460 Notes #1

CPTS 460 NOTES #1
Booting Operating Systems

Reference: Chapter 3 of the MTX OS book covers booting from FD,HD,CDROM and USB.

Booting OS on Intel x86 based PCs:

(1). PowerOn ===> CPU executes BIOS in ROM :
BIOS initializes itself, then it checks memory and other devices.
Some information needed by BIOS are kept in a small cmos RAM, powered
by a battery.

(2). Then, BIOS looks for a system to boot. The usual booting sequence is
A drive (floppy disk), C drive (Hard disk), CDROM, etc.
which can be changed by programming the BIOS (cmos RAM).

First, BIOS tries to boot from drive A: If there is no diskette is A:,
it will try to boot from C:

Booting from a device is to load the FIRST sector of that device into
(segment,offset)=(0x0000:0x7C00) and let CPU execute that piece of code.

For a floppy disk, there is only ONE boot sector. A hard disk is usually
divided into several Partitions, each with its own boot sector.
The very first sector on a HD is called the Master Boot Record (MBR).

(3). Disk Parameters:
All disks for PCs have Sector size = 512 bytes

Floppy disks have 80 cylinders, each cylinder has 2 heads or tracks, and
each track has 18 sectors.

Hard disks usually use linear Block Addressing (LBA).

To issue I/O operations to a floppy disk (controller), we must specify
(cyl, head, sector). BIOS also uses such values in its disk I/O
functions; NOTE: cyl and head count from 0, but sector counts from 1.

(4). Booting from Hard Disks

A hard disk can be divided into 4 Primary Partitions. The partitions
are recorded in a Partition Table in the MBR at byte offset 0x1BE.
Each Partition Table entry is a structure as shown below.

```
struct partition {
   u8 drive;            /* drive number: 0 for A, 0x80 for C */
   u8 head;             /* starting head */
   u8 sector;           /* starting sector */
   u8 cyl;              /* starting cylinder */

   u8 sys_type;         /* partition type */

   u8 end_head;         /* end head */
   u8 end_sector;       /* end sector */
   u8 end_cyl;          /* end cylinder */

   u32 start_sector;    /* start sector number */
   u32 nr_sectors;      /* number sectors */
};
```

The beginning part of the MBR contains a BOOTER. During booting, BIOS loads
the MBR to (0x0000, 0x7C00) and execute the loaded code in MBR.
The code in MBR is usually a stage1 booter, which loads and excutes a stage2
booter, etc.