UEFI

Introducción

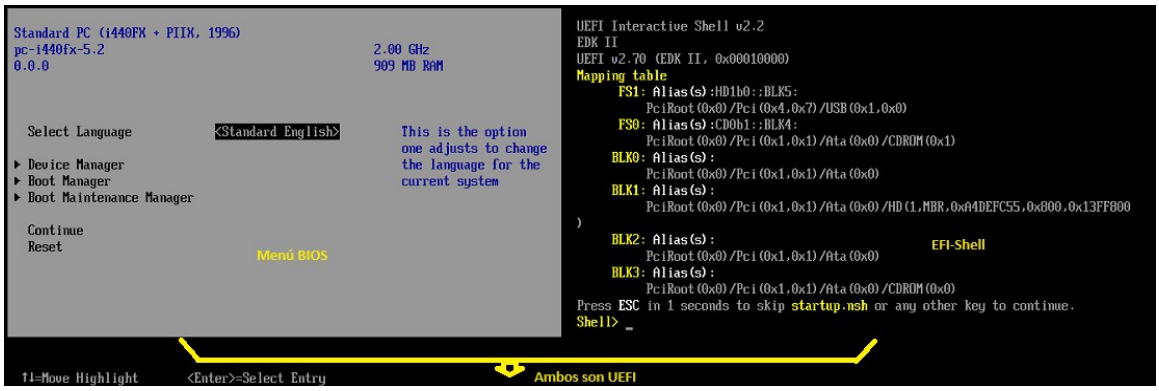
la Unified Extensible Firmware Interface (UEFI), es un especificacion que define la arquitectura del firmware de la plataforma usado pora inicializar el hardware de la computadorea y es la interface para la interacción con el sistema operativo.

UEFI es independiente de la plataforma y el leguanje de programación, pero usualmente C es usado por referencia para la implementacion de TianoCore EDKII.

Contrario a sus predecesor el BIOS el cual es un estandar *de facto* originalmente creado por IBM como software propietario, UEFI es un estandar abierto mantenido por la consorcio industrial.

La propuesta UEFI no es solo reemplazar al BIOS, sino tambien proveer un entorno que permita diagnosticar y resolver problemas relacionados con el arranque del OS o funcionamiento de hardware, por lo que provee un ambiente EFI-Shell, el cual cuenta con un rico conjunto de comandos que extienden y mejoran la capacidades de la interfaz.

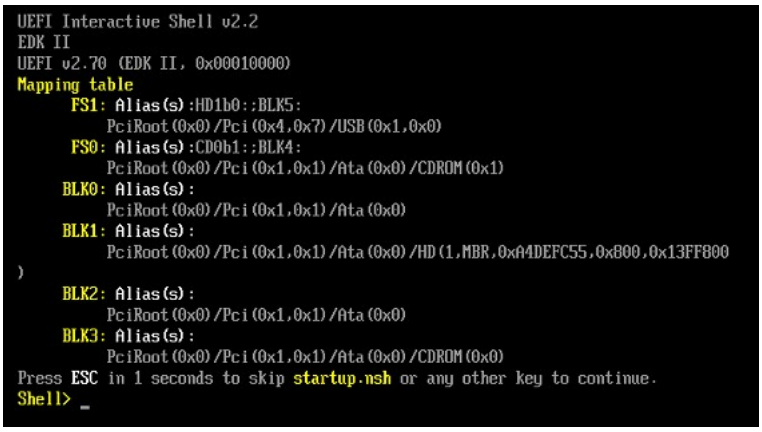
Nota: *aunque usamos la plabara* ***BIOS*** *no nos referimos al PC BIOS, sino a la parte de menus grafica de EDKII a la cual estamos acostumbrados desde las PC compatibles, la cual forma parte de EFI, pero guarda la apariencia para generar un ambiente homogeneo, asi pues tendremos los menus "BIOS" y la interface EFI-Shell*



EFI-Shell

La interface no es muy diferente a una Ventana de Comandos en Windows o a la terminal Bash de Linux, compuesta de un conjunto de comandos, los cuales pueden agruparse en Scripts para realizar tareas mas complejas, se presentan los comandos mas comunes para introducir al uso de la plataforma, conforme se avance en los temas se iran agreagando mas comandos, el proposito no es cubrir todo los comandos referentes al EFI-Shell, es mas una guia rapida para que el usuario comience a usar la interface.

Se muestra a continuacion la ventana de la interface:



Examinando la imagen de arriba hacia abajo tenemos primero el titulo que nos indica cual es la version y modelo de UEFI que estamos manejando (en este caso es EDKII).

La tabla de Mapeo, indica cuales dispositivos estan instalados en nuestro sistema, tenemos que FS0 FS1 se refieren a dispositivos Flash, mientras que BLK1 BLK2 y BLK3 son dispositivos de Bloque de puertos PCI, dependiendo de nuestro hardware, podriamos tener listados otros tipos de dispositiovos.

Podemos observar que algunos dispositivos cuentan con un *Alias(s)* el cual indica que como es que el BIOS lo tratara, podemos ver que FS1 es una unidad Flash extraible, pero que es tratado como si fuera un disco duro HD1B0 el cual es a su vez un dispositivo de bloque BLK5.

El dispositivo de BLK3 es tratado como un CDROM instalado en un puerto PCI, observamos que la direccion de lbase del arbol de dispositivos PCI es 0x0 el dispositivo esta listado como el primero *Pci (0x1, 0x1)* es tratado como un dispositivo tio ATA, ya que fisicamente es una unidad CDROM, por lo que se le asigna un controlador de dispositivo acorde.

Mencion especial recibe el dispositivo BLK1, el cual es un HD que posee un sistema Operativo, ya que tiene un Master Boot Record (MRB), y nos indica la direccion en hexadecimal que el BIOS le ha asignado.

Para mayor informacion sobre la asignacion de las direcciones en memoria consulte:

White Paper
Jenny M Pelner
James A Pelner
Firmware Architects
Intel Corporation

Minimal Intel Architecture Boot Loader

Bare Bones Functionality Required for Booting an Intel Architecture Platform

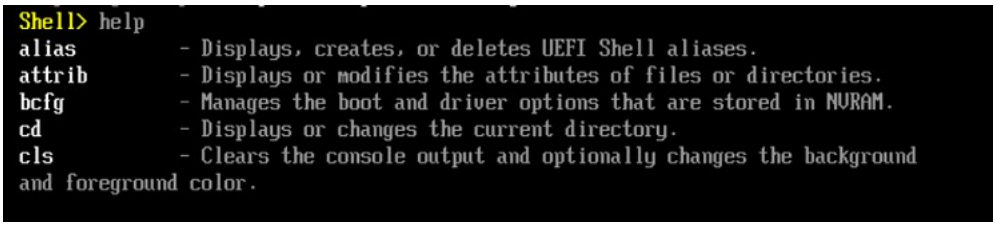
En la seccion de Mapa de la memoria, asi como la bibliografia de referencia del documento.

Por ultimo tenemos la linea con la plabra **startup.nsh**, este es un comportamiento por defacto del Shell EFI, al iniciar buscara automaticamente ese archivo, el cual podria contener un script de recuperacion o de diagnostico provisto por el fabricante del equipo de computo o tareas de rutina que nostros hallamos escrito y que deseamos se ejecuten al inicio, en caso de no encontrarse ese archivo o de haber presionado **ESC** se nos mostrara ahora si el Prompt de EFI-Shell.

En ese momento estamos listos para trabajar con nuestra Comand Line Interface (CLI), en caso de que nuestro sistema no cuente habilitado el EFI-Shell tendremos que instalarlo, consulte el Anexo A para mas informacion, mostraremos algunos comandos usuales a continuacion.

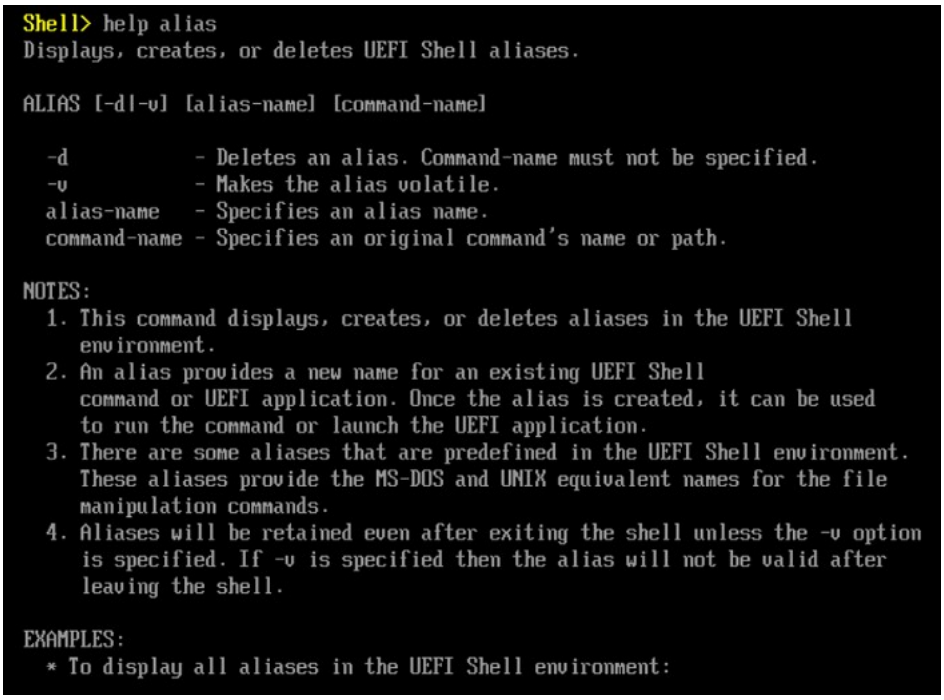
Comandos mas comunes de EFI-Shell

help



Nos muestra todos los comandos disponibles en nuestro Shell, la informacion puede ocupar varias pantallas por lo que para poder verla usaremos las teclas de *AvPag Repag* para movernos en el listado. es posible usar configuraciones *more less* para mostrar en partes la informacion, justo como se haria en Ventana de Comandos o Bash.

Si desesamos información de un comando en especificao escribimos *help* ***commando***, como se muestra en la imagen:



Al igual que en otras CLI, los comandos usaran parametros en su sintaxis, donde [] significa parametros que pueden estar presentes o no, asi como los parametros dentrod e <> son obligatorios, *help* tambien mostrara para cada comando la sintaxis la descripcion de los parametros Notas y ejemplos de uso asi como informacion adicional, se recomienda familiarizarse con la interpretacion de la linea de sintaxis:

```
ALIAS [-d|-v] [alias-name] [command-name]
```

En este caso el comando ALIAS tiene tres parametros los cuales son opcionales y en el primer parametro [-d | -v] indica que se puede elegir entre dos parametros para esta opcion (-d o -v), asi el simbolo | se interpreta como el condicional logico *OR*.

Se recomienda usar el comando *help* para encontrar mas informacion sobre los comandos, asi como acudir a la bibliografia provista al final de este documento.

Anexo A

Bibliografia:

Configuracion de UEFI en sistemas

Intel-Basic Instructions for Using the Extensible Firmware Interface (EFI)

Intel-Shell Command Reference Manual

Intel-Minimal Intel Architecture Boot Loader

https://www.tianocore.org/

EDKII User Manual