



Basic Instructions for Using EFI (Extensible Firmware Interface)

***for Server Configuration
on Intel® Server Boards and
Intel® Server Systems***

Revision 1.0

January 2, 2008

Enterprise Platforms and Services Division

Table of Contents

1.1	Booting to the EFI Shell	4
1.2	Basic EFI Commands	4
1.3	Using Removeable Media with the EFI Shell.....	4
1.4	Update and Configuration Utilities for EFI	5
1.4.1	System Stack Updates	5
1.4.2	BIOS and BMC Configuration	6
Appendix A: EFI Commands.....		7
Appendix B: Sample Scripts.....		9
	Typical Update script.....	9
	Syscfg configuration script	9

Disclaimers

Information in this document is provided in connection with Intel® products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

This document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, Pentium, Celeron, and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Copyright © Intel Corporation 2008. *Other names and brands may be claimed as the property of others.

1.1 Booting to the EFI Shell

All Intel® Server Boards using the 5000, 5400, 7000, 3000 and 3200 series chipsets support booting to an EFI shell without any additional Operating System present.

To boot to the EFI shell, press **F2** when prompted during POST to enter the BIOS utility. Within the BIOS utility, arrow over to the Boot Devices option and press **<Enter>**. Then scroll down and highlight the EFI Shell option. Press **<Enter>** to go directly to the EFI Shell.

The EFI shell can be set as the first item in the Boot Order Menu so that the system always boots into EFI. (By default, the EFI shell is the last item in the Boot order option list.) Simply enter the BIOS utility (**F2**) and arrow over to Boot Options and press **<Enter>**. The highlight the first option and press **<Enter>** and arrow down to the EFI Shell option and select **<Enter>**. Then press **F10** to save this as the new boot order.

Upon booting to the EFI shell, the embedded OS will always look for a file named startup.nsh (searching through the “path” defined). Startup.nsh is the equivalent of autoexec.bat in the DOS/Windows environment. After running this startup script (if found), the user will be presented with the command line prompt `Shell>`.

1.2 Basic EFI Commands

EFI commands are very similar (often identical) to common DOS and Linux commands. The most frequently used commands include:

- `dir` (or `ls`): lists directory contents
- Other navigation commands: `cd`, `mkdir`, `move`, `rm` (remove)
- `help`: displays help
- `exit`: leaves the EFI shell and returns to BIOS utility.
- `edit`: starts a basic text editor with onscreen function key help
- `map`: Commonly used to refresh mapped drives as `map -r`.
- Common screen commands: `cls`, `echo`

Note: for most commands, EFI does not care about capitalization. However EFI can be case sensitive for file paths and names depending on the media.

For a full list of commands, see Appendix A

1.3 Using Removeable Media with the EFI Shell

Media (USB Key, CD, harddrives) must be formatted as FAT (16 or 32) to be readable from EFI.

CDs do not need to be bootable nor contain the EFI OS. The server is capable of booting into EFI and automatically mapping the CD if it is present in the drive at boot. This is true for any FAT formatted CD (including Windows or DOS based CDs). This allows scripts named `startup.nsh` on the FAT formatted CD to automatically run.

If the CD is bootable, make sure that you have EFI selected as the first boot media and NOT the CD so that you will not boot into DOS or Windows instead of EFI.

Note: Instructions for making a CD in FAT format are readily available on the Internet and with commercial CD burning software, simply search on "build CD" or "build ISO". (An ISO is the formatted file to burn to the CD).

The first USB device found will show in the mapping as `fs0`. (File system 0). The next as `fs1`, etc.

If media is added after the initial mapping or changed at any time while in the EFI shell, type the command:

```
map -r (refreshes the mounting, mapping).
```

One can change from the shell to the media by typing the `fs#` followed by a colon(:) . The prompt will display the current directory. So

```
shell> fs0: will become fs0:>
```

Note: once you have changed to media, you may change to other media including RAMdisks (commonly `fsz`;) but there is no need to return to the base shell.

1.4 Update and Configuration Utilities for EFI

Intel has written multiple utilities to do common tasks from EFI. The most common are the system stack update and configuration tools to manipulate BIOS, Firmware and FRUSDRs. These tools are available in the `EFI_Setup_and Configuration Toolkits` on Intel's support website. Additionally configuration tools are on the resource CD and update tools are in the System Update Packs.

1.4.1 System Stack Updates

1. Download the System stack Update Pack (FSUP) for your board model from support.intel.com.
2. Unpack the contents of the zip file onto a USB key.
3. Boot to the EFI shell.
4. Plug in the USB device and type `map -r`.
5. Type the name of the mounted file system (usually `fs0`:).
NOTE: Remember to include the colon after the number.

6. To update all of the basic stack, type “startup”.

To update only certain components, cd to the EFI scripts directory and type the name of the component you wish to update.

Warning! EFI should never be used to update expander backplanes (those Intel backplanes supporting 4 or more drives but having only 2 drive port connections.)

6. Depending on the update script, the server may automatically reboot after the update or you can type “reset” to reboot.

Note: typing startup invokes the startup.nsh script which Intel codes to do all the included updates of the pack.

1.4.2 BIOS and BMC Configuration

The most common configuration tool from Intel is Syscfg and the utility is available for multiple OS using the same commands. These commands can be found in the readme and the User Guide included with the utility. A configuration script including the most commonly used commands are listed in Appendix B.

Note: EFI, like DOS and Linux, can only run programs from the root that are listed in the environmental path. Directories can be added or removed from the path using the “set” command. Specifically to run Syscfg commands type:

```
set SYSCFG_PATH fs0:\syscfg
```

where syscfg is the directory holding the utility in the root of your first USB drive.

Warning! This command is case sensitive!

The path setting will hold across reboots, but not across complete AC power cycles.

To clear something out of the path use “set -d (item to clear)”.

Other utilities are also available. See the EFI update pack for your server board. Some utilities are only available in Windows including the BIOS logo and AMIBCP (one time master BIOS reconfiguration tool). Additional EFI utilities are available that are not developed by Intel including DMIEDIT from AMI.

Appendix A: EFI Commands

This appendix will provide a chart of commands. A document covering full syntax, variables, wildcards, redirection, error handling, etc is available at http://www.intel.com/software/products/college/efishell/images/efi_shell_cmd_1_1.pdf.

Command	Description
alias	Displays, creates, or deletes aliases (can alias commands, drives, executables)
attrib [+][-][a s h r] file directory	Displays or changes the attributes of files or directories
bcfg	Displays/modifies the driver/boot configuration
break	Executes a debugger break point.
cd	Change directory
cls	Clears the screen, can change background color
comp file1 file2	Compares the contents of two files for up to 10 differences
connect	Binds an EFI driver to a device and starts the driver
cp [-r][-q] src file [dst]	Copies one (r=all recursively)files/directories to another location
date	Displays the current date or sets the date in the system
dblk	Displays the contents of blocks from a block device
devices	Displays the list of devices being managed by EFI drivers
devtree	Displays the tree of devices that follow the EFI Driver Model
dh	Displays the handles in the EFI environment
disconnect	Disconnects one or more drivers from a device
dmem	Displays the contents of memory
dmpstore	Displays all NVRAM variables
drivers	Displays the list of drivers that follow the EFI Driver Model
drvcfg	Invokes the Driver Configuration Protocol
drvdiag	Invokes the Driver Diagnostics Protocol
echo	Displays messages or turns command echoing on or off
edit	Edits an ASCII or UNICODE file in full screen.
EfiCompress	Compress a file
EfiDecompress	Decompress a file
err	Displays or changes the error level
exit	Exits the EFI Shell
getmtc	Displays the current monotonic counter value
goto	Makes batch file execution jump to another location
guid	Displays all the GUIDs in the EFI environment
help [-b]	Displays commands list or verbose help of a command. -b one page at a time
hexedit	Edits with hex mode in full screen
load	Loads EFI drivers (ex. Load ipmi.efi)
LoadBmp -w(seconds)	Displays a Bitmap file onto the screen
LoadPciRom	Loads a PCI Option ROM image from a file

ls [-b -r -a]	Display filelist (one page recursive attributes)
map [-r -v -d]	Displays, resets, deletes mappings with verbose option
memmap	Displays the memory map
mkdir dirname	Creates a directory at current location
mm	Displays or modifies MEM/IO/PCI
mode [row col]	Displays or changes console output
mount BlkDevice [sname]	Mounts a file system on a block device. Lost at next map -r.
mv src dest	Moves one or more files/directories to destination
OpenInfo	Displays the protocols on a handle and the agents
pause	Prints a message and suspends for keyboard input. Options are q to quit or any other key resumes the script.
pci	Displays PCI devices or PCI function configuration space
reconnect	Reconnects one or more drivers from a device
reset [-w -s] [string]	Resets the system with warm reboot or complete shutdown. Can pass a string to the reset service.
rm [-q] file dir	Deletes one or more files or directories (-q no confirmation)
set [-d -v -b] [sname [value]]	(Set) Displays,(set -d) deletes, changes, (set sname value) creates environment variables
setsize	Sets the size of a file
stall	Stalls the processor for some microseconds
time	Displays the current time or sets the time of the system
touch [-r] filename	Sets the time and date of a file to the current time and date
type [-a -u] file	Displays the contents of a file (ASCII or Unicode)
unload	Unloads a protocol image
ver	Displays the version information
vol [fs] [VolLabel]	Displays volume information of the file system

Batch only commands
 for/endfor
 if/endif

Appendix B: Sample Scripts

Typical Update script

Update pack includes frusdr.efi, fwpiaupd.efi, ipmi.efi, and iflash32.efi and master.cfg plus *.hex and Rxx.cap files)

```
Cls
echo Update FRUSDRS
echo ""
frusdr /cfg master.cfg

echo Update BMC flash using FWPIAUPD utility
echo ""
fwpiaupd -u -o -pia -ni -p X11.hex

echo ""
echo Update BIOS
iflash32 /u -ni R0029.cap

echo "updates done. Reboot will occur automatically.
reset
```

Syscfg configuration script

```
set SYSCFG_PATH fs0:\syscfg
cd syscfg
syscfg /bqb disable \\disable quiet boot
syscfg /bcr COM1 9600 CTS VT100 \\set console redirection

syscfg /le 1 static 192.168.1.200 255.255.255.0
\\enable lan channel 1 with static IP address

syscfg /u 3 admin password \\set user 3 name and password
syscfg /ue 3 enable 1 \\enable user 3 on channel 1
syscfg /up 3 1 admin sol
\\give user 3 admin privilege to channel 1 including SOL
syscfg /sole 1 enable admin 19200 7 200 \\enable SOL on ch 1 for admin

syscfg /d BIOS > fs0:\BIOS.txt \\display BIOS settings and save to \\BIOS.txt
on root of first USB drive.
syscfg /d LAN 1 > fs0:\L1.txt
syscfg /d LAN 2 > fs0:\L2.txt
syscfg /d Channel 1 > fs0:\ch1.txt
syscfg /d SOL 1 > fs0:\sol.txt
syscfg /d user 3 > fs0:\u3.txt
```