컴퓨터학부 20132397 윤건택

1. 과제 개요

이번 과제는 우분투 16.04 32비트 버전을 설치하고 설치된 가상환경 상에서 bochs 에뮬레이터를 통해 제공된 ssuos를 실행하여 원하는 문자를 출력하는 것이 목표이다. 이러한 목표를 수행하기 위해선 부트로더를 수정해야 하는데, 부트로더란 운영체제가 시동되기 이전에 미리 실행되면서 커널이 올바르게 시동되기 위해 필요한 모든 관련 작업을 마무리하고 최종적으로 운영체제를 시동시키기 위한 목적을 가진 프로그램을 말한다. 현재 어셈블리에 대한 지식이 없으므로 이 과제를 통해 어셈블리에 대한 기초를 쌓고 부트로더의 실행 과정을 이해하도록 한다.

1. 설계

우선 현재 사용하고 있는 환경에서 Ubuntu를 사용하려면 가상 머신을 이용해야 하므로 VirtureBox프로그램을 이용해 파티션을 생성, Ubuntu를 설치하였다.

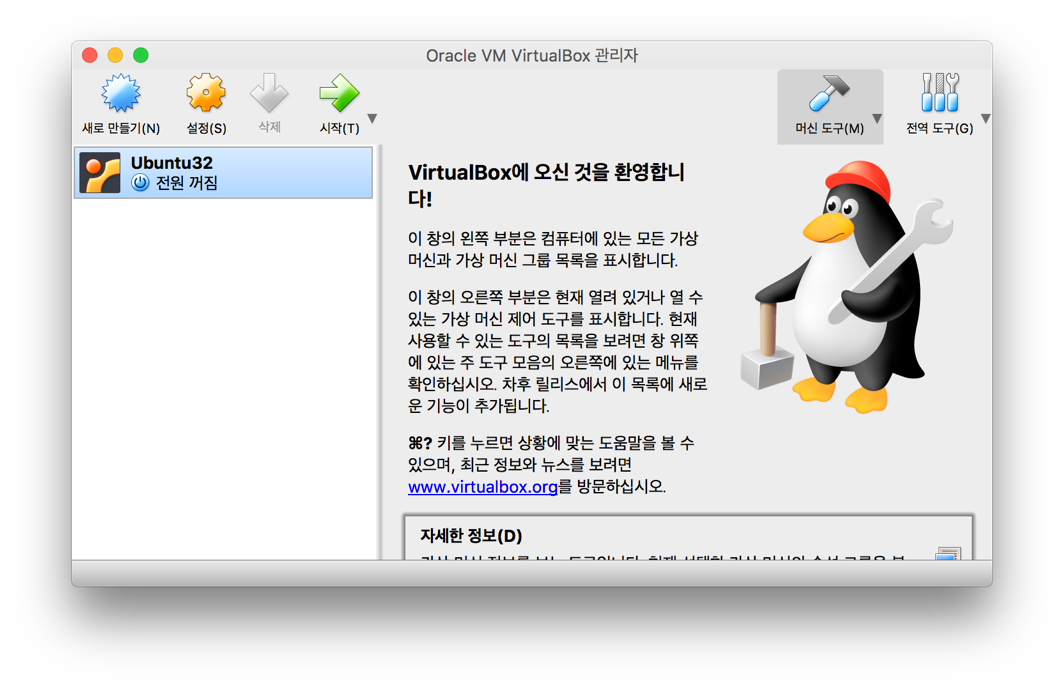


그림 1 VirtualBox 실행화면

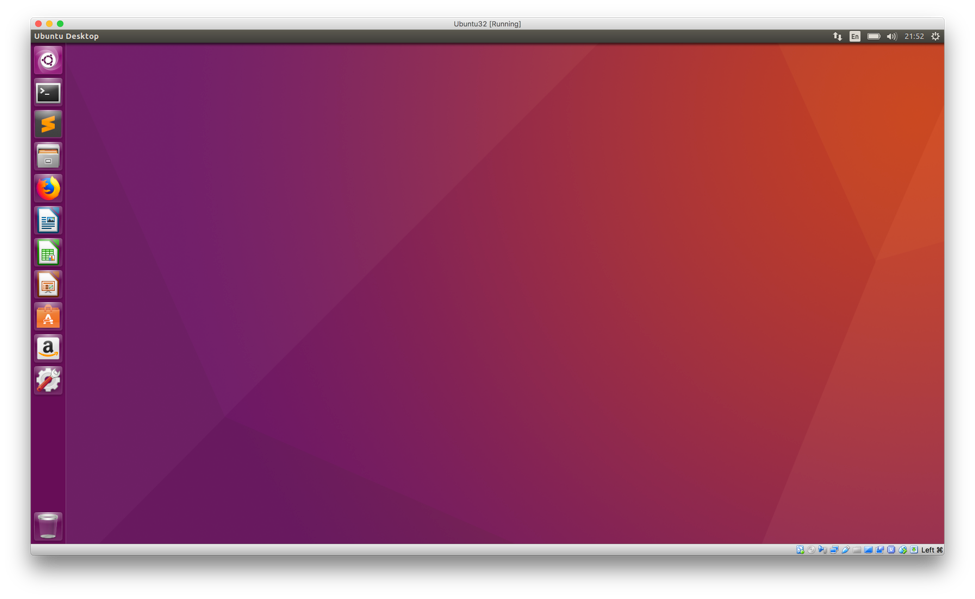


그림 2 Ubuntu 16.04 LTS i386 버전 실행 화면

그 이후, 과제 수행 내용에 있는 libxrandr-dev 및 g++, Bochs, Nasm 설치를 수행하였다.



그림 3 bochs 공식홈페이지에서 다운로드



그림 4 bochs 설치 전 필요한 설정 추가





그림 5 bochs 컴파일 및 설치



그림 6 nasm 설치

그 후 bochs를 통해 SSUOS를 실행시킨다.



그림 7 bochs에서 SSUOS 컴파일 및 실행 화면

1. 구현

PC 부팅 후 기본으로 설정되는 화면 모드는 텍스트 모드로 화면 크기는 가로 80문자, 세로 25문자이며 비디오 메모리 주소는 0xB800에서 시작한다. es 레지스터를 통해 이 주소에 문자와 문자의 색을 지정하여 글자를 출력시킬 수 있다. ax 레지스터로 글자값과 글자 속성을 저장 시킨 후 mov를 통해 해당 좌표에 글씨를 입력하는 것으로 구현하였다.

1. 테스트 프로그램의 구성 및 테스트 주안점



프로그램은 비디오 메모리 지정 및 ax와 bx 초기화, cx를 화면 크기 값 \* 2로 설정해주는 START, 80\*25칸의 비디오를 0으로 초기화 시켜주는 CLS, 그리고 원하는 글자를 출력하는 출력 파트 3가지로 나뉜다.

1. 테스트 및 결과

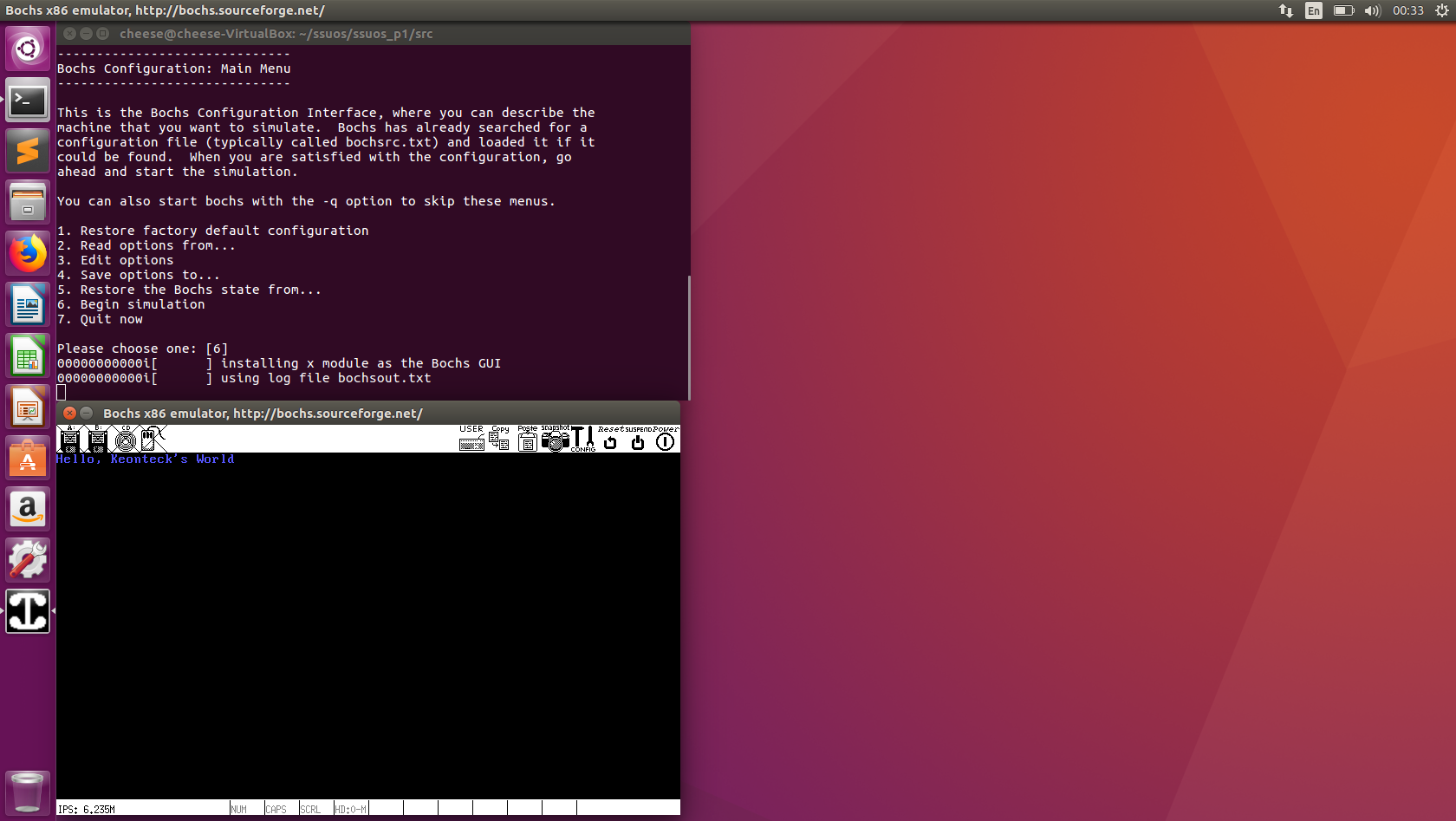


그림 8 bootloader.asm을 변경한 후의 실행 사진

1. 소스코드

[BITS 16]

START:

mov ax, 0xb800 ;비디오 메모리 지정.

mov es, ax ;es에 비디오 메모리 저장

mov ax, 0x00 ;ax 0으로 초기화

mov bx, 0 ;bx 0으로 초기화

mov cx, 80\*25\*2 ;화면 크기가 80\*25이므로 80 \* 25에 글자 용량이 2byte이기에 \*2를 붙임.

CLS:

mov [es:bx], ax ;es:bx 위치에 ax의 값, 0을 넣음

add bx, 1 ;그 이후 bx의 값을 1 증가 시킴

loop CLS ;bx의 값이 80 \* 25 \* 2가 될때까지 루프

mov ah, 0x09 ;ax의 왼쪽 8비트 레지스터에 문자 양식 저장.

mov al, 'H' ;ax의 오른쪽 8비트 레지스터에 문자 아스키 코드 저장.

mov [es:0000], ax ;ax 레지스터를 비디오 메모리 주소에 넣음으로서 글자를 출력함.

mov al, 'e' ;이후 이 과정을 반복하여 "Hello, Keonteck's World"를 화면에 출력함.

mov [es:0002], ax

mov al, 'l'

mov [es:0004], ax

mov al, 'l'

mov [es:0006], ax

mov al, 'o'

mov [es:0008], ax

mov al, ','

mov [es:0010], ax

mov al, ' '

mov [es:0012], ax

mov al, 'K'

mov [es:0014], ax

mov al, 'e'

mov [es:0016], ax

mov al, 'o'

mov [es:0018], ax

mov al, 'n'

mov [es:0020], ax

mov al, 't'

mov [es:0022], ax

mov al, 'e'

mov [es:0024], ax

mov al, 'c'

mov [es:0026], ax

mov al, 'k'

mov [es:0028], ax

mov al, 39

mov [es:0030], ax

mov al, 's'

mov [es:0032], ax

mov al, ' '

mov [es:0034], ax

mov al, 'W'

mov [es:0036], ax

mov al, 'o'

mov [es:0038], ax

mov al, 'r'

mov [es:0040], ax

mov al, 'l'

mov [es:0042], ax

mov al, 'd'

mov [es:0044], ax