**Modifying Josh to natively boot from a fat32 usb drive**

Name: P.L.B. Sampath

Index No: 100476M

**Problems to answer before analysing the boot-loader code and references for answers**

1. What is a boot-loader?

* <http://www.viralpatel.net/taj/tutorial>
* <http://www.brokenthorn.com/Resources/OSDevIndex.html>

1. What are fat file systems and what are the differences between fat12 and fat32?

* <http://www.ntfs.com/fat-systems.htm>l
* <http://home.freeuk.net/foxy2k/disk/disk1.htm>l
* <http://www.easeus.com/resource/fat32-disk-structure.htm>
* <http://en.wikipedia.org/wiki/File_Allocation_Table>
* <http://averstak.tripod.com/fatdox/bootsec.htm>

1. What are the registers in X86 processor and how can we use them?

* http://www.eecg.toronto.edu/~amza/www.mindsec.com/files/x86regs.html

1. How to write a code in assembly?

* <http://cyberasylum.wordpress.com/2010/11/19/assembly-tips-and-tricks/>
* http://en.wikipedia.org/wiki/Assembly\_language

1. What do the assembly codes in boot.asm in Josh mean, which parts doesn't depend on the differences between fat12 and fat32, which parts does and how to change them?

* <http://cyberasylum.wordpress.com/2010/11/19/assembly-tips-and-tricks/>
* http://en.wikipedia.org/wiki/Assembly\_language

1. How to add boot-loader and kernel file to fat32 usb drive to make Josh boot from fat32?

* Guide was given in the moodle

**Analyzing the fat12 boot-loader code**

Let's go through to code of Josh of **fat12** and analyse the code.

* **Parameters**

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| OEM\_ID | OEM identifier indicates the OS that formatted the disk. |
| BytesPerSector | both fat12 and fat32 have 512 bytes per sector. |
| SectorsPerCluster | fat32 has 8 sectors per cluster while fat12 has only one. |
| ReservedSectors | fat32 has 20h reserved sectors while fat12 has only one. |
| TotalFATs | Each has 2 file alocation tables. |
| MaxRootEntries | In fat12 the root directory is fixed and root can contain only fixed # of directories and files. But in fat32 the root directory is treated as another directory that doesn't contain a parent directory. |
| TotalSectors | Number of sectors on the disk. This is defined using 'dw' for a floppy. As the # of sectors in a usb drive greater than 65535 this should be redefined using 'dd'. |
| MediaDescriptor | 0xF0 means a 3 1/2 floppy(1.44MB ) while 0xF8 means a hard drive. As this is a usb pen drive this is treated as a hard disk. |
| SectorsPerFAT | Sectors per file alocation table on the disk. This is defined using 'dw' for a floppy. As the # of sectors in a usb drive greater than 65535 this should be redefined using 'dd'. |
| SectorsPerTrack | No of sectors for a track in a floppy is 12h while it can be taken as 3Dh in a usb drive. |
| SectorsPerHead | No of sectots in one head |
| HiddenSectors | No of sectors from the beginning to start of the partition table |
| TotalSectorsLarge | This can be also used to represent number of sectors if # of sectors cannot be represent using two words. |
| DriveNumber | Physical drive number. |
| Flags | This is defined only for FAT32 disks. They are defined differently for FAT12 andFAT16. If the left-most bit of ExtFlags value is set then only the active copy of FAT is changed. If the bit is cleared then FATs will be kept in synchronization. Disk analyzing programs should set this bit only if some copies of the FAT contain defective sectors. Low four bits define which copy should be active. |
| Signature | Extended boot signature. |
| VolumeID | serial number of the drive. |
| VolumeLabel | Lable of the drive. |
| SystemID | FAT file system type. |

* **Functions**

|  |  |
| --- | --- |
| START: | This adjusts segment registers, creates stacks and post a message and post a message. We can use the same function for fat32 as this doesn't depend of the type of file system. |
| LOADROOT:  LOADFAT:  LOADIMAGE: | These functions are for loading the kernel into memory using other functions. In fat12 there is a fixed root directory. But In fat32 there is not. Because of that these functions should be removed from fat32 boot.asm and we have to find another way of loading the kernel. |
| FAILURE: | This code is for acting when a failure occurs. This displays a message, awaits till a key is pressed, and reboots the machine. No need to modify this. |
| DisplayMessage: | display ASCIIZ string at ds:si via BIOS. When we want to show a message using this function, firstly we should modify 'si' and then we should call the function. This doesn't depend of the differences of fat12 and fat32 and no need to modify. |
| LBACHS: | calculates the Sector, Head, Track numbers from the given LBA address. sets the values of absoluteSector, absoluteHead and absoluteTrack for the use of ReadSectors: function.  No need to modify. |
| ReadSectors: | Reads the hard drive using interrupt 13 till a file record is found.    This function calls LBACHS, DisplayMessage function those should not be modified. Also this uses parameter values of absoluteTrack, absoluteSector and absoluteHead set by the function LBACHS.  Before this is called register values should be set as follows,  AH 02h  AL sectors to read count  CH track  CL sector  DH head  DL drive  ES:BX buffer address pointer  we can get following output from the registers given belove,  CF set of error  AH return code  AL actual sectors read count  no need to modify. |
| ClusterLBA: | calculates the Logical Block Address for a given cluster number which is stored in the 'ax' register. No need to modify. |

After analysing the code I can get an idea about how to modify this.

* I have to modify some parameter values.
* I have to remove functions LOADROOT, LOADFAT and LOADIMAGE. And to add some new functions to do the same procedure.

**Modifying the code**

* parameters are changed considering analysis.
* New three functions
  1. CalcBegOfData
  2. ReadFirstCluster
  3. ReadKernel

are added replacing functions LOADROOT, LOADFAT and LOADIMAGE.

* All other functions are same as fat12 boot.asm.