Operating System

CS10210 Project

Booting

1. Power on
2. CPU reset
3. POST(Power On Self Test)
4. Load boot sector into 0x7c00
5. Enter protected mode

Boot sector

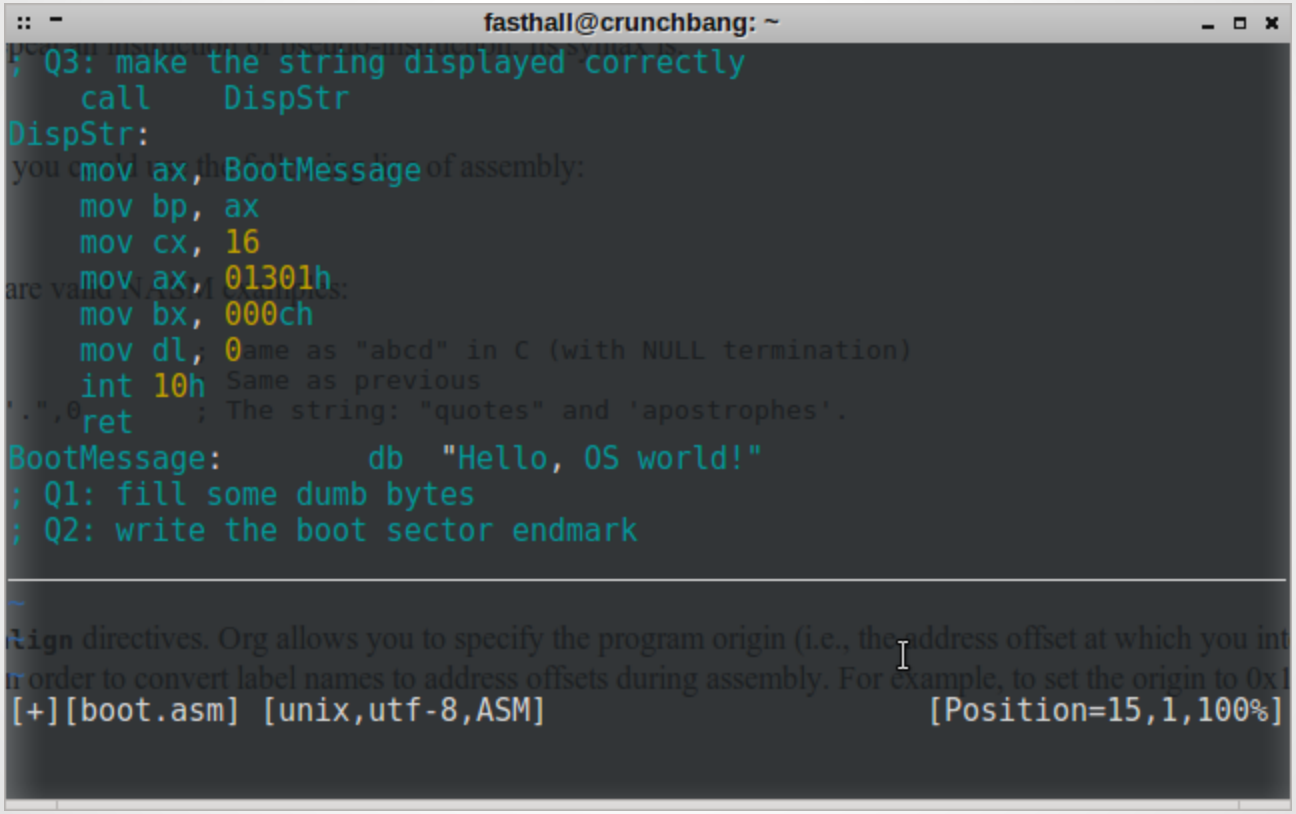
- Boot sector is a region of a data storage device that contains machine code to be loaded into RAM by a firmware.
- The purpose of a boot sector is to allow the boot process of computer to load a program (usually, an operating system)

Boot sector

On a IBM compatible machine, a boot sector:

- is at memory address 0x7C00
- is 512 bytes
- ends with 0xAA55(2 bytes)

Assignment


fasthall@crunchbang: ~
; Q3: make the string displayed correctly
call DispStr
DispStr:
mov ax, BootMessage
mov bp, ax
mov cx, 16
mov ax, 01301h
mov bx, 000ch
mov dl, 0
int 10h
ret
BootMessage: db "Hello, OS world!"
; Q1: fill some dumb bytes
; Q2: write the boot sector endmark
[+][boot.asm] [unix,utf-8,ASM]
[Position=15,1,100%]

INT 10h

BIOS interrupt call 10hex

- The BIOS typically sets up a real mode interrupt handler at this vector that provides video services.

http://en.wikipedia.org/wiki/INT_10H

Assignment

Q1: fill some dumb bytes

Q2: write the boot sector endmark

Reference:

NASM Syntax (http://ece425web.groups.et.byu.net/stable/labs/NASM.html)

Assignment

Try to run it!

```
$ nasm boot.asm -o boot.bin
$ dd if=boot.bin of=boot.img bs=512 count=1
$ qemu -fda boot.img
```

Does it work properly?

Assignment

Of course not.

Q3: Set the address offset.

Explain what this line means and how does it work. Write the answer as the comment and upload your source code.

Deadline: 10/15 23:59