

jnz up1

mov eax,4

mov ebx,1

mov ecx,dnum\_buff

mov edx,4

int 80h

ret

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* O U T P U T \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

