

# Trabajo Practico

## LiveFDisk

Alejandro Criolani  
Valentino Rosso

C-6065/8  
R-3960/8

28/02/2018



# Informe General

## Introducción:

La función de nuestro programa es la de analizar la tabla de particiones de un disco, cambiar el formato de una partición y mostrar los datos del disco al usuario.

Para esto, utilizamos interrupciones de la BIOS (INT 13) para leer y escribir la tabla de particiones.

Accedemos a la diremoria de video ubicada en 0xB8000 para mostrar el tamaño de las particiones, su formato y el inicio y final de la particion.

## Aclaraciones previas:

### Master Boot Record (MBR)

El Master Boot Record es un tabla de particiones siempre ubicada en el primer sector de un disco.

#### Estructura básica del Master Boot Record Sector:

Offset (dentro del sector)		Longitud (en bytes)	Descripción
En Decimal	En Hexadecimal		
000 - 445	000 - 1BD	446	Área de Código
446 - 509	1BE - 1FD	64	Tabla de Particiones
510 - 511	1FE - 1FF	2	Firma de MBR

#### Tabla de Particiones:

El estandard de la estrutura del MBR contiene una tabla de particiones con cuatro entradas de 16 bytes como lo muestra la siguiente tabla:

Offsets (dentro del sector del MBR)		Longitud (en bytes)	Contenido
En Decimal	En Hexadecimal		
446 - 461	1BE - 1CD	16	Entrada de tabla de la particion primaria N.º 1
462 - 477	1CE - 1DD	16	Entrada de tabla de la particion primaria N.º 2
478 - 493	1DE - 1ED	16	Entrada de tabla de la particion primaria N.º 3
494 - 509	1EE – 1FD	16	Entrada de tabla de la particion primaria N.º 4

Por lo tanto, los discos que usan este estandar no pueden tener mas de 4 particiones primarias.

### **Entrada de una tabla de particiones:**

El MBR usa el siguiente formato para las distintas entradas:

Offsets relativos (entre cada entrada)	Longitud (en bytes)	Contenido
0	1	Indicador de Booteo (80h = activo)
1 - 3	3	Valores CHS iniciales
4	1	Formato de la partición
5 - 7	3	Valores CHS finales
8 - 11	4	Comienzo de Sector
12- 15	4	Tamaño de la partición (en sectores)

Indicador de booteo: Por lo general este byte va a estar siempre en 0h.

Valores CHS iniciales: estos valores dan la localizacion del primer sector de la particion, si esta esta dentro de los primeros 1024 cilindros del disco duro. Si esta mas alla, se setean a los valores maximos permitidos.

Formato de la particion: este byte solo permite 256 posibles valores para indicar todos los tipos de particiones que pueden existir, por ejemplo 0Bh corresponde a FAT32.

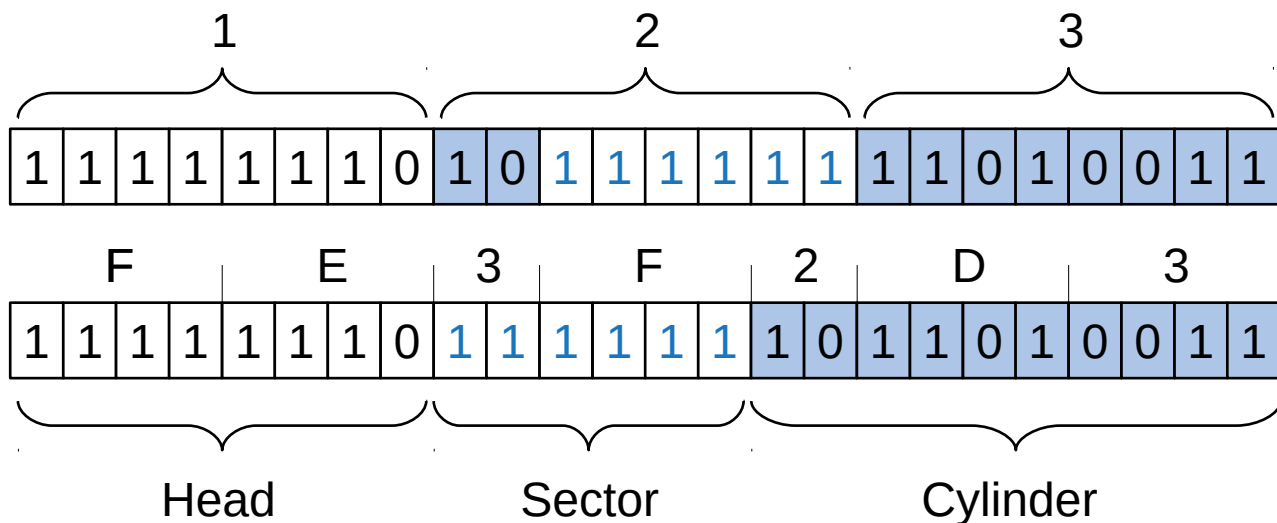
Valores CHS finales: estos valores dan la localización del último sector de la partición.

Comienzo de Sector: Este valor identifica el primer sector de la partición tal y como los valores CHS iniciales. Pero lo hace con un Logical Block Address (LBA) de 4 bytes, por lo que puede localizar el inicio de una partición dentro de los primeros FFFF FFFFh sectores para disco duros de hasta 2048 GB.

Tamaño de la partición: Como los valores LBA del Comienzo de Sector, estos 4 bytes permiten un tamaño de hasta 2048 GB por partición y estan guardados en little-endian.

### **Decodificando los valores CHS:**

El head está en el primer byte por lo que es fácil de analizar. En el segundo byte, los 6 bits menos significativos pertenecen a la información del sector. Los 2 bits restantes pertenecen a los bits mas significativos del cilindro. El tercer byte son los 8 bits menos significativos del cilindro. Como ejemplo:



## Interrupciones del BIOS

Las llamadas de interrupción del BIOS son una facilidad soportada por el BIOS en el que los programas como cargadores de arranques, así como sistemas operativos lo utilizan para acceder al hardware más básico de una computadora.

Algunas de estas interrupciones son:

**INT 10h:** Esta llamada a interrupción controla los servicios de pantalla de una computadora. Para usar esta llamada, hay que cargar en el registro AH el número de la subfunción deseada, cargar los otros registros con los parámetros correspondientes y, finalmente, hacer la llamada a la interrupción.

**INT 13h:** Esta interrupción se encarga de las operaciones típicas, como lectura y escritura, de acceso a las unidades de almacenamiento tales como discos duros y disquetes.

**INT 15h:** Esta interrupción controla los servicios extendidos de una computadora. Antiguamente, esta interrupción se encargaba únicamente de ofrecer acceso a las unidades de cinta. En la actualidad, tiene un amplio rango de funciones como el control del joystick, control del APM (Advance Power Management), establecer el modo de velocidad de la cpu, etc.

**INT 16h:** Esta interrupción se encarga de controlar el teclado de una computadora. Esta interrupción se encarga de obtener las funcionalidades básicas del teclado, es decir, se encarga de recoger las pulsaciones del teclado, obtener el estado del buffer del teclado. La codificación estándar del teclado que ofrece la INT 16h es la de un teclado Estadounidense.

## Ejecución del programa

Primero, la BIOS se encarga de leer el sector de arranque de nuestro diskette. Este sector de arranque siempre está en la dirección 0x7C00. Luego debemos cargar más sectores para poder expandir la aplicación y continuar su ejecución mediante la interrupción 13h. Los valores en los registros encargado de la lectura de los sectores restantes del diskette son:

AH = 0x02, para leer sectores.  
 AL = 4, número de sectores a leer.  
 CH = 0, número de cilindro.  
 CL = 2, número de sector donde comienza la lectura.  
 DH = 0, número de cabeza.  
 DL = 0, leer el primer diskette.

Después nos encargamos de leer el disco a analizar nuevamente con la interrupción 13h.  
 Los valores en los registros son:

AH = 0x02, para leer sectores.  
 AL = 1, número de sectores a leer.  
 CH = 0, número de cilindro.  
 CL = 1, número de sector.  
 DH = 0, número de cabeza.  
 DL = 0x80, leer el primer disco duro.  
 La dirección del registro ES está en 0x50000.

Luego, con el acceso a la memoria de video, mostramos los datos por pantalla.

Tabla de particiones				
Partición	Inicio CHS	Fin CHS	Espacio	Formato
Partición 1	0000 20 21	0003 50 0D	0000C800	EXT4
Partición 2	0003 50 0E	0007 84 3D	00010800	NTFS
Partición 3	0007 84 3E	0010 D3 03	00024800	FAT32
Partición 4	0000 00 00	0000 00 00	00000000	Vacio

En nuestro caso, se observa que el disco tiene tres particiones. En la tabla se observan sus características. En la fila de Partición 4, se observa que solo tiene 0, por lo tanto esta no existe.

A modo de comparación en la siguiente figura se observa la tabla de particion visto desde un editor hexadecimal.

```

0000:01B0 00 00 00 00 00 00 00 00 13 14 9A 45 00 00 00 20
0000:01C0 21 00 83 50 0D 03 00 08 00 00 00 C8 00 00 00 50
0000:01D0 0E 03 07 84 3D 07 00 D0 00 00 00 08 01 00 00 84
0000:01E0 3E 07 0B D3 03 10 00 D8 01 00 00 48 02 00 00 00
0000:01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA
  
```

Finalmente, podemos elegir una partición para formatear en tres tipos distintos que son EXT4, FAT32 y NTFS. El caso de “vacío” no representa un formato desconocido, si no que para una entrada de partición vacía.

Para cambiar este formato, volvimos a usar la interrupción 13h, pero en este caso en modo escritura. Los valores en los registros son:

AH = 3, para escribir sectores.  
 AL = 1, escribimos un sector.

CH = 0, número de cilindro.  
CL = 1, número de sector.  
DH = 0, número de cabeza.  
DL = 0x80, primer disco duro.

## Problemas encontrados

El único problema que se nos presentó en el momento del desarrollo del programa fue que la BIOS cuando bootea, solo carga el primer sector. Este primer sector de solo 512 bytes es de un tamaño reducido para las funciones que necesitabamos implementar.

Mediante las llamadas a interrupciones, nos encargamos de leer los sectores restantes y así poder completar el programa.

## Posibles extensiones

Una posible extensión del trabajo práctico es ver el tamaño del disco en megabytes o gigabytes, también ver su espacio libre y ocupado. Otra extensión posible es agregar opciones para eliminar, modificar tamaños o crear nuevas particiones con un tamaño personalizado.