0-1) When you boot the Linux system, the first program that runs is BIOS. Where is this program (the memory location)

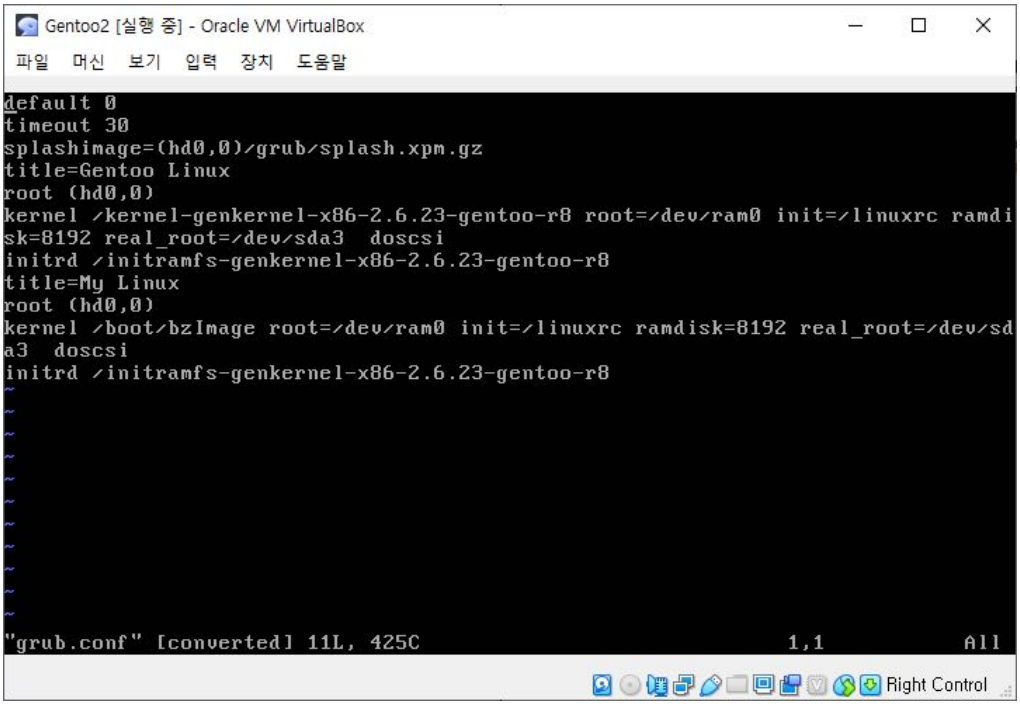
답변 : FFFFFFF0

0-2) BIOS loads and runs the boot loader program (GRUB in Linux). Where is this GRUB program?

답변 : 00007000

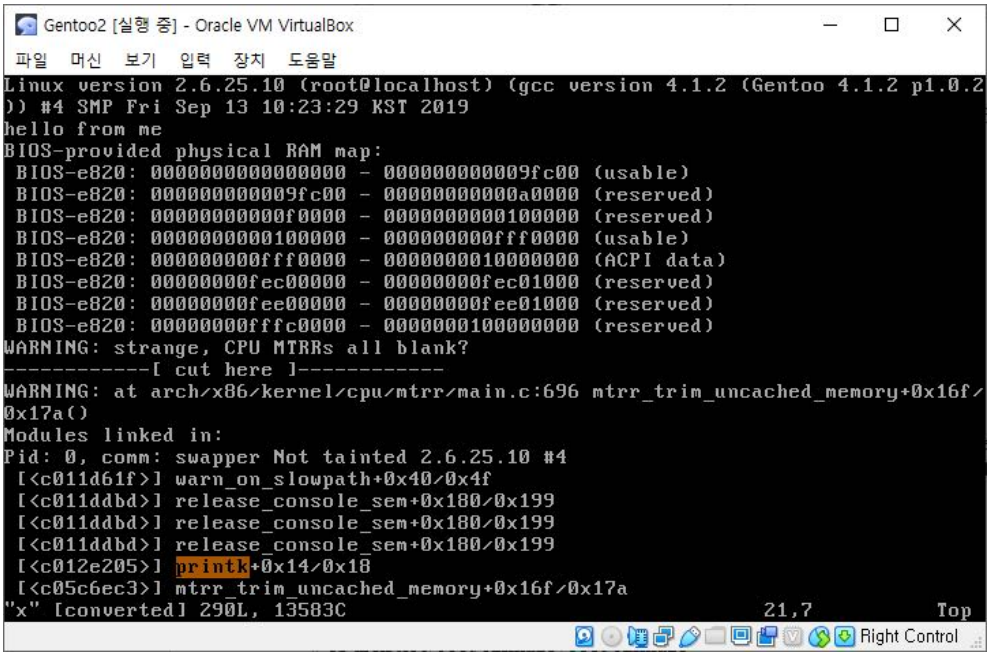
0-3) GRUB loads and runs the Linux executable file. Where is Linux executable file? How GRUB knows the location of Linux executable file?

답변 : Linux executable file의 위치는 '/boot/bzImage'이고, 'grub.conf' 내 grub이 불러드릴 항목들 중 '/boot/bzImage'이 명시되어 있다.



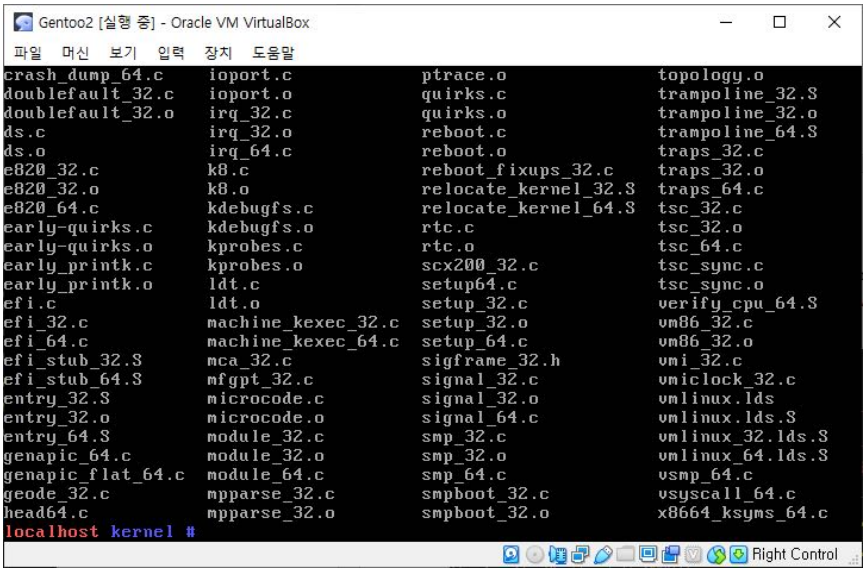
<'grub.conf'파일 내 적혀져있는 사항들>

1) Simple modification of the kernel

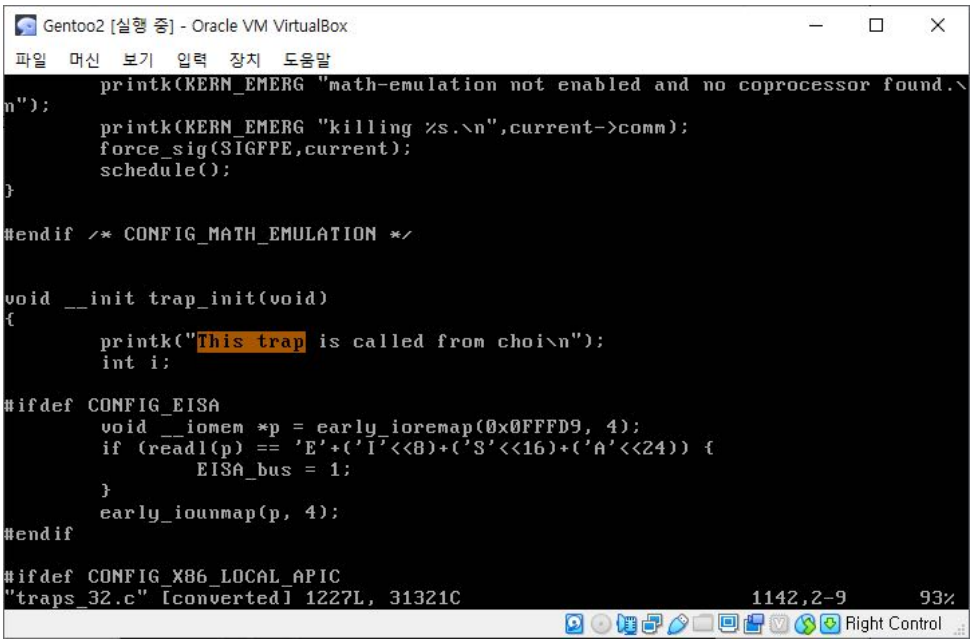


<main.c를 수정하여 부팅 때 "hello from me"출력하기>

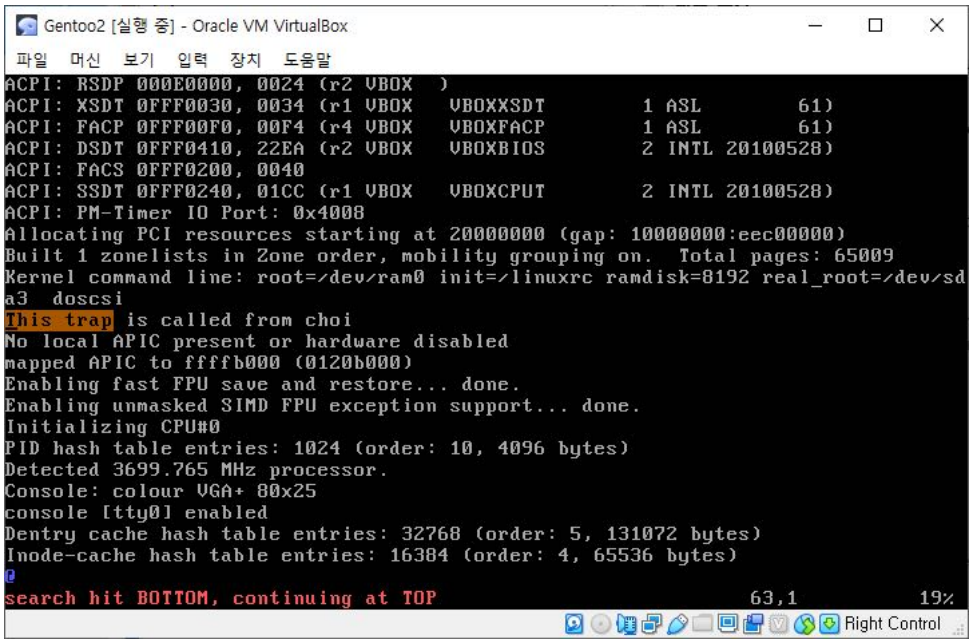
2) start_kernel() calls trap_init(), and there are many trap_init() functions defined in the kernel code. Make an intelligent guess about which trap_init() would be called and insert some printk() in the beginning of this trap_init() to see if it is really called by the kernel. Use grep in the top directory of the linux source tree to find out the locations of trap_init():



<'trap_init()'는 '/linux-2.6.25.10/arch/x86/kernel/traps_32.c'에 정의되어 있음>

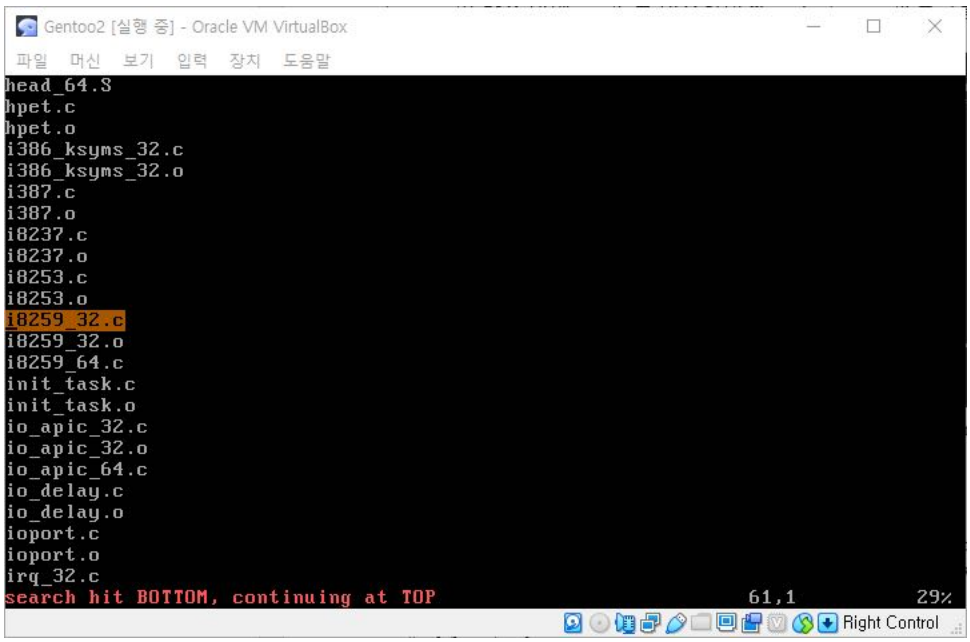


<'trap_init()'에 'printk("This trap is called from choi") 코드 삽입>

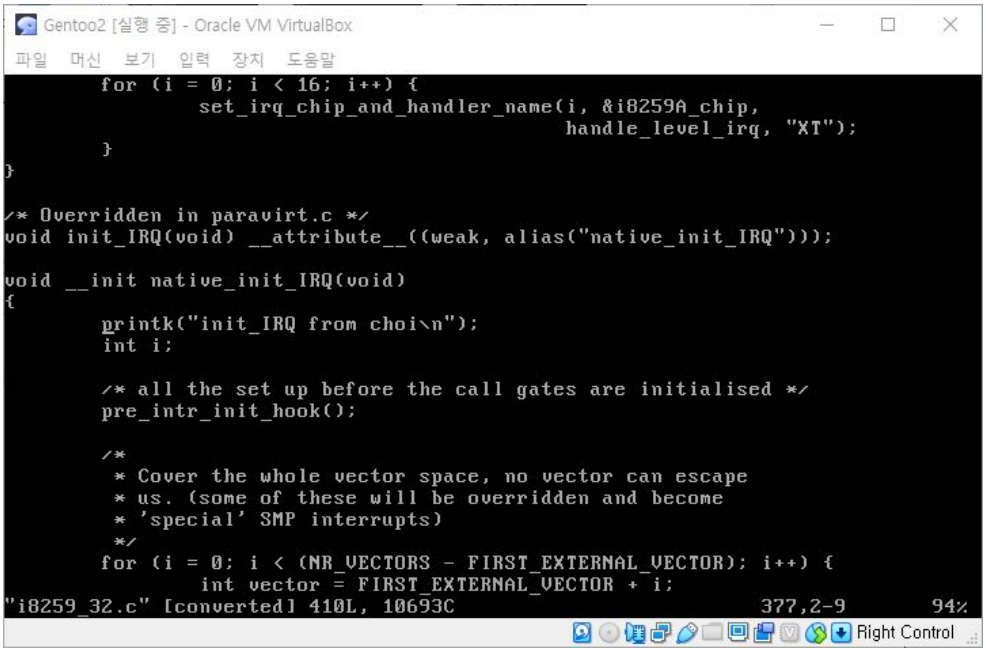


<부팅 때 위에서 추가한 printk구문이 출력된 것을 확인함>

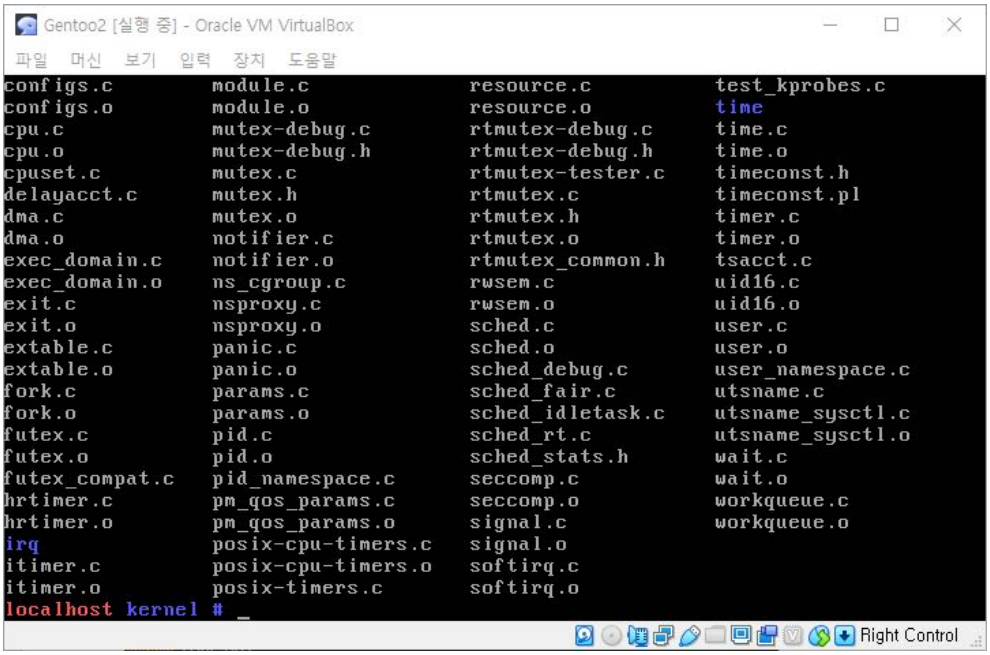
3) Find also the exact locations of init_IRQ() and insert some printk() in the beginning of init_IRQ() to confirm (actually you insert it in native_init_IRQ). Do the same thing for init_timers() and time_init()



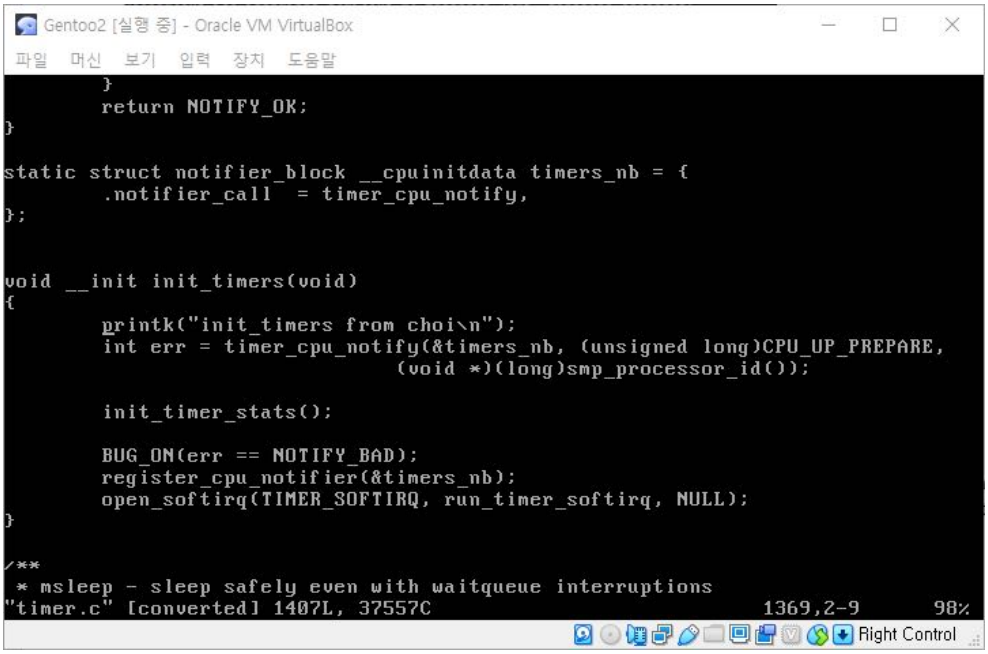
<'naitve_init_IRQ()'는 '/linux-2.6.25.10/arch/x86/kernel/i8259_32.c'에 정의되어 있음>



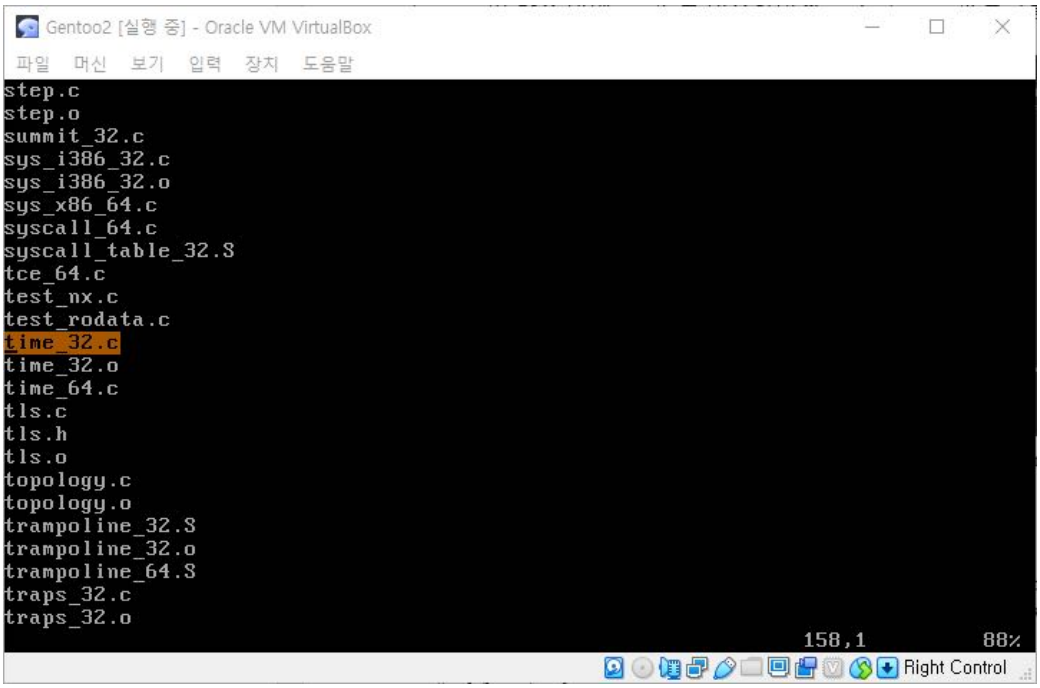
<'native_init_IRQ()'에 printk()추가>



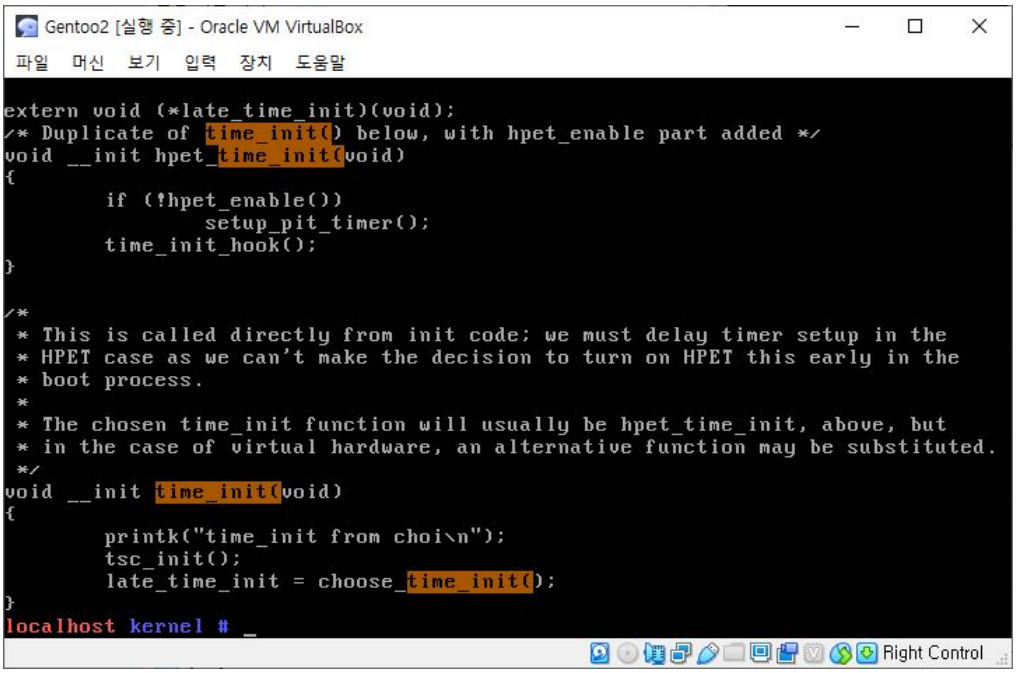
<'init_timers()'는 '/linux-2.6.25.10/kernel/timer.c'에 정의되어 있다.>



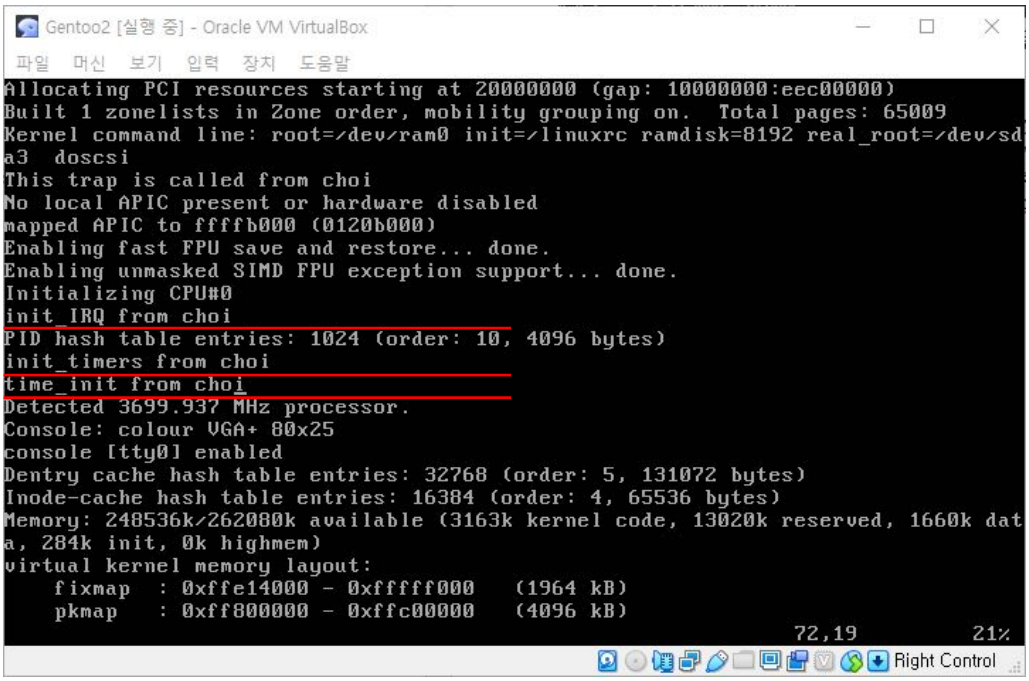
<'init_timers()'에 'printk()'추가>'



<'time_init()'은 '/linux-2.6.25.10/arch/x86/kernel/time_32.c'에 정의되어 있음>

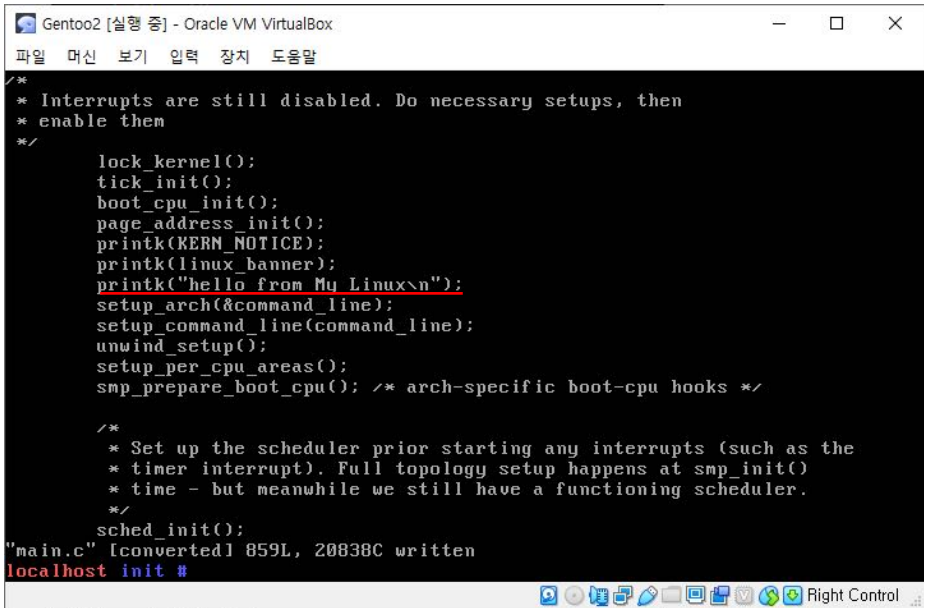


<'time_init()'에 'printk()'추가>

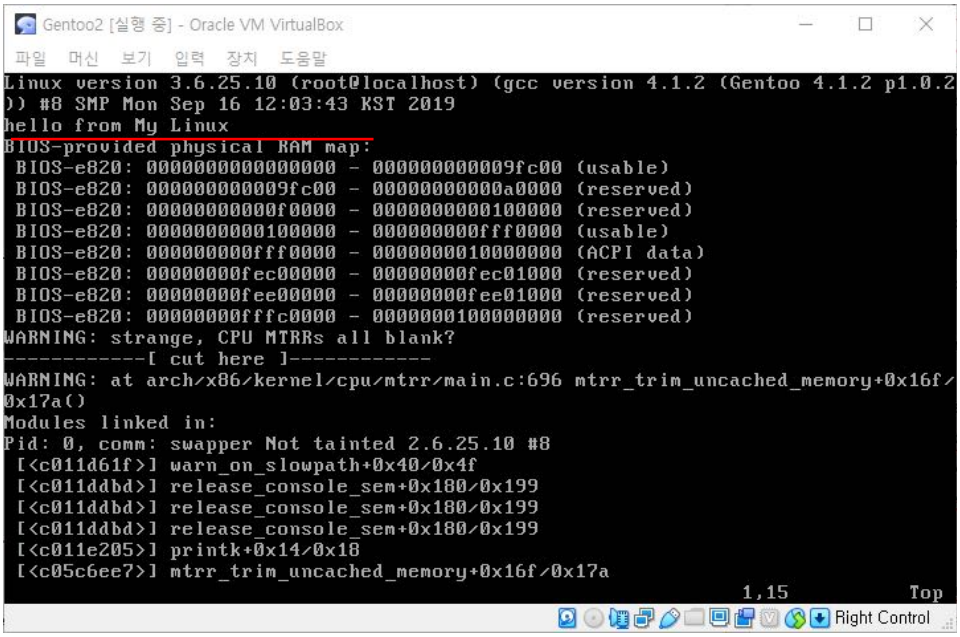


<부팅 때 추가한 printk문 확인 완료>

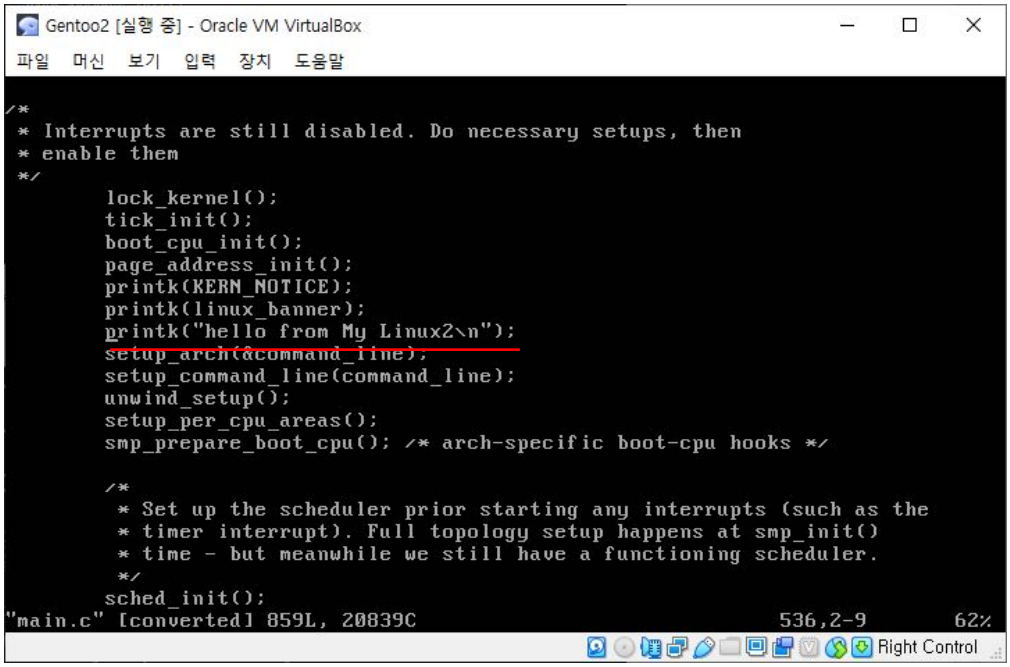
4. Modify /boot/grub/grub.conf so that GRUB displays another Linux selection, My Linux2. Set the location of the kernel for this linux as /boot/bzImage2. Prepare two versions of My Linux such that when you select "My Linux" the kernel will display "hello from My Linux", and when you select "My Linux2", it displays "hello from My Linux2".



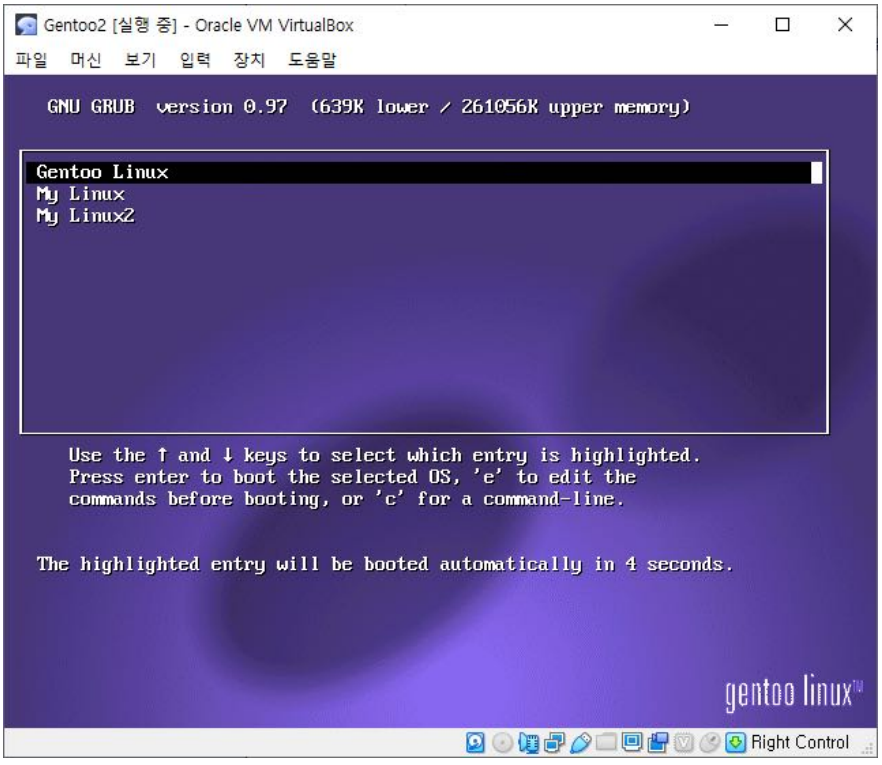
<'main.c'에 'printk("hello from My Linux")'를 추가함>



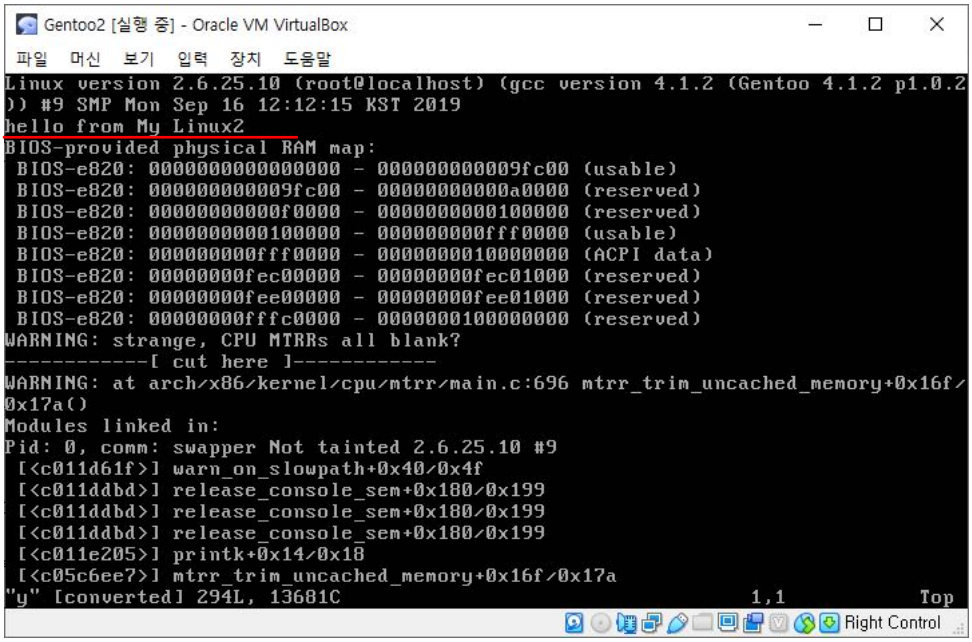
<My Linux의 부팅 메시지>



<'main.c'에 'printk("hello from My Linux2")'를 추가함>



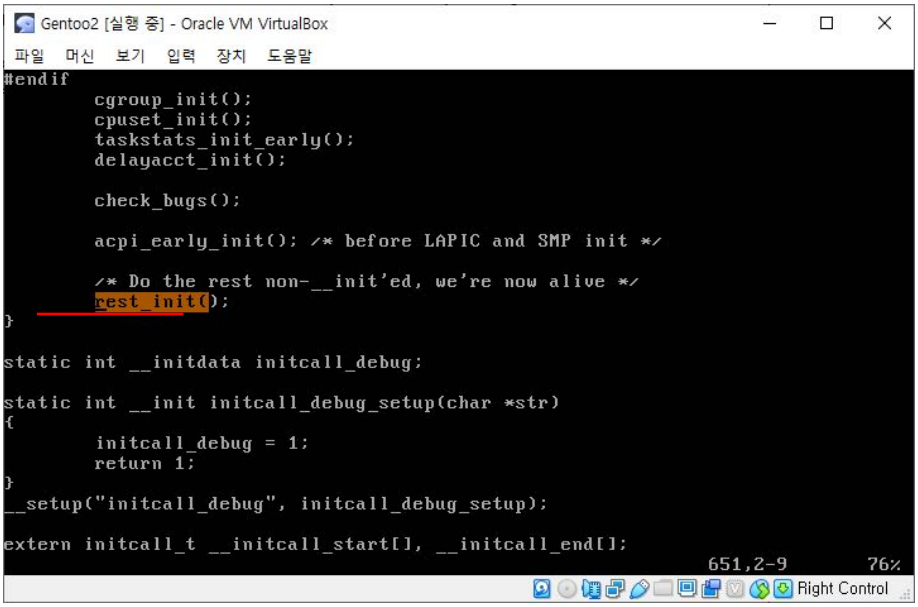
<My Linux2로 부팅,(My Linux2를 부팅하기 이전,
'cp/arch/x86/boot/bzImage /boot/bzImage2'해당 명령어를 실행함)>



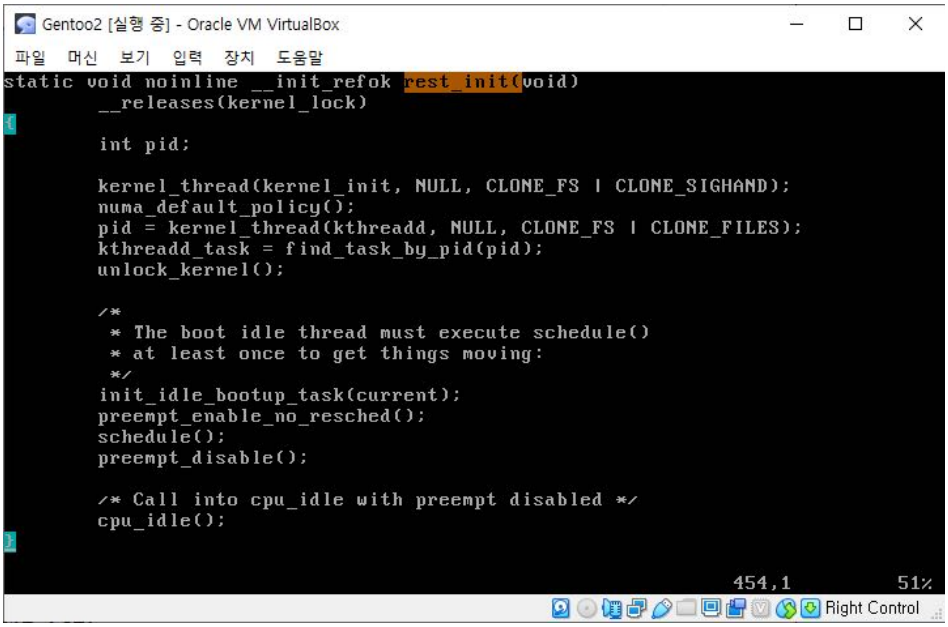
<My Linux2의 부팅 메시지>

5) Where is CPU at the end of the boot sequence when it prints "login" and waits for the user login? Explain your reasoning.

답변 : 'start_kernel()'함수의 마지막 부분에는 'rest_init()'함수를 호출한다. 해당 'rest_init()'함수선언의 마지막 부분에선 'cpu_idle()'함수를 호출한다. 즉, "login" 문자열을 호출한 부분에선 CPU가 IDLE상태가 된다.



<'start_kernel()'선언 내 'rest_init()'함수 호출>



<'rest_init()'함수 선언부분 내 'cpu_idle()'함수 호출>

밑 사진의 'cpu_idle()'함수 선언처럼, CPU가 idle모드가 되면 'while(1)'의 무한 루프가 실행될 것이다.