x86 HW5

2019. 06. 07

Jeon Jae Wook
Sungkyunkwan Univ.

Contents

- Describe about 4th Homework
- 5th Homework

Describe about 4th Homework

TSS and Task Gate descriptor

```
TSS1Selector    equ     20h
gdt4:
    dw  068h    ; Segment Limit 15:0
    dw  0000h   ; Base Address 15:0
    db  00h     ; Base Address 23:16
    db  89h     ; present, ring 0, system, 32-bit TSS Type
    db  00h     ; limit 19:16, flags
    db  00h     ; Base Address 31:24
TSS2Selector    equ     28h
gdt5:
    dw  068h    ; Segment Limit 15:0
    dw  0000h   ; Base Address 15:0
    db  00h     ; Base Address 23:16
    db  89h     ; present, ring 0, system, 32-bit TSS Type
    db  00h     ; limit 19:16, flags
    db  00h     ; Base Address 31:24
TSS3Selector    equ     30h
gdt6:
    dw  068h    ; Segment Limit 15:0
    dw  0000h   ; Base Address 15:0
    db  00h     ; Base Address 23:16
    db  89h     ; present, ring 0, system, 32-bit TSS Type
    db  00h     ; limit 19:16, flags
    db  00h     ; Base Address 31:24
```

```
Task_Gate_Descriptor equ 50h
gdt10:
    dw  00h             ; Reserved
    dw  TSS3Selector    ; TSS Segment Selector
    db  00h             ; Reserved
    db  85h             ; present, ring 0, system, Task Gate Type
    dw  00h             ; Reserved
```

Describe about 4th Homework

- **Make TSS descriptor**
  - **Base address field of TSS descriptor ← Start address of TSS**

```
mov eax, tss1
mov word [gdt4+2], ax
shr eax, 16
mov byte [gdt4+4], al
mov byte [gdt4+7], ah
```

```
TSS1Selector    equ     20h
gdt4:
    dw  068h    ; Segment Limit 15:0
    dw  0000h   ; Base Address 15:0
    db  00h     ; Base Address 23:16
    db  89h     ; present, ring 0, system, 32-bit TSS Type
    db  00h     ; limit 19:16, flags
    db  00h     ; Base Address 31:24
```

Describe about 4th Homework

- **Make Task Gate descriptor**
  - **TSS Segment Selector field of Task Gate descriptor**

← TSS segment selector

```
Task_Gate_Descriptor equ 50h
gdt10:
    dw  00h             ; Reserved
    dw  TSS3Selector    ; TSS Segment Selector
    db  00h             ; Reserved
    db  85h             ; present, ring 0, system, Task Gate Type
    dw  00h             ; Reserved
```

Describe about 4th Homework

- **Task Switching**
  - **Initialize TSS field**
  - **Task switching using Task Gate and TSS segment selector**

```
;fill the value of tss1
mov word [tss1+96], LDTR1            ;LDT seg sel
mov word [tss1+76], LDT_CODE_SEL1    ; CS
mov word [tss1+84], LDT_DATA_SEL1    ; DS
mov word [tss1+80], LDT_DATA_SEL1    ; SS
mov word [tss1+72], Video_SEL        ; ES
mov dword [tss1+32], Task1           ; EIP
mov dword [tss1+56], 0xA000          ; ESP

;fill the value of tss2
mov word [tss2+96], LDTR2            ; LDT seg sel
mov word [tss2+76], LDT_CODE_SEL2    ; CS
mov word [tss2+84], LDT_DATA_SEL2    ; DS
mov word [tss2+80], LDT_DATA_SEL2    ; SS
mov word [tss2+72], Video_SEL        ; ES
mov dword [tss2+32], Task2           ; EIP
mov dword [tss2+56], 0xB000          ; ESP

;fill the value of tss3
mov word [tss3+96], LDTR3            ; LDT seg sel
mov word [tss3+76], LDT_CODE_SEL3_0  ; CS
mov word [tss3+84], LDT_DATA_SEL3    ; DS
mov word [tss3+80], LDT_DATA_SEL3    ; SS
mov word [tss3+72], Video_SEL        ; ES
mov dword [tss3+32], Task3           ; EIP
mov dword [tss3+56], 0xC000          ; ESP
```

Initialize TSS field

```
jmp TSS1Selector:0
```

Switching to Task1
using TSS segment selector

```
call TSS2Selector:0
```

Switching to Task2
using TSS segment selector

```
call Task_Gate_Descriptor:0
```

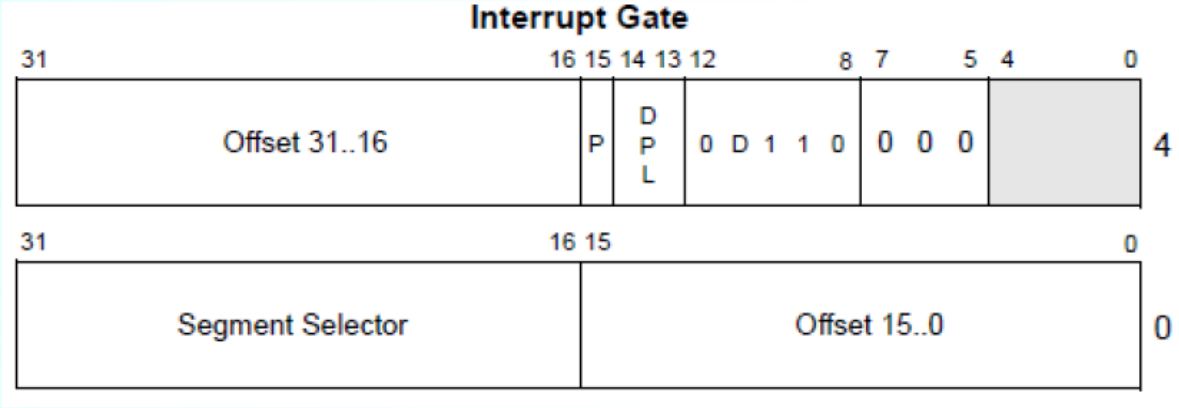
Switching to Task3
using Task Gate

5th Homework

- 5th Homework Describe
  - Make IDT
    - Task Switching
    - Exception
  - Load IDT
  - Make Interrupt Service Routine
    - ISR_00
    - ISR_13
    - ISR_80

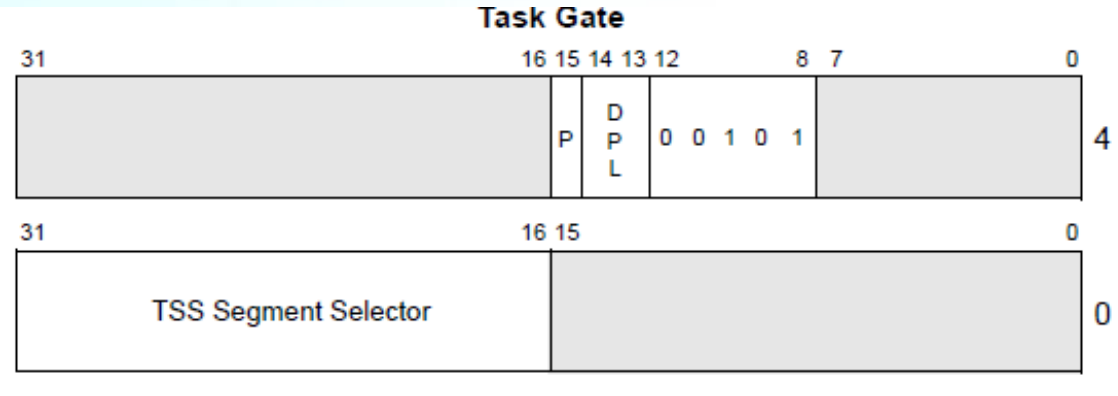
5th Homework

- **Make IDT**
  - **Interrupt descriptor whose vector number is 00**
    - **Offset : ISR address**
    - **Segment Selector : SYS_EXT_CODE segment selector**
    - **Present in memory and privileged level is 0**
    - **32 bit size**



5th Homework

- **Make IDT**
  - **Interrupt descriptor whose vector number is 30**
    - **Segment Selector : TSS2 segment selector**
    - **Present in memory and privileged level is 0**



- **Load IDTR**
  - **Use LIDT instruction to load IDTR**

5th Homework

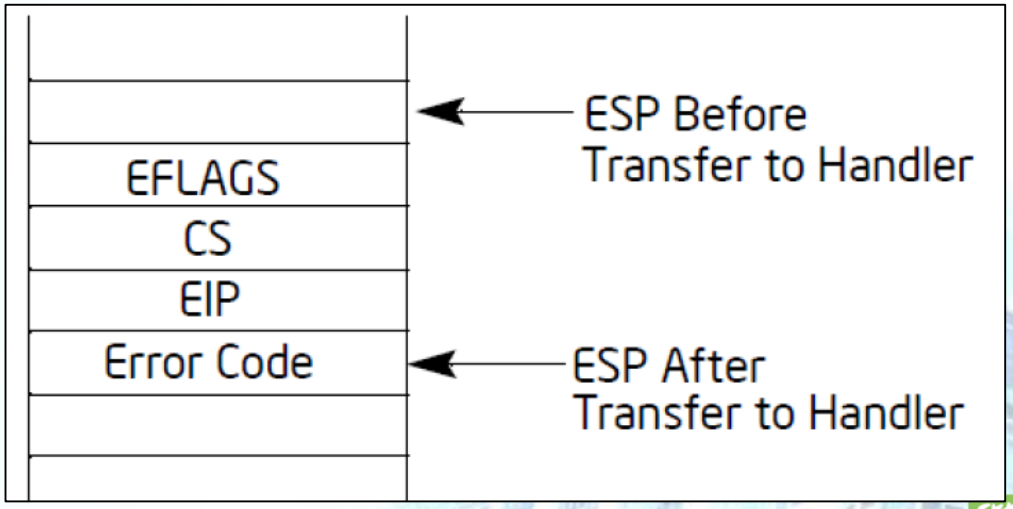
Exceptions and Interrupts

Table 6-1. Protected-Mode Exceptions and Interrupts

Vector	Mne-monic	Description	Type	Error Code	Source
0	#DE	Divide Error	Fault	No	DIV and IDIV instructions.
1	#DB	Debug Exception	Fault/ Trap	No	Instruction, data, and I/O breakpoints; single-step; and others.
2	—	NMI Interrupt	Interrupt	No	Nonmaskable external interrupt.
3	#BP	Breakpoint	Trap	No	INT 3 instruction.
4	#OF	Overflow	Trap	No	INTO instruction.
5	#BR	BOUND Range Exceeded	Fault	No	BOUND instruction.
6	#UD	Invalid Opcode (Undefined Opcode)	Fault	No	UD2 instruction or reserved opcode.[1]
7	#NM	Device Not Available (No Math Coprocessor)	Fault	No	Floating-point or WAIT/FWAIT instruction.
8	#DF	Double Fault	Abort	Yes (zero)	Any instruction that can generate an exception, an NMI, or an INTR.
9		Coprocessor Segment Overrun (reserved)	Fault	No	Floating-point instruction.[2]
10	#TS	Invalid TSS	Fault	Yes	Task switch or TSS access.
11	#NP	Segment Not Present	Fault	Yes	Loading segment registers or accessing system segments.
12	#SS	Stack-Segment Fault	Fault	Yes	Stack operations and SS register loads.
13	#GP	General Protection	Fault	Yes	Any memory reference and other protection checks.
14	#PF	Page Fault	Fault	Yes	Any memory reference.
15	—	(Intel reserved. Do not use.)		No	
16	#MF	x87 FPU Floating-Point Error (Math Fault)	Fault	No	x87 FPU floating-point or WAIT/FWAIT instruction.
17	#AC	Alignment Check	Fault	Yes (Zero)	Any data reference in memory.[3]
18	#MC	Machine Check	Abort	No	Error codes (if any) and source are model dependent.[4]
19	#XM	SIMD Floating-Point Exception	Fault	No	SSE/SSE2/SSE3 floating-point instructions[5]
20	#VE	Virtualization Exception	Fault	No	EPT violations[6]
21-31	—	Intel reserved. Do not use.			
32-255	—	User Defined (Non-reserved) Interrupts	Interrupt		External interrupt or INT *n* instruction.

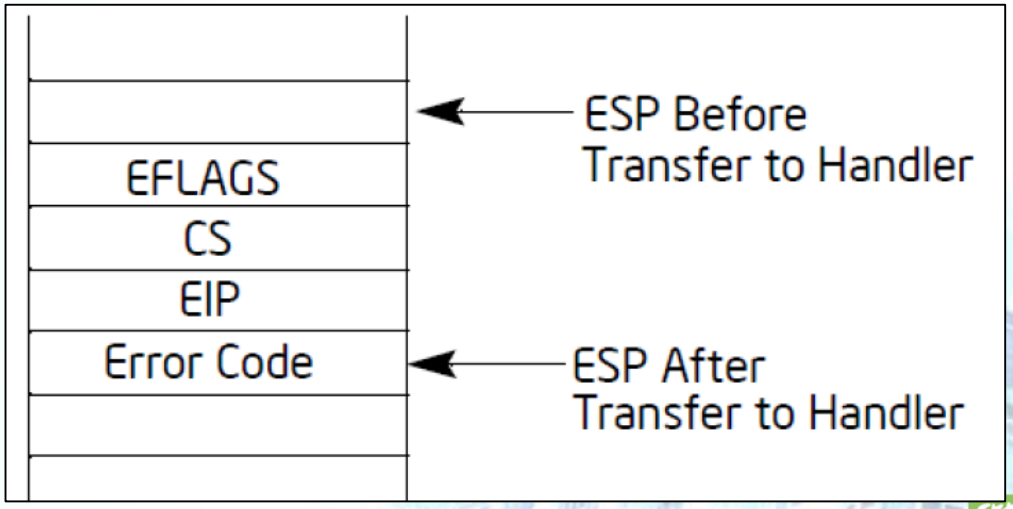
5th Homework

- **Interrupt Service Routine**
  - **ISR_00**
    - **Print string**
      - **"#DE : Divided by Zero"**
    - **Return to Task1**
      - **Restore new EIP registers**
        - **New EIP value is the address of 'return' label**
      - **Remove error code**
        - **iret does not pop error code**
      - **Use iret instruction**
        - **pop eip, cs, eflags**



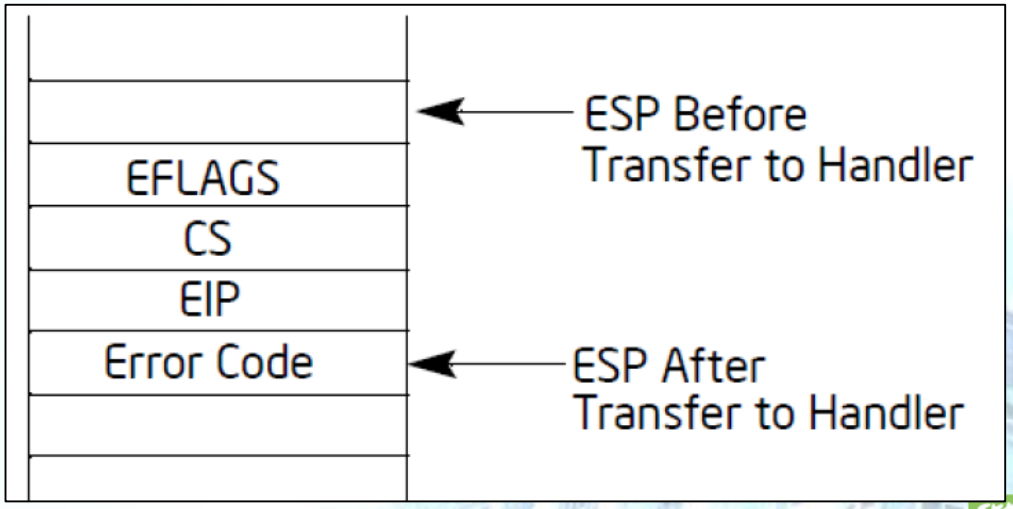
5th Homework

- **Interrupt Service Routine**
  - **ISR_13**
    - **Print string**
      - **"#GP : General Protection Fault"**
    - **Return to Task2**
      - **Restore new EIP registers**
        - **New EIP value is the address of 'return' label**
      - **Remove error code**
        - **iret does not pop error code**
      - **Use iret instruction**
        - **pop eip, cs, eflags**



5th Homework

- Interrupt Service Routine
  - ISR_80
    - Print string
      - "User Defined Interrupt"
    - Return to Task2
      - Restore new EIP registers
        - New EIP value is the address of 'return' label
      - Remove error code
        - iret does not pop error code
      - Use iret instruction
        - pop eip, cs, eflags



5th Homework

Global Descriptor Table

Index	Segment Selector	TYPE
0	-	NULL Descriptor
1	SYS_CODE_SEL	Code Segment Descriptor
2	SYS_DATA_SEL	Data Segment Descriptor
3	VIDEO_SEL	Data Segment Descriptor
4	SYS_EXT_SEL	Code Segment Descriptor
5	TASK1_CODE_SEL	Code Segment Descriptor
6	TASK2_CODE_SEL	Code Segment Descriptor
7	TSS1Selector	System Descriptor(TSS Descriptor)
8	TSS2Selector	System Descriptor(TSS Descriptor)

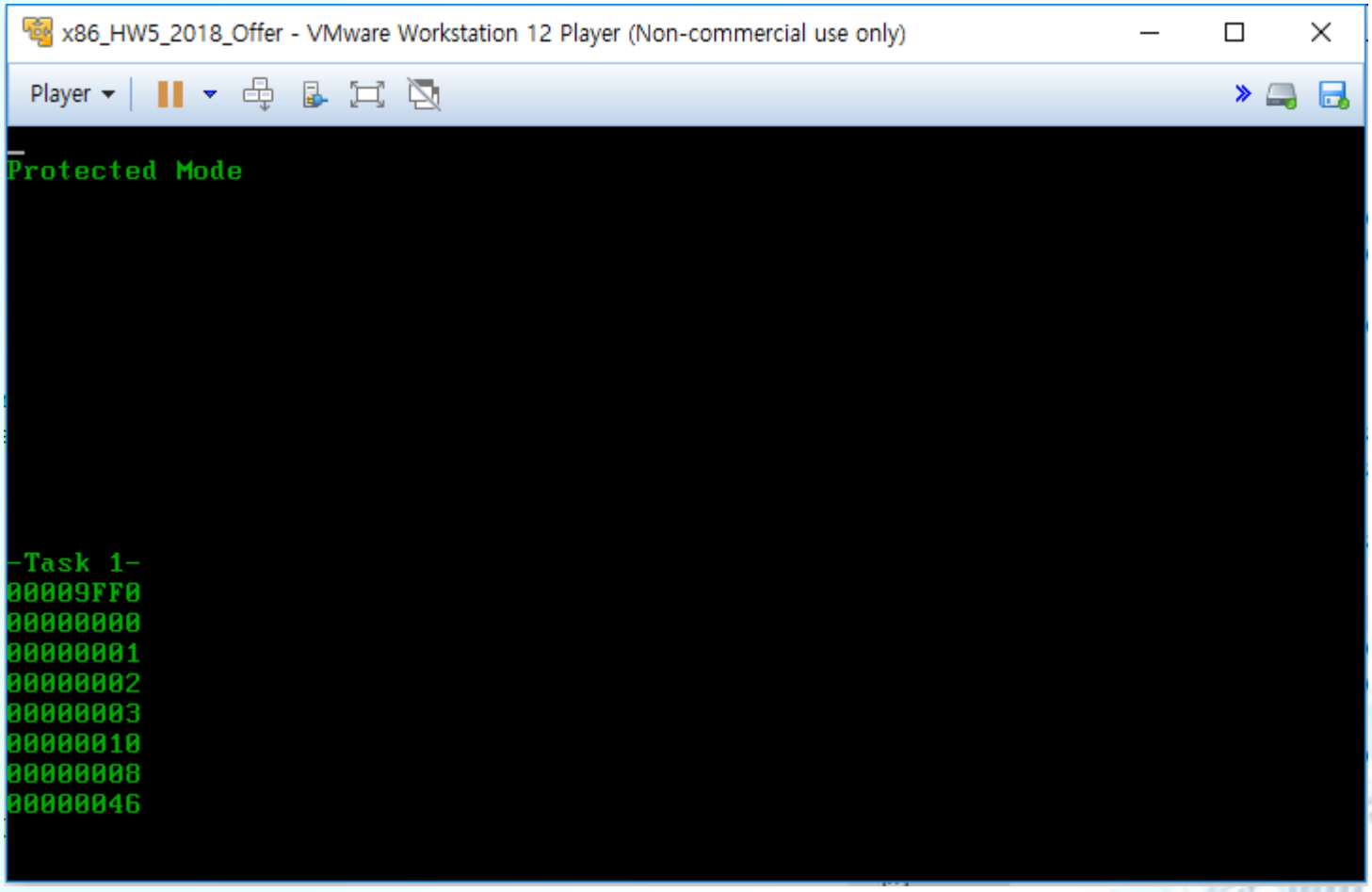
5th Homework

Interrupt Descriptor Table

Index	TYPE
0	Divide Error Exception
13	General Protection Exception
48	Task Gate Descriptor
80	User Defined Interrupt

5th Homework

Initial program



5th Homework

Result program (press keyboard)

x86_HW5_2018 - VMware Workstation 12 Player (Non-commercial use only)

```
Protected Mode
Entering Task1
Task1 switched BACK from IRQ 00h
Task2 switched from Task1
Task2 switched BACK from IRQ 0Dh
Task2 switched BACK from IRQ 50h


#DE : Divided by Zero
#GP : General Protection Fault
User Defined Interrupt


-Task 1-  -ISR_00-  -Task 2-  -ISR_13-  -ISR_80-  -Task 2-
0000AFF0  0000AFE4  0000BFF0  0000BFE0  0000BFD4  0000BFE0
00000000  0000000A  00000004  00000004  00000004  0000000F
00000001  00000000  00000005  00000005  00000005  00000005
00000002  00000002  00000006  00000006  00000006  00000004
00000003  00000003  00000007  00000007  00000007  00000000
00000010  00000010  00000010  00000010  00000010  00000010
00000028  00000020  00000030  00000020  00000020  00000030
00000046  00000046  00004046  00000046  00000046  00004046
```

ESP
EAX
EBX
ECX
EDX
DS
CS
EFLAGS

5th Homework

- How to submit
  - .asm and .bin files
  - I-Campus, until June 14th 18:59
    - format
      - 2010310000_HW4.asm
      - 2010310000_HW4.bin