

# Memory Map

lea esp, [PM_Start]		0x1000:PM_Start
call 0x38:0	아래는 특권레벨 0의 스택 영역 유저모드 SS 유저모드 ESP 80 * 2 * 7                      ← ebp + 12 msg_user_parameter1       ← ebp + 8 유저모드 CS 유저모드 EIP                      ← ebp 지정 ES eax	
lea eax, [PM_Start-256]		0x1000:PM_Start - 256
	아래는 유저모드 특권레벨 3의 스택 영역 SS                                      80 * 2 * 7 ESP                                      msg_user_parameter1 EFLAGS CS EIP  cld	0x1000:0000
	boot.asm 전체 화면에 . 출력 Kernel을 0x1000:0000에 Load PIC 초기화 Load된 kernel로 jmp	0x07C0:0000
Kernel 실행 시 SS	BIOS가 Power On Self Test 종료 이후 CMOS를 참조하여 booting device의 Master Boot Record를 읽어 0x07C0:0000에 로드한다. (Size는 512바이트이다)	0x0000:0000

PIC에 인터럽트 벡터 기준 값을 설정 해야 하는 이유

PIC는 1개당 8개의 인터럽트 입력을 처리할 수 있도록 인터럽트 관련 8개 핀을 가지고 있다.  
PC는 2개(마스터, 슬레이브)의 PIC를 사용하므로 16개의 인터럽트를 처리할 수 있으나  
슬레이브 PIC의 INT핀과 마스터 PIC의 2번(3번째) 핀이 연결되어 있어 15개의 인터럽트를  
처리할 수 있다.

그러므로 PIC는 인터럽트 번호를 0번에서 15번까지를 발생 시킨다.

그러나 X86 프로세서는 인터럽트 번호 상위 32개(0~31)를 예외(Exception)처리에 사용한다.  
실제로 사용하는 것은 20개이며 12개는 앞으로의 사용을 위해 예약한 영역이다.

그러므로 PIC에서 인터럽트를 탐지할 경우 인터럽트 번호를 그대로 CPU에 전송할 경우  
예외번호와 중복이 된다.

그래서 PIC는 ICW2번 명령을 통하여 PIC가 인터럽트를 탐지할 경우 탐지한 인터럽트 번호에  
얼마를 더하여 CPU에 전송할지를 설정한다.

# TSS 세그먼트 디스크립터

15

0

Limit의 15 ~ 0 비트							
Base Address 15 ~ 0 비트							
P	DPL	0	Type	Base Address 23 ~ 16 비트			
Base Address 31 ~ 24 비트				G	0	0	AVL
				Limit의 19 ~ 16 비트			

TSS의 물리주소가 0x10000 + 0x2345인 경우 = 0x00012345(Little endian: 0x01004523)

eax = 0x00012345 → ax = 0x2345(Little endian: 0x4523)

shr eax, 16 → eax = 0x00000100 → ah = 0x01, al = 0x00

Base Address 31 ~ 24 비트	mov [descriptor4+7], ah 0x01	[descriptor4+7]
G 0 0 AVL Limit 19~16		[descriptor4+6]
P DPL 0 Type		[descriptor4+5]
Base Address 23 ~ 16 비트	mov [descriptor4+4], al 0x00	[descriptor4+4]
Base Address 15 ~ 8 비트	0x23	[descriptor4+3]
Base Address 7 ~ 0 비트	mov [descriptor4+2], ax 0x45	[descriptor4+2]
Limit의 15 ~ 8		[descriptor4+1]
Limit의 0 ~ 7		[descriptor4]