System Tools User Guide for Intel® Management Engine Firmware 6.0 (Corporate)

User Guide

June 2009

Revision 0.60

Intel Confidential

Contents

Figures

Tables

Revision History

Revision Number	Description	Revision Date
0.5	Alpha	04/01/2009
0.55	Adding description for Consumer SKU Update MEINFO/MEMANUF error code list	04/15/2009
0.57	Add SKU manager Update Intel® RPAT support Remove MAC support for FPT Update FOV table	05/28/2009
0.58	MEINFO table update	06/03/2009
0.60	MEINFO example update Update OEMID for FWupdate	06/08/2009

§

1 Introduction

The purpose of this document is to provide guidance on the usage of the tools that are used in the platform design, manufacturing, testing and validation process.

1.1 Terminology

1.1.1 In General

Acronym or Term	Definition
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BIOS	Basic Input Output System
CPU	Central Processing Unit
DIMM	Dual In-line Memory Module
DLL	Dynamic Link Library
EC	Embedded Controller
EEPROM	Electrically Erasable Programmable Read Only Memory
FW	Firmware
GbE	Gigabit Ethernet
HECI (deprecated)	Host Embedded Controller Interface
IBV	Independent BIOS Vendor
ID	Identification
Intel® ME	Intel® Management Engine
Intel® MEI	Intel® Management Engine Interface (renamed from HECI)
ISV	Independent Software Vendor
JTAG	Joint Test Action Group
KVM	Keyboard, Video, Mouse
LAN	Local Area Network
LED	Light Emitting Diode
NVM	Non Volatile Memory
NVRAM	Non-Volatile Random Access Memory
OOB	Out of Band
OS	Operating System

Acronym or Term	Definition
PAVP	Protected Video and Audio Path
PCI	Peripheral Component Interconnect
PCIe*	Peripheral Component Interconnect Express
PHY	Physical Layer
PRTC	Protected Real Time Clock
RNG	Random Number Generator
RSA	RSA is a public key encryption method.
RTC	Real Time Clock
SDK	Software Development Kit
SHA	Secure Hash Algorithm
SMBus	System Management Bus
SPI Flash	Serial Peripheral Interface Flash
TCP / IP	Transmission Control Protocol / Internet Protocol
UI	User Interface

1.1.2 Intel® Management Engine

Acronym or Term	Definition
3PDS	3rd Party Data Store
Agent	Software that runs on a client PC with OS running.
Intel® AT	Intel® Anti-Theft Technology. Contains Intel® AT-d (previously known as Danbury) and Intel® AT-p (previously known as TDT).
CBM	ME CBMs - Core Base Modules. Refer to Figure: ME FW partitioning
CEM	ME CEMs - Core Extension Modules. Also called ME CS. Refer to Figure: ME FW partitioning
Corwin Spring	See Intel® Remote Wake Technology
End User	The person who uses the computer (either Desktop or Mobile). In corporate, the user usually does not have an administrator privileges. The end user may not be aware to the fact that the platform is managed by Intel® AMT.
Host or Host CPU	The processor that is running the operating system. This is different than the management processor running the Intel® Management Engine Firmware.
Host Service/Application	An application that is running on the host CPU.
INF	An information file (.inf) used by Microsoft operating systems that support the Plug & Play feature. When installing a driver, this file provides the OS the necessary information about driver filenames, driver components, and supported hardware.

Acronym or Term	Definition
Intel® AMT Firmware	The Intel® AMT Firmware running on the embedded processor.
Intel® Identity Protection Technology (Intel® IPT)	Also known as Sentry Peak, Intel® IPT is a chipset feature that helps protect online banking and transactions by adding hardware authentication mechanisms.
Intel® Management Engine Interface (Intel® MEI)	Interface between the Management Engine and the Host system.
Intel® MEI driver	Intel® AMT host driver that runs on the host and interfaces between ISV Agent and the AMT HW.
Intel® Quiet System Technology (Intel® QST)	Fan speed control architecture that allows multiple sensors to control a single fan as well as allow a single sensor control of multiple fans.
Intel® Remote Wake Technology (Intel® RWT)	Also known as Wake-on-Event or Corwin Springs Intel® RWT makes a PC remotely accessible to applications even when it is in a low power state.
Intel® Remote PC Assist Technology (Intel® RPAT)	Also known as Castle Peak, Intel® RPAT is a consumer PC manageability technology that helps connect PC support desks to a User's PC regardless of the state of the OS.
IT User	Information Technology User. Typically very technical and uses a management console to ensure multiple PCs on a network function.
LMS	Local Management Service, A SW application which runs on the host machine and provide a secured communication between the ISV agent and the Intel® Management Engine Firmware.
Intel® ME	Intel® Management Engine, The embedded processor residing in the chipset GMCH.
Intel® MEBx	Intel® Management Engine BIOS Extensions
MECI	ME-VE Communication Interface
NVM	Non-Volatile Memory. A type of memory that will retain its contents even if power is removed. In the Intel® AMT current implementation, this is achieved using a FLASH memory device.
OOB interface	Out Of Band interface. This is SOAP/XML interface over secure or non secure TCP protocol.
OS not Functional	The Host OS is considered non-functional in Sx power state any one of the following cases when system is in S0 power state: • OS is hung • After PCI reset • OS watch dog expires • OS is not present
SP	Sentry Peak
System States	Operating System power states such as S0. See detailed definitions in system state section.
TDT	Theft Deterrence Technology. Previous name for AT-p, which is part of the Intel® Anti-Theft Technology.
UIM	User Identifiable Mark

Acronym or Term	Definition
Un-configured state	The state of the Intel® Management Engine Firmware when it leaves the OEM factory. At this stage the Intel® Management Engine Firmware is not functional and must be configured.

1.1.3   System States and Power Management

Acronym or Term	Definition
G3	A system state of Mechanical Off where all power is disconnected from the system. G3 power state does not necessarily indicate that RTC power is removed.
M0	Intel® Management Engine power state where all HW power planes are activated. Host power state is S0.
M1	Intel® Management Engine power state where all HW power planes are activated however the host power state is different than S0 (Some host power planes are not activated). Host PCI-E* interface are unavailable to the host SW. **This power state is not available in Ibex Peak.**
M3	Intel® Management Engine power state where all HW power planes are activated however the host power state is different than S0 (Some host power planes are not activated). Host PCI-E* interface are unavailable to the host SW. Main memory is not available for Intel® Management Engine use.
M-Off	No power is applied to the management processor subsystem. Intel® Management Engine is shut down.
OS Hibernate	OS state where the OS state is saved on the hard drive.
S0	A system state where power is applied to all HW devices and system is running normally.
S1, S2, S3	A system state where the host CPU is not running however power is connected to the memory system (memory is in self refresh).
S4	A system state where the host CPU and memory are not active.
S5	A system state where all power to the host system is off, however the power cord is still connected.
Shut Down	All power is off for the host machine however the power cord is still connected.
Snooze mode	Intel® Management Engine activities are mostly suspended to save power. The Intel® Management Engine monitors HW activities and can restore its activities depending on the HW event.
Standby	OS state where the OS state is saved in memory and resumed from the memory when mouse/keyboard is clicked.
Sx	All S states which are different than S0.

1.2　　　Reference Documents

Document	Document No./Location
OEM Bring Up Guide	Release kit
Firmware Variable Structures for Intel® Management Engine and Intel® Active Management Technology 6.0	
PCH EDS	
PCH SPI Programming Guide	Release kit

§

2   *Preface*

2.1   Overview

The system tools described in this document cover the tools used to create, modify and write binary image files, manufacturing testing, ME setting information gathering, and ME firmware update with is located in \Tools\System tools\ directory. If you need more detail for other tools, you may refer to tools user guide in other directory in firmware release.

All the tools which are included in Ibexpeak firmware release kit are designed for Ibexpeak platform only. These tools will not work properly on any other legacy platforms (Santa Rosa, Weybridge, Montevina, McCreary platform). The tools designed for other generations will not work properly on Ibexpeak platform either.

- All the features listed in this document are available for vPro platform with Intel® Management Firmware 6.0. There are some features are designed and will only work on Intel ® vPro platform.

2.2   Image Editing Tools

The following tools create and write flash images:

- Flash Image Tool (FITC)—combines the GBE, BIOS, PDR and ME firmware into one image and also config softstraps and NVARs for ME setting that can be programmed by a flash programming device or the Flash Programming Tool (FPT).

- Flash Programming Tool—programs the flash memory. This tool can program individual regions or the entire flash device and also modify some of ME settings (FOV) after ME is flashed on the SPI part.

- FWUpdate—updates the ME firmware code region on of a flash device that has already been programmed with a complete SPI image. (Firmware update tool provided by Intel only works on the platform that support this feature)

2.3   Manufacturing Line Validation Tools (MEMANUF)

Manufacturing line validation tools allow testing of Intel® ME, Intel®AMT and Intel® VE functionality immediately after the platform silicon is generated. These tools are designed to be able to run quickly and on simple operating systems, such as MS-DOS 6.22, Windows* 98 DOS, FreeDOS, and DRMK DOS. The Windows versions are written to run on Windows* XP (SP1/2) and Windows Vista*. Details refer to section 2.5

2.4 ME setting checker Tool (MEINFO)

ME setting checking tools will retrieve and display some of the ME setting, ME firmware version and also FW capability on the platform.

2.5 Operating System Support

Table 1. OS support for tools

	MS DOS V 6.22	Windows 98 DOS	DRMK DOS Version 8.00	FreeDOS Version 1.1.32a	PC-DOS Version 7.01	PC-DOS Version 7.00C/V (US)	Windows PE (Based on Vista	Windows Server 2003 w/	Windows Server 2008 w/	Windows XP SP3 32/64	Windows Vista 32/ 64 SP1
Flash Image Tool										X†	
Flash Programming Tool	X	X	X	X	X		X			X†	X†
MEManuf	X	X	X	X	X		X*			X	X
MEInfo	X	X	X	X	X		X			X	X
Update Params tool						X					
ME FW Update Tool	X	X	X	X	X		X			X	X

NOTE: * Not required for consumer platform

NOTE: † 32 bit only

NOTE: 64 bit support does **NOT** mean that a tool is compiled as a 64 bit application – but that it can run as a 32 bit application on a 64 bit platform.
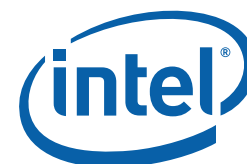
2.6 Generic system requirements

Integration validation tools that run locally on the System Under Test with Intel®
Manageability Engine require one or more of the following services to be installed:

- Intel® Management Engine Interface (Intel® MEI) driver.

- Intel® AMT Local Manageability Service (LMS)

Check individual tool descriptions for the exact requirements.

Table 2. Tools Summary

Tool Name	Feature Tested	Runs on Intel® AMT device
MEManuf and MEManufWin	Connectivity between ME Devices	X
MEInfo and MEInfoWin	Firmware Aliveness—outputs certain ME parameters	X
Flash Programming Tool (fpt)	Programs the image onto the flash memory	X
Firmware Update	Updates the firmware code while maintaining the values previously set	X

2.7 Error return

Tools will always return 0/1 for the error code. 0 = success, 1= error. Detail error
code is displayed on the screen and stored on an error.log file in the same directory
where you run the tools. Detail error code list please refer to Appendix.

§

3   *Flash Image Tool (FITC)*

The Flash Image tool (FITC.exe) creates and configures a complete SPI image file for Ibexpeak platform. The FITC takes a combination of the following regions in the form of binary files, and assembles them into a single flash image:

- BIOS

- Gigabit Ethernet

- Intel® Management Engine (ME)

- Platform Descriptor Region (PDR).

FITC also creates and allows configuration of the Flash Descriptor Region, which contains configuration information for platform hardware and firmware. When combining the four regions listed above, it also integrated the Flash Descriptor Region into the SPI flash image.

The user is able to manipulate the complete SPI image via a Graphical User Interface (GUI) and change the various chipset parameters to match the target hardware. Various configurations can be saved to independent files, obviating the need to recreate a new image each time.

FITC supports a set of command line parameters that can be used to build an image from the command prompt or from a makefile. A previously stored configuration can be used to define the image layout making interacting with the GUI unnecessary.

Note that FITC does not program the flash device. FITC only generates a complete SPI image file. This complete SPI image must be programmed into the flash, either using FPT, any third-party flash burning tool or some other flash burner device.

3.1   System Requirements

FITC will run on Windows* XP or Windows Vista* (32 bit). It is not necessary for the tool to run on an ME-enabled system.

3.2   Flash Image Details

A flash image is composed of five regions. The locations of these regions are referred to in terms of where they can be found within the total memory of the flash.

Figure 1. Firmware Image Components

Descriptor	ME/VE				GbE	PDR	BIOS
	Intel® AMT	Intel® RPAT, iRWT, IPT	Braidwood Technology	Intel® QST			
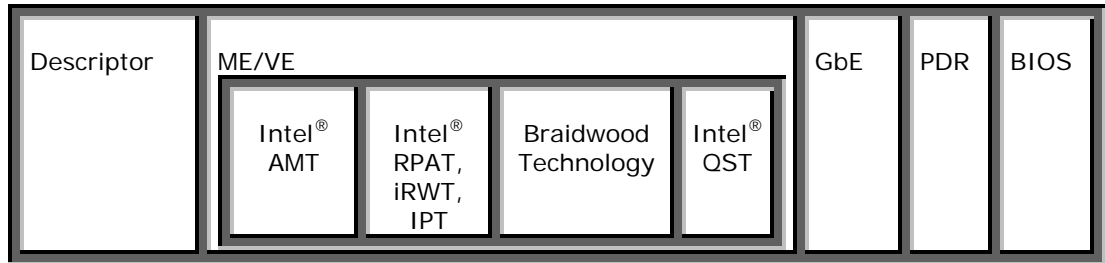
The following is a description of these regions:

- Descriptor—takes up a fixed amount of space at the beginning of the flash memory. The descriptor contains information, such as, space allocated for each region of the flash image, read-write permissions for each region, and a space which can be used for vendor-specific data.

Note: This region MUST be locked before the serial flash device is shipped to end users. Please see the 4.10 below for more information. Failure to lock the Descriptor Region will leave the Intel® AMT device vulnerable to security attacks.

- ME—region that takes up a variable amount of space at the end of the descriptor. Contains code and configuration data for Intel® ME applications, such as Intel® AMT technology, Intel® NAND, Intel® AT-P, and Intel® Quiet System Technology (Intel® QST).

- GbE—region that takes up a variable amount of space at the end of the ME region. Contains code and configuration data for Gigabit Ethernet.

- BIOS—region that takes up a variable amount of space at the end of flash memory. The BIOS contains code and configuration for the entire computer.

- PDR—Platform Descriptor Region allows system manufactures to describe custom features for the platform.

3.2.1   Flash Space Allocation

Space allocation for each region is determined as follows:

1. Each region can be assigned a fixed amount of space. If no fixed space is assigned, then the region will occupy only as much space as it requires.
2. If there is still space left in the flash after allocating space for all of the regions, the ME region will expand to fill the remaining space.
3. If there is leftover space and the BIOS region is not implemented, then the GbE region will expand to occupy the remaining space.
4. Lastly, if only the Descriptor region is implemented, it will expand to occupy the entire flash.

3.3　　　Required Files

The FITC main executable is fitc.exe. This program requires that the following files be in the same directory as fitc.exe:

- fitcmplc.xml

- newfiletmpl.xml

- vsccommn.bin

- fitc.ini

FITC will not run correctly if any of these files are missing.

3.4　　　Configuration Files

The flash image can be configured in many different ways, depending on the target hardware and the firmware options required. FITC enables the user to change this configuration in a graphical manner (via the GUI). Each configuration can be saved to an XML file. These XML files can be loaded at a later time and used to build subsequent flash images.

3.4.1　　　Creating a new configuration

The FITC provides a default configuration file from which the user can build a new image. This default configuration can be loaded by clicking **File** > **New** from the menu bar.

3.4.2　　　Opening an existing configuration

To open an existing configuration file:
1.　　　Click **File** on the menu bar.
2.　　　Select **Open**. This will cause the Open File dialog to appear.
3.　　　Select the XML file you want to load
4.　　　Click **Open**.

It is also possible to open a file by dragging and dropping a configuration file onto the main window of the application.

3.4.3   Saving a configuration

To save the current configuration in an XML file:
1.   Click **File** on the menu bar.
2.   Select **Save**.

—OR—
1.   Click **File** on the menu bar.
2.   Select **Save As...**. If **Save As...** is selected or if the configuration has not been given a name, the **Save File** dialog will appear.
3.   Select the path and file name under which to save the configuration.
4.   Click **Save**.

3.5   Environment Variables

To modify the environment variables:
1.   Click **Build** on the menu bar.
2.   Select **Environment Variables...**. A dialog box will appear showing the current working directory on top, followed by the current values of all the environment variables.

A set of environment variables is provided to make the image configuration files more portable. By making all of the paths in the configuration relative to environment variables, the configuration is not tied to a particular root directory structure. Each user can set their environment variables appropriate for their computer, or override the variables using command line options.

It is recommended that the environment variables are the first thing the user sets when working with a new configuration. This ensures that FITC can properly substitute environment variables into paths to keep them relative. Doing this also speeds up configuration because many of the Open File dialogs default to particular environment variable paths.

The variables are:

- $WorkingDir—the directory where the log file is kept. This is also where the components of an image are stored when an image is decomposed.

- $SourceDir—the directory that contains base image binary files from which a complete flash image will be prepared. Usually these base image binary files are obtained from Intel VIP on the Web, a BIOS programming resource, or other source.

- $DestDir—the directory in which the final combined image will be saved, including all intermediate files generated during the build.

- $UserVar1-3 – are used when the above variables are not populated

Figure 2. Environment Variables Dialog
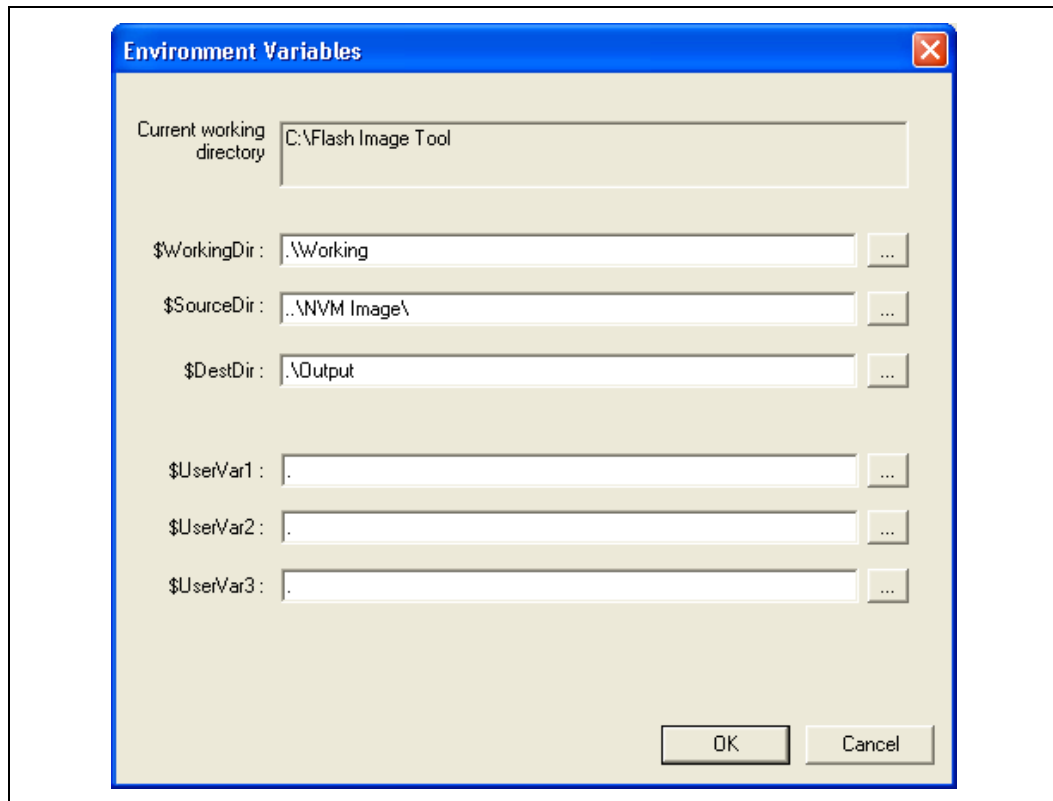
Note: The environment variables are saved in the application's INI file, not the XML configuration file. This is to allow the configuration files to be portable across different computers and directory structures.

3.6   Build Settings

To modify the build setting:
1.   Click **Build** on the menu bar.
2.   Select **Build Settings...** A dialog box will appear showing the current build settings.

FITC allows the user to set several options that control how the image is built. The Output path is the path and filename where the final image should be saved after it is built. (Use the $DestDir environment variable to make the configuration more portable.)

An option is provided (the **Generate intermediate build files** checkbox) that causes the application to generate separate (intermediate) binary files for each region, in addition to the final image file (see Figure 3). These files will be located in the int folder located inside the specified output folder. These image files can be programmed individually using the Flash Programming Tool (FPT).

The user can also elect to build a compact image which creates the smallest flash image possible. (By default, the application uses the flash component sizes in the descriptor to determine the image length.)

Finally, the user must select the flash component sector erase size. It is critical that this option is set correctly to ensure that the flash regions can be properly updated at runtime. All regions in the flash conform to the 4Kbyte sector erase size.

The **Asymmetric** option allows the user to specify a different sector erase size for the upper and lower flash block. This option also allows the user to modify the flash partition boundary address.

Figure 3. Build Settings Dialog
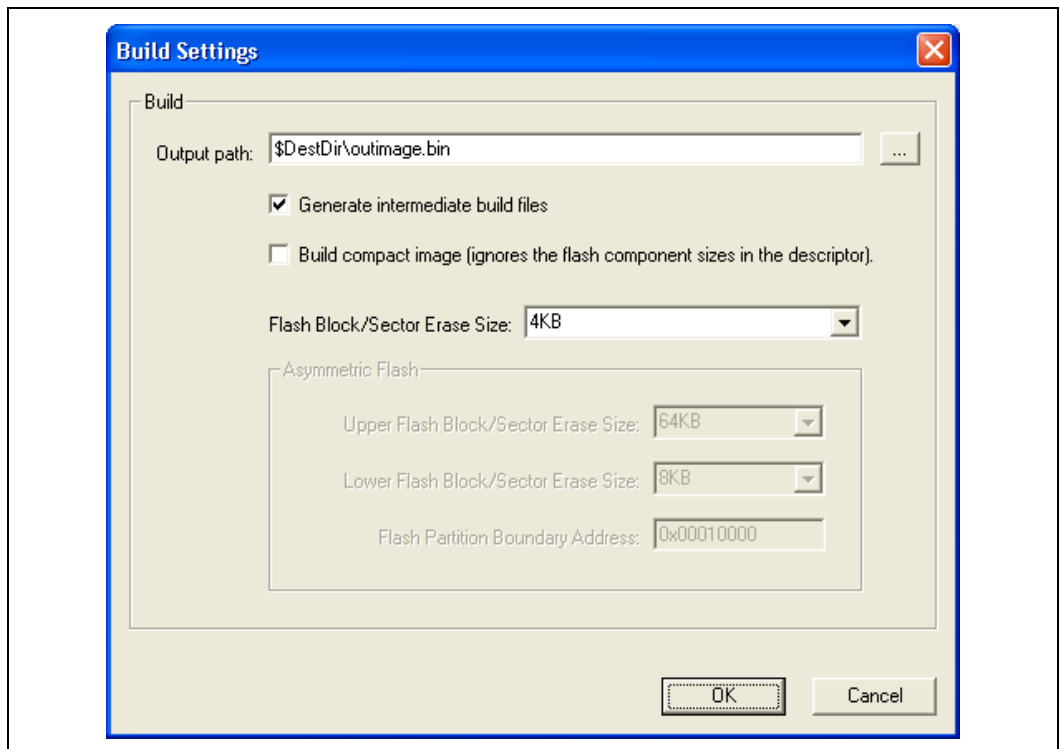
Note: The build settings are saved in the XML configuration file.

3.6.1   Selecting Platform SKU

Note: Selecting the Platform SKU needs to be done after the ME region has been loaded to ensure that the proper firmware settings are loaded into FITc.

Use the SKU Manager drop down box to select the appropriate platform type for your specific chipset.

This new feature allows testing how firmware behaves like production Intel® 5 Series Chipset using "Full Featured" Engineering samples

– Certain features only work with particular Chipset SKUs and firmware kit.
  (For example Intel® AMT only works with corporate SKUs with the 8 MB firmware kit)

– When a SKU is selected in FITc the "Full Featured Engineering Samples"will behave as if it were the selected SKU silicon.

The SKU Manager Selection option has no effect on Production Silicon
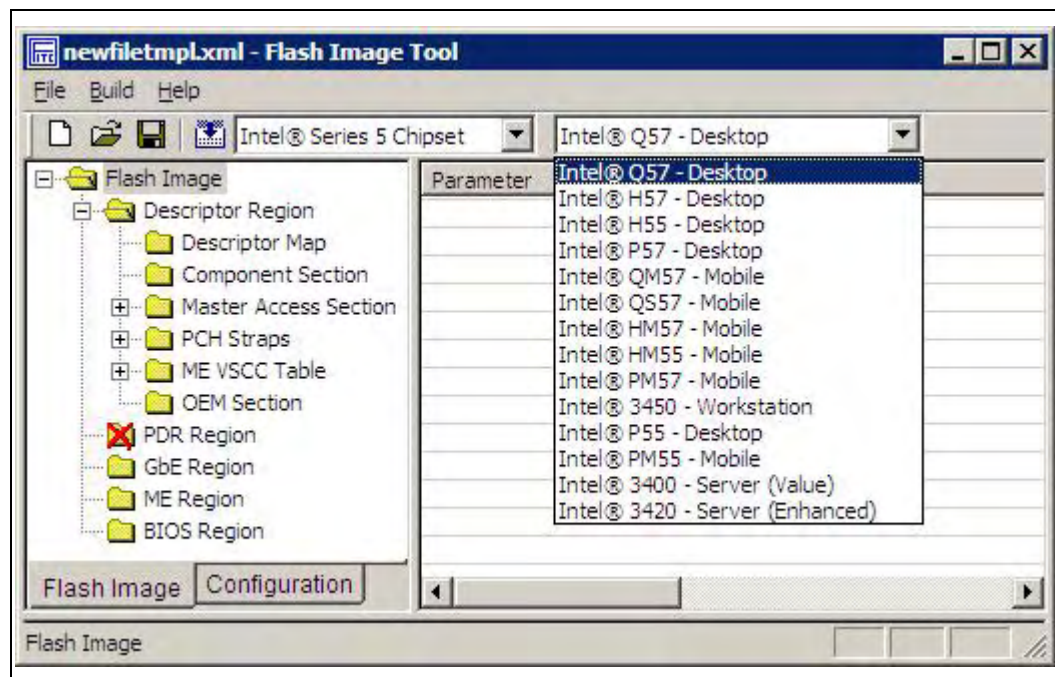
Note: The Features Supported and other Configuration tabs in FITc will show the appropriate changes to the firmware features under '**Configuration / Features Supported**' according to the SKU selected.

Note: For the 8MB firmware kit the only valid SKU choices are Intel® Q57, Intel®H57, Intel® H55, Intel® QM57, Intel® QS57, Intel® 3450, Intel® PM57, Intel® HM57 and Intel®HM55.

       Intel Confidential

3.6.1.1 Selecting Intel® RPAT (Intel® Remote PC Assist Technology) Consumer/Business Platform SKUs

When a SKU is selected in FITc the Super SKU Ibex Peak will then behaves as if it were the selected SKU silicon from Intel ME perspective.

Intel® RPAT supports Note that there are two options for Intel® RPAT which are for Consumer and Business (vPro system).

- **Intel® RPAT Business** Platform supports several SKUs, please select the appropriate platform type for your specific chipset (to be configured in the table above):
    - For Desktop– **Intel® Q57**
    - For Mobile - **Intel®QM57 Intel® QS57**
- **Intel® RPAT Consumer** Platform supports several SKUs, please select the appropriate platform type for your specific chipset (to be configured in the table above):
    - For Desktop– **Intel® H57, Intel® H55**
    - For Mobile - **Intel® HM57, Intel® PM57**

3.7 Modifying the Flash Descriptor Region (FDR)

The FDR contains information about the flash image and the target hardware. It is important for this region to be configured correctly or the target computer may not function as expected. This region contains the read/write values. This region needs to be configured correctly to ensure the system is secure.

3.7.1 Descriptor Region length

Selecting the Descriptor Region will allow the user to specify the size of the region. If a non-zero value is entered, this value will be used to determine the length of the region.

Figure 4. Descriptor Region length
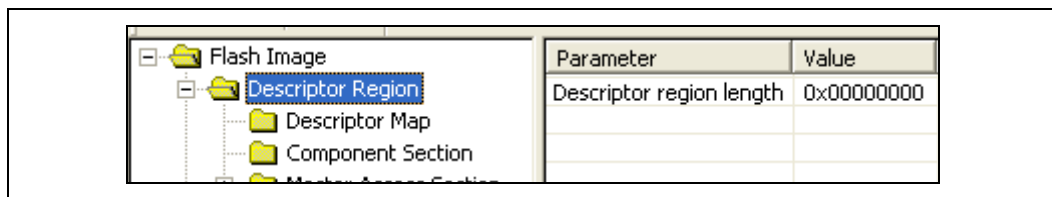
3.7.2 Setting the number and size of the flash components

To set the number of flash components:

1. Expand the Descriptor Region node of the tree in the left pane of the main window.

2. Select **Descriptor Map** (see Figure 5). All of the parameters for the Descriptor Map section will appear in the list in the right pane of the main window.

Figure 5: Editable Flash Image Region List
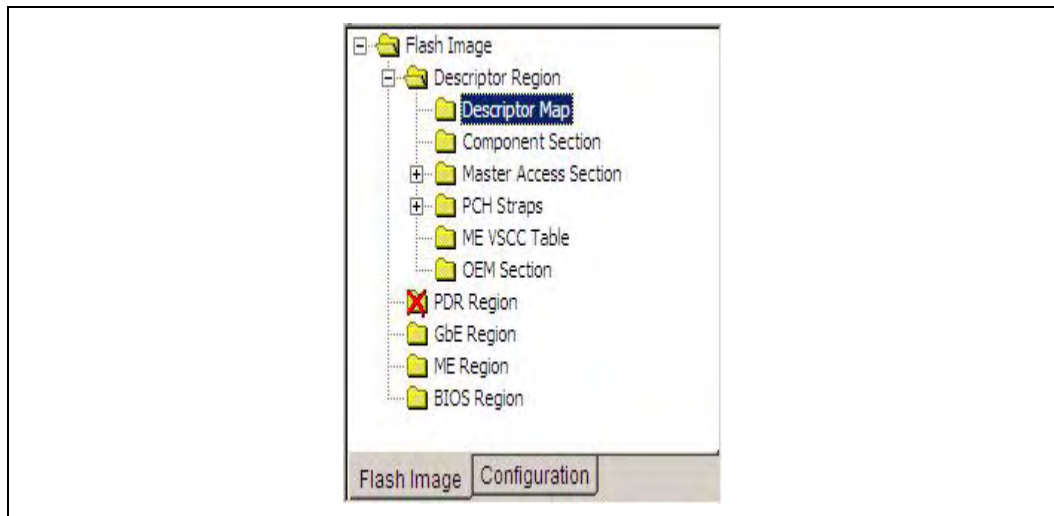
3. Double-click the list item named **Number of Flash Components** (see Figure 6). A dialog will appear allowing the user to enter the number of flash components (valid values are 1 or 2).

4. Click **OK** to update the parameter.

Figure 6: Descriptor Region Map Options
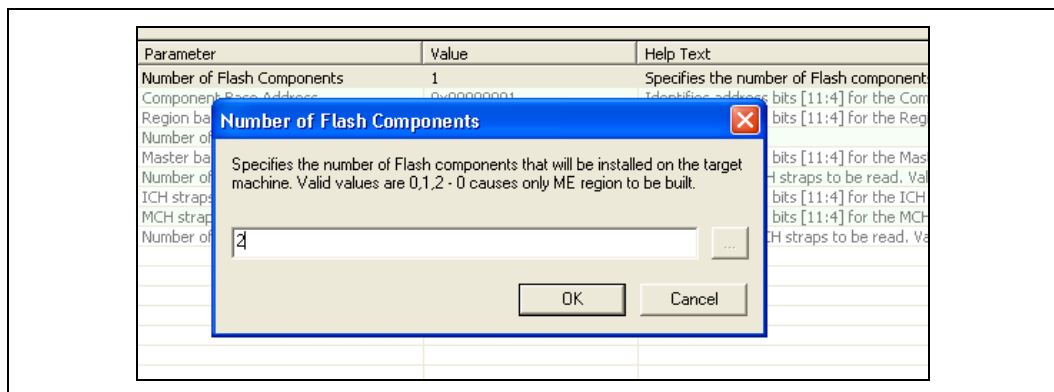
Some SPI flash devices support both standard and fast read opcodes. Fast reads are able to operate at faster frequencies than the regular reads. For PCH to support these faster read commands, fast read support must be set to true.

If the system has two SPI devices, the system will not recognize the 2nd SPI device until the 1st SPI device is programmed. For this reason, the SPI flash devices need to be programmed twice before both SPI devices are recognized. The first time the first device is programmed the image should specify two devices. The first image file should contain the Descriptor region and the BIOS Region only. This can be done using any hex editor. A future release of the flash image tool will support this feature. After the system returns from a G3 state, both SPI devices will be recognized and both will be programmable.

If the SPI devices are programmed using a flash programmer, both devices will be present after the first program

Figure 7: Descriptor Region Fast Read Support Options

Read ID and Read Status clock frequency	20MHz	If more that one Flash component exists, this field must be the
Write and erase clock frequency	20MHz	If more that one Flash component exists, this field must be the
Fast read clock frequency	33MHz	This field is undefined if the Fast Read Support is set to false.
Fast read support	true	Enables/disables "Fast Read" support.
Read clock frequency	20MHz	Sets the Flash read frequency
Flash component 1 density	512KB	This field identifies the size of the 1st Flash component.
Flash component 2 density	512KB	This field identifies the size of the 2nd Flash component.
Illegal Instruction 0	0	Op-code for an illegal instruction that the Flash Controller shou
Illegal Instruction 1	0	Op-code for an illegal instruction that the Flash Controller shou
Illegal Instruction 2	0	Op-code for an illegal instruction that the Flash Controller shou
Illegal Instruction 3	0	Op-code for an illegal instruction that the Flash Controller shou

To set the size of each flash component:

1. Expand the Descriptor Region tree node and select the **Component Section** node. The parameters Flash component 1 density and Flash component 2 density specify the size of each flash component.
2. Double-click on each parameter and select the correct component size from the drop-down list.
3. Click **OK** to update the parameters.

Note: The size of the second flash component will only be editable if the number of flash components is set to 2.

Figure 8: Descriptor Region Component Section Options

Read ID and Read Status clock frequ...	20MHz	If more that one Flash component exists, this f
Write and erase clock frequency	20MHz	If more that one Flash component exists, this f
Fast read clock frequency	33MHz	This field is undefined if the Fast Read Support
Fast read support	true	Enables/disables "Fast Read" support.
Read clock frequency	20MHz	Sets the Flash read frequency
Flash component 1 density	512KB	This field identifies the size of the 1st Flash com
Flash component 2 density	512KB	This field identifies the size of the 2nd Flash com
Illegal Instruction 0	0	Op-code for an illegal instruction that the Flash
Illegal Instruction 1	0	Op-code for an illegal instruction that the Flash
Illegal Instruction 2	0	Op-code for an illegal instruction that the Flash
Illegal Instruction 3	0	Op-code for an illegal instruction that the Flash

3.7.3   Region access control

Regions of the flash can be protected from read or write access by setting a protection parameter in the Descriptor Region. Before ME devices are shipped, the Descriptor Region must be locked. If the Descriptor Region is not locked, the ME device is vulnerable to security attacks. The level of read/write access provided is at the discretion of the OEM/ODM. A cross-reference of access settings is shown below.

Table 3. Region Access Control Table

Regions that can be accessed

Region to Grant Access	PDR	ME	GBE	BIOS	Descriptor
ME	None / Read / Write	None / Read / Write	Write only. ME can always read from and write to ME Region	None / Read / Write	None / Read / Write
GBE	None / Read / Write	Write only. GBE always read from and write to GBE Region	None / Read / Write	None / Read / Write	None / Read / Write
BIOS	None / Read / Write	None / Read / Write	None / Read / Write	Write only. BIOS can always read from and write to BIOS Region	None / Read / Write

Three parameters in the Descriptor exist to specify access for each chipset. The bit structure of these parameters are shown below.

Key:

0—denied access

1—allowed access

NC—bit may be either 0 or 1 since it is unused.

CPU /BIOS gets...

Read Access

	Unused			PDR	GbE	ME	BIOS	Desc
Bit Number	7	6	5	4	3	2	1	0
Bit Value	X	X	X	0/1	0/1	0/1	NC	0/1

Write Access

	Unused			PDR	GbE	ME	BIOS	Desc
Bit Number	7	6	5	4	3	2	1	0
Bit Value	X	X	X	0/1	0/1	0/1	NC	0/1

For example, if the CPU/BIOS needs read access to the GbE and ME and write access to ME, then the bits will be set to:

Read Access—0b 0000 1110

Write Access—0b 0000 0110

In hexadecimal:

Read Access—0x 0E

Write Access—0x 06

In FITC these access values can be set by selecting the Descriptor Region tree node and selecting CPU/BIOS under the Master Access, Manageability Engine and GBE section (see Figure 9).

Figure 9: Descriptor Region Master Access Section Location
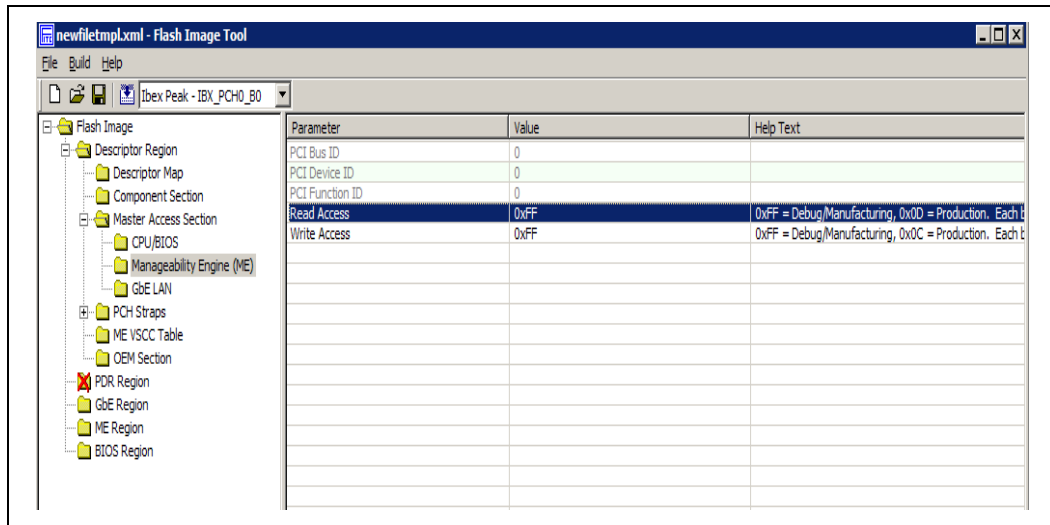
When you want to generate image for debug purpose or want to leave the SPI region open, you can select 0xFF for both read and write access in all three section. When you wan to lock the SPI in image creation phase, please select recommended setting for production. For example, you should select 0x0D for read access of ME and 0x0C for write access of ME

3.8 PCH Soft straps

These sections contain configuration options for the PCH. The number of Soft Strap sections and their functionality differ based on the target PCH. Improper settings could lead undesirable behavior from the target platform. More details on how to set them correctly please refer to FW Bringup Guide or PCH SPI programming guide Appendix A –, for more detail.

Figure 10: configuration tab
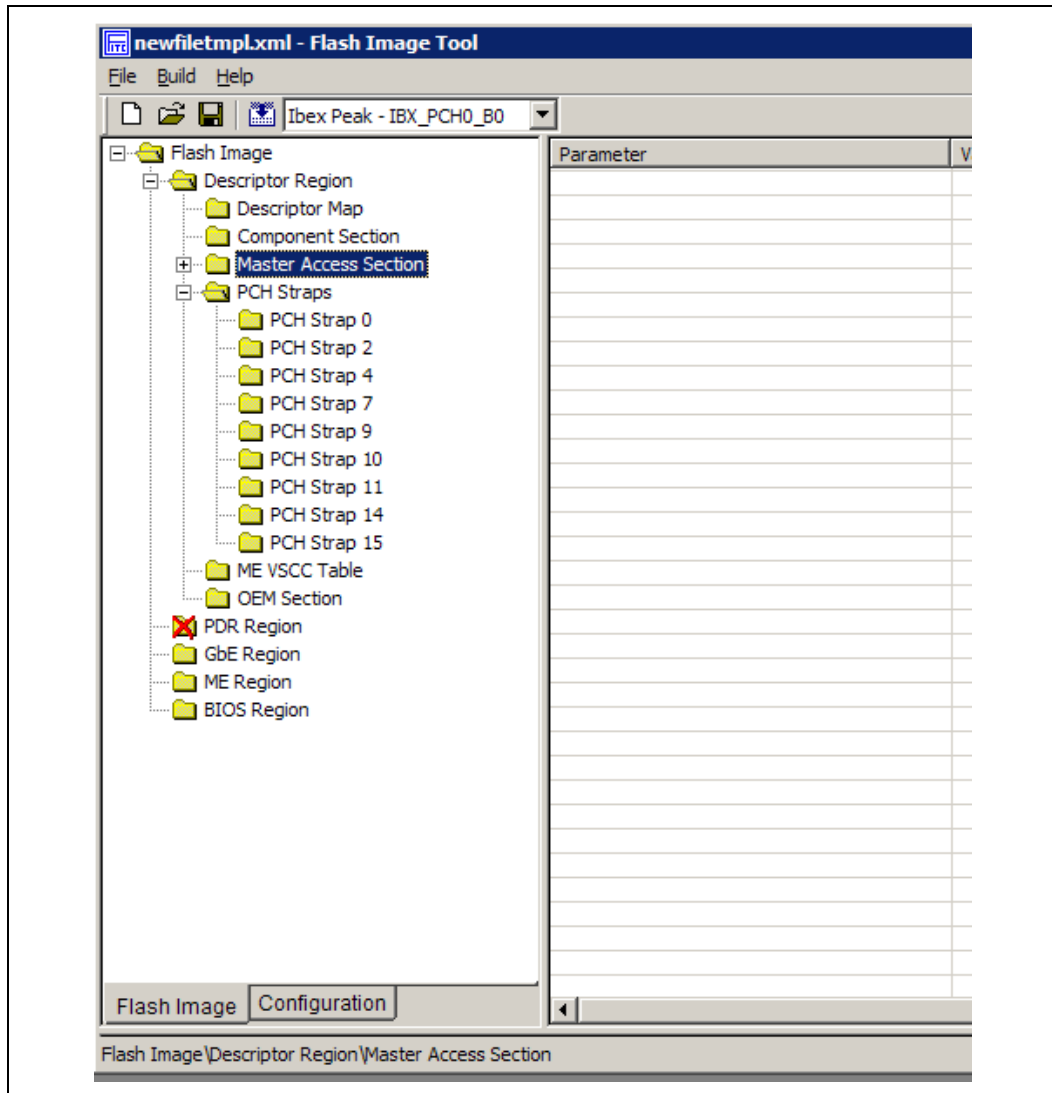
3.9   VSCC Table

This section is used to store information to setup flash access for ME. This does not have any effect on the usage of the Flash programming Tool (FPT). **If the information in this section is incorrect, the Intel® ME Firmware may not communicate with the flash device.** This information provided is dependent on the flash device used on the system. More detail please refer to PCH SPI programming guide Section 6.4

3.9.1   Adding a new table

To add a new table:

1.   Right-click on **VSCC table**.
2.   Select **Add Table Entry...**

Figure 11: Add New VSCC table entry
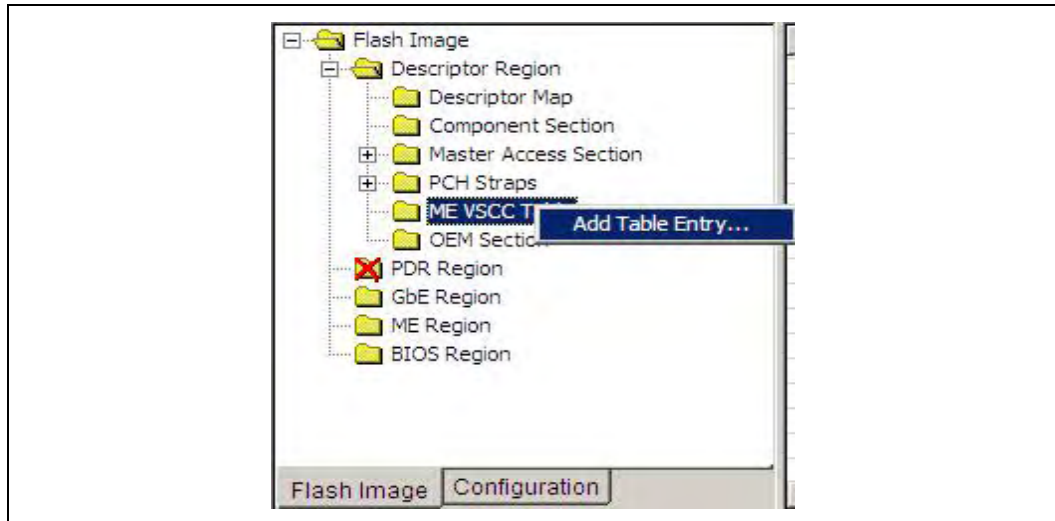
The program will then prompt the user for a table entry name. To avoid confusion it is recommended that each table entry be unique. There is no checking mechanism in FITC to prevent table entries that have the same name and no error message will be displayed in such cases.

Figure 12: Add VSCC table entry
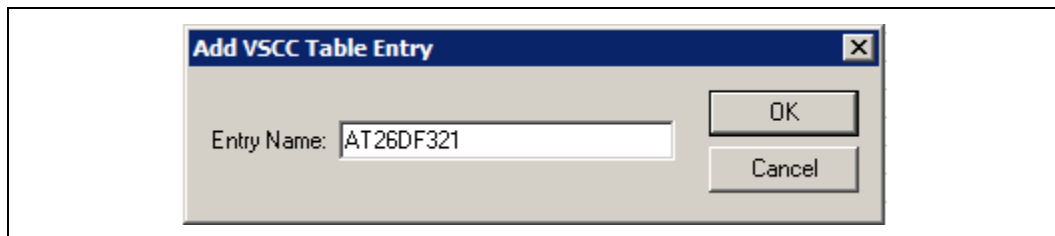
After a table entry has been added, the user will be able to fill in values for the flash device. The values in the VSCC table are provided by your flash vendor. Users should contact their flash vendor for the specific values mentioned in this table.

The screenshot below shows the values for the flash part SST25vf016b.

Figure 13: VSCC Table Entry

Parameter	Value	Help Text
Vendor ID	0x1F	The vendor specific byte of the JEDEC ID.
Device ID 0	0x47	The first device specific byte of the JEDEC ID.
Device ID 1	0x00	The second device specific byte of the JEDEC ID.
VSCC register value	0x20152015	This entry will only add SPI flash support for Intel? Management Engine
Right-Click folder to delete this table entry		To delete this VSCC table entry right-click the folder.

3.9.2   Removing an existing table

To remove an existing table:

1.   Right-click on the table that needs to be removed.
2.   Select **Remove Table Entry**. All information in the table along with the table entry will be removed.

Figure 14: Remove VSCC table entry
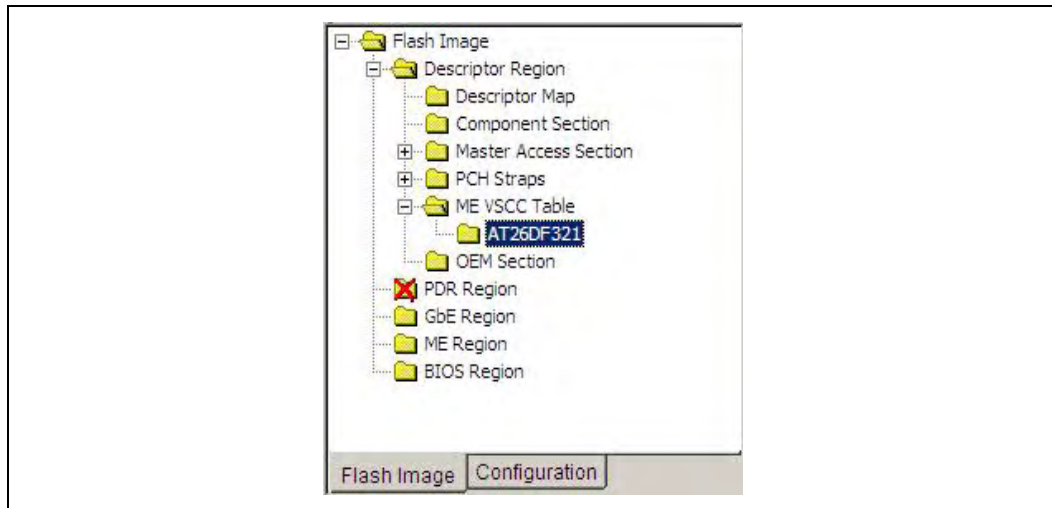
3.10   Modifying the ME Region

The ME Region contains all of the firmware and data for the Intel® ME (which includes the kernel and Intel® AMT).

3.10.1   Setting the ME Region binary file

To set the ME region binary file:
1.   Select the ME Region tree node.
2.   Double-click on the **Binary file parameter** in the list. A dialog box will appear allowing the user to specify the ME file to use.
3.   Click **OK** to update the parameter.

When the flash image is built, the contents of this file will be copied into the ME Region.

The ME Region length option should not be altered. A value of 0x00000000 indicates that the ME Region will be auto-sized as described in Section 4.2, Flash space allocation.

If the user has specified in the PCH Strap 0 Section 3.8 that the ME must boot from flash, the firmware loaded must contain a ROM Bypass section. If the firmware does not contain a ROM bypass section, a section will become available in which to enter the location of the ROM bypass file.

3.10.2   Enabling/disabling the ME Region

The ME Region can be excluded from the flash image by disabling it in FITC.

To disable the ME Region:
1.   Right-click on the ME Region tree node.
2.   Select **Disable Region** from the pop-up menu.

The user will then need to increase the size in one of the other regions. FITC will "pad" the remaining space. For example, if the user wants to disable the ME Region and "pad" the GbE Region he would subtract the size of the BIOS Region, PDR Region (if a PDR Region is included) and the Descriptor Region from the full SPI image size. This will determine the new size of the GbE Region.

3.10.2.1 Example 1

The example below assumes a symmetric 4kb flash with a 1 KB BIOS with no PDR Region.

Full SPI Image size – BIOS Region size – Descriptor Region size = GBE Region Size

0x400000   –   0x100000   –   0x1000   =   0x2ff000

The GbE Region size value should be entered for the GbE LAN Region length in the GbE section. "Padding" the BIOS Region is not recommended.

The ME Region can be enabled by right-clicking on the ME Region tree node and selecting **Enable Region** from the pop-up menu.

Figure 15: Enabling the ME Region
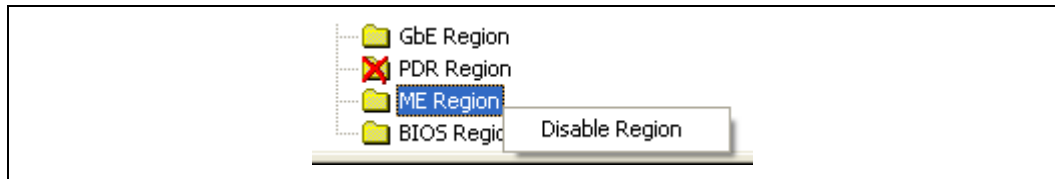
3.11   Modifying the GbE (LAN) Region

The GbE Region contains various configuration parameters (such as, the MAC address) for the embedded Ethernet controller.

3.11.1   Setting the GbE Region binary file

To set the GbE Region binary file:
1.   Select the GbE Region tree node.
2.   Double-click on the **Binary input file parameter** from the list. A dialog box will appear allowing the user to specify which GbE file to use. Select a file.
3.   Click **OK** to update the parameter.

When the flash image is built, the contents of this file will be copied into the GbE Region.

The GbE Region length option should not be altered. A value of 0x00000000 indicates that the GbE Region will be auto-sized as described in Section 3.2.1.

Figure 16: GbE Region Options

This is the location where the user can modify the Ethernet MAC address.

To configure the Ethernet MAC address:
1.   Double-click the MAC address parameter from the list. A dialog box will appear allowing the user to specify the Ethernet MAC address.
2.   Enter the required value.
3.   Click **OK** to update the parameter.

3.11.2   Enabling/disabling the GbE Region

The GbE Region can be excluded from the flash image by disabling it in the FITC.

To disable the GbE Region:
1.   Right-click on the GbE Region tree node.
2.   Select **Disable Region** from the pop-up menu. When the flash image is built it will not contain a GbE Region.

Figure 17: Disabling the GbE Region
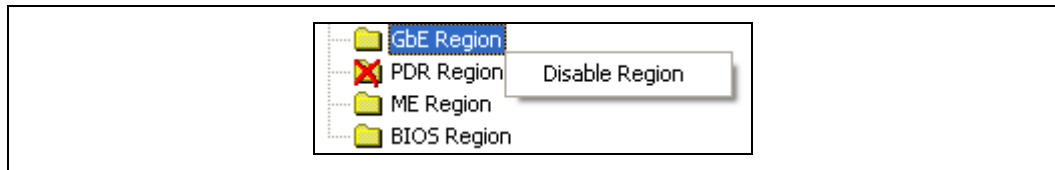
To enable the GbE Region:
1.   Right-click on the GbE Region tree node.
2.   Select **Enable Region** from the pop-up menu.

3.12   Modifying the PDR Region

The PDR Region contains various configuration parameters that allow for the customization of the computer's behavior.

3.12.1 Setting the PDR Region binary file

To set the PDR region binary file:

1. Select the PDR Region tree node.
2. Double-click the **Binary input file parameter** from the list. A dialog box will appear allowing the user to specify the PDR file to use.
3. Click **OK** to update the parameter. When the flash image is built, the contents of this file will be copied into the BIOS region.

The PDR Region length option should not be altered. A value of 0x00000000 indicates that the PDR Region will be auto-sized as described in Section 3.2.1.

Figure 18: PDR Region Options

3.12.2 Enabling/disabling the PDR Region

The PDR Region can be excluded from the flash image by disabling it in FITC.

To disable the PDR Region:

1. Right-click on the PDR Region tree node.
2. Select **Disable Region** from the pop-up menu. When the flash image is built, there will be no PDR Region in it.

By default this region is disabled.

Figure 19: Disabling the PDR Region
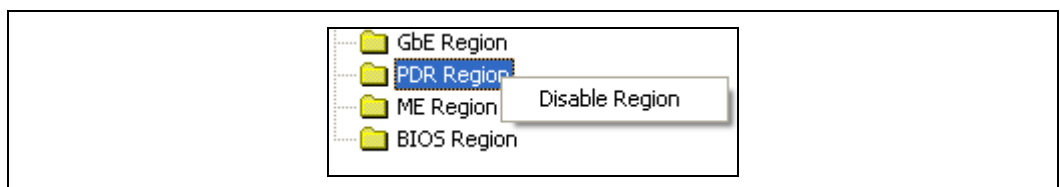
To enable the PDR Region:

1. Right-click on the PDR Region tree node.
2. Select **Enable Region** from the pop-up menu.

3.13   Modifying the BIOS Region

The BIOS Region contains the BIOS code run by the host processor. FITC always aligns this region with the end of the flash image. This is done so that in the event that the flash descriptor becomes corrupt for any reason, the PCH will default to legacy mode and look for the reset at the end of the flash memory. By placing the BIOS Region at the end there is a chance the system will still boot. It is also important to note that the BIOS binary file will be aligned with the end of the BIOS Region so that the reset vector is in the correct place. This means that if the binary file is smaller than the BIOS Region, the region will be padded at the beginning instead of at the end.

3.13.1   Setting the BIOS Region binary file

Figure 20: BIOS Region Options

BIOS region length	0x00000000	This is the size of the BIOS region in bytes. Set this to 0 to make the region le...
Binary input file		This is the BIOS image binary that will be copied into this region.

To set the BIOS region binary file:
1.   Select the BIOS Region tree node.
2.   Double-click the **Binary input file parameter** from the list. A dialog box will appear allowing the user to specify the BIOS file to use.
3.   Click **OK** to update the parameter. When the flash image is built, the contents of this file will be copied into the BIOS region.

The BIOS Region length option should not be altered. A value of 0x00000000 indicates that the BIOS Region will be auto-sized as described in Section 3.2.1.

3.13.2   Enabling/disabling the BIOS Region

The BIOS Region can be excluded from the flash image by disabling it in FITC.

To disable the BIOS Region:
1.   Right-click on the BIOS Region tree node.
2.   Select **Disable Region** from the pop-up menu. When the flash image is built, there will be no BIOS Region in it.

Flash Image Tool (FITC)

Figure 21: Disabling the BIOS Region
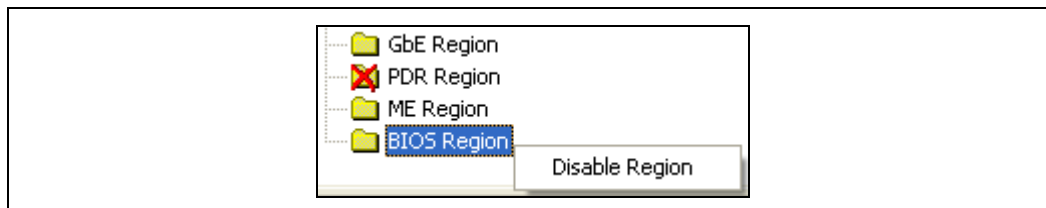
To enable the PDR Region:
1.   Right-click on the BIOS Region tree node.
2.   Select **Enable Region** from the pop-up menu.

3.14   Configuration Tab

The Configuration tab located at the bottom of the window allows the user to set specific parameters.

If any of the parameters are changed from the Intel recommended value the offending row will be highlighted yellow. No errors will be reported. The highlighted yellow is designed to draw attention to these values were ensure these parameters and set correctly

Figure 22: Configuration Tab
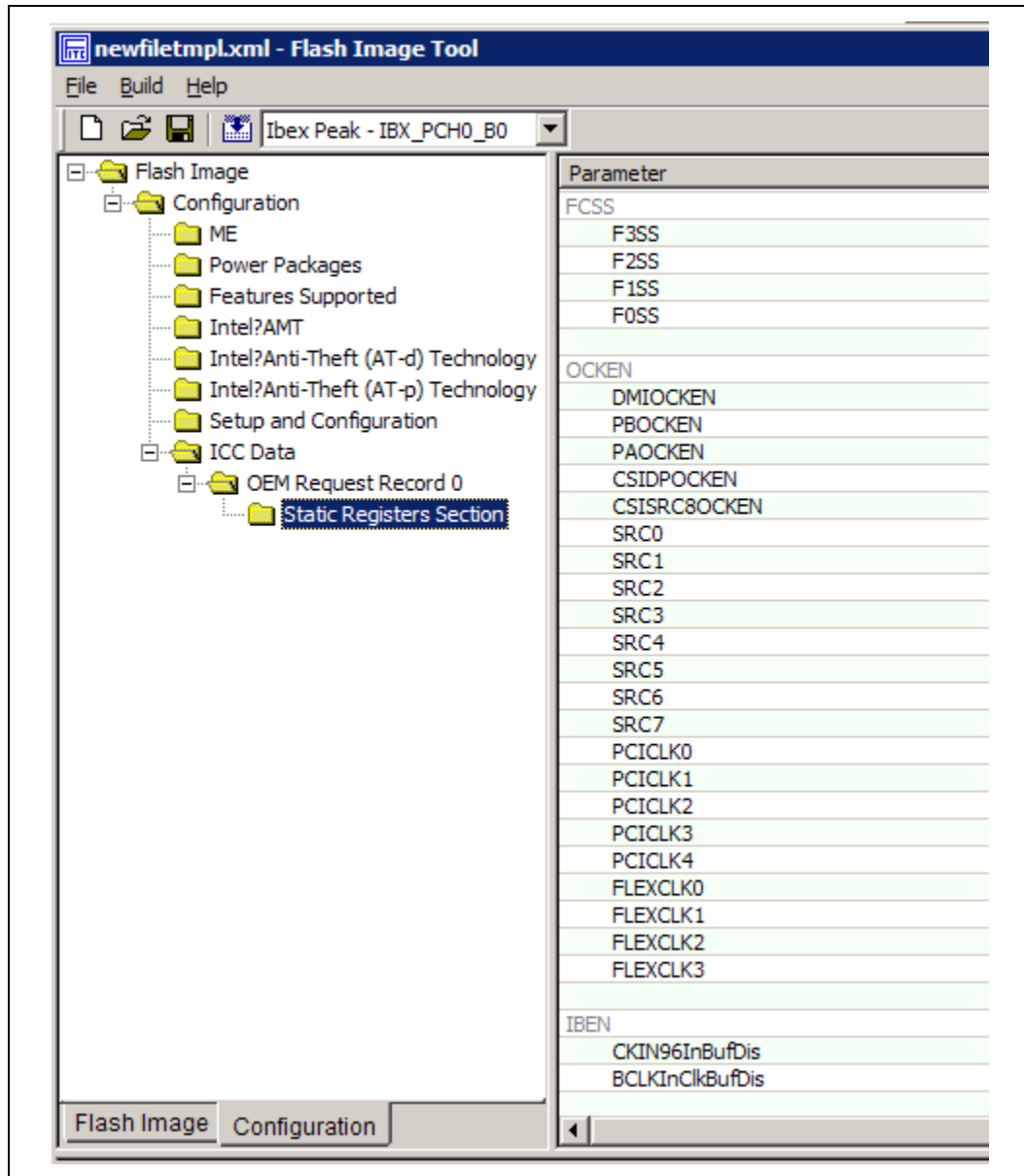
3.14.1   ME Section

The ME section allows the user to specify the computer's manageability features. The parameters values are can be found in the Help Text alongside to the parameter value as shown in Figure 23.

Figure 23: ME Section

Parameter	Value
Local FWU Override Counter	0
Local FWU Override Qualifier	0
FW Update OEM ID	00000000-0000-0000-0000-000000000000
ME State on Flash Desc OVR	false
BIOS Reflash Capable	false
LAN Power Well Config	3
WLAN Power Well Config	0x80
M3 Power Rails Availability	true
HECI ME Region Unlockable	true
Sub System Vendor ID	0x0000
Debug Si Features	0x00000000
Prod Si Features	0x00000000

M3 Power Rails Available: This value will determine if M3 functionality will be available for firmware. For the Desktop and Mobile CRB platforms this value needs to be set to '**true'**.

Note: M3 Power Rail availability depends on the specific platform design and needs to be set appropriately.

For platforms with M3 support the value needs to be set **'true'** for proper firmware operation.

For platforms without M3 support this value needs to be set to **'false'** for proper firmware operation.

FW Update OEM ID: This UUID will make sure that customers can only update a platform with an image coming from the platform OEM. If set to an all zero value then any input is valid when doing a firmware update

3.14.1.1 Flash Descriptor Override Pin Strap Ignore

This bit determines if ME will be disabled when the Flash descriptor override jumper (GPIO 33) set.

False – ME will enters a disabled state to safely program the full SPI device if the manufacturing mode jumper is set

True – ME will NOT enter a disabled if the Flash descriptor override jumper (GPIO 33) is set.

3.14.1.2 Temporary firmware update parameters

If the Local FWU Override Counter has a value between 1 and 255, firmware updates are allowed even if updates are disabled in the ME BIOS Extension (MEBx) settings. After the flash is programmed, each time the computer restarts it causes the Local FWU Override Counter to be decremented. When the Local FWU Override Counter reaches 0, firmware updates are no longer allowed if they are not enabled in the MEBx settings.

Note: The restart that takes place after the flash memory has been programmed also causes the Local FWU Override Counter to be decremented. Therefore, if it is necessary to enable updating the firmware N times, you need to assign the Local Firmware Update Override Counter the initial value N+1.

If the Local FWU Override Counter is set to -1 and the Local Firmware Override Qualifier is set to 0, firmware updates are always allowed regardless of the settings in the MEBx.

The following table shows the possible value combinations for the two variables. To enable local firmware updates, make sure both variables are assigned the correct values.

Table 4. Firmware Override Update Variables

	Local FWU Override Qualifier = 0 (zero)	Local FWU Override Qualifier = 1 (one)	Local FWU Override Qualifier = 2 (two)
Local FWU Override counter = 0 (zero)	Local Firmware Updates NOT Allowed	Local Firmware Updates NOT Allowed	Local Firmware Updates NOT Allowed
Local FWU Override Counter = -1 (minus one)	Local Firmware Updates Allowed	Local Firmware Updates NOT Allowed	Local Firmware Updates Allowed only until ME is configured
Local FWU Override Counter = 0<n<255	Local Firmware Updates Allowed	Local Firmware Updates Allowed	Local Firmware Updates Allowed

3.14.1.3 Debug Si Features/ Prod Si Features

This option will output the firmware status register to a specified bus address. This is a debug feature that should not be enabled in the production image.

The D*ebug Si feature* parameter should be used for non-production hardware. The P*rod Si feature* parameter should be used for production hardware.
Both of these parameters should not be set in the final production image.

1 – Firmware status register to the default bus address of the MDDD device inserted in Channel 0 memory slot

Bus Address – Firmware status register will output to the specified bus address. Bits 7:1 are used for the Bus Address. Bit 0 is used for the enable bit with 1 denoting enabled and 0 is disabled.

3.14.2   AMT Section

The AMT section allows the user to specify the default AMT parameters. The values specified in this section will be used after the Intel® AMT device is un-provisioned (full or partial).

Figure 24: AMT Section

Parameter	Value
Intel? AMT Ping Response Enabled	true
VLAN	0
Boot into BIOS Setup Capable	false
Pause during BIOS Boot Capable	false
HostIf IDER Enabled	true
HostIf SOL Enabled	true
Idle Timeout - Manageability Engine	1
Full Test Counter	8
KVM Host I/F Enabled	11b Enabled
KVM Opt-In PTNI Editable Policy	11b Enabled
KVM Opt-In Enabled Policy	11b Enabled
USBr EHCI 1 Enabled	11b Enabled
USBr EHCI 2 Enabled	00b Not Set

Be careful when setting these parameters as some of them cannot be modified by the end user, such as the Boot into BIOS setup Capable.

HostIF IDER Enabled – Determines if IDE-R sessions are permitted. Before an IDE-R session can be opened, this option must be set to true. This parameter can be modified in the MEBX
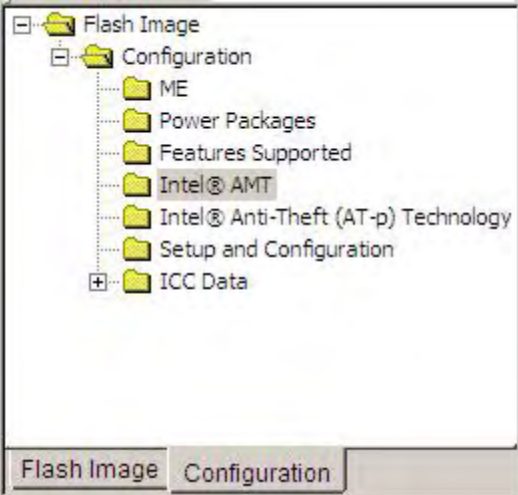
HostIF SOL Enabled – Determines if SOL sessions are permitted. Before an SOL session can be opened, this option must be set to true. This parameter can be modified in the MEBX.

Idle Timeout -Specifies the amount of time (<u>in minutes</u>) before the system goes into an M-off state if ME-WOL is enabled. This value can be modified by the end user in the MEBx. To reduce the amount of end user configuration time, this value should be set to a reasonable value.

3.14.2.1   Intel® RPAT Consumer/Business configuration under Intel AMT tab

In order to have Intel® RPAT both business/Consumer enabled settings must be configure as below

Location			
	Parameter	**Default**	**Comments**
	Intel® AMT Ping Response Enabled	true	
	Boot into BIOS Setup Capable	true	
	Pause during BIOS Boot Capable	true	
	HostIf IDER Enabled	true	
	HostIf SOL Enabled	true	

3.14.3   Power Packages Section

The Power Packages section allows the OEM/ODM to specify which power packages are supported.

Figure 25: Power Packages Section

If the Power Package Supported value is set to false, that specific power package cannot be selected and will not be visible to the end user.

The Default Power Package selected must be supported. This is the value that will be selected when the system is shipped. This value will affect energy star compliance if not set correctly.

3.14.3.1   Power Packages for Intel® RPAT Consumer/Business

- Intel® RPAT Consumer Platform supports the below configuration (only S0 support in RPAT consumer SKU's both Desktop and Mobile).

Figure 26: Power Packages for Intel® RPAT Consumer

- Intel® RPAT Business Platform supports the below configuration (configures ME for operation in S0 and ME Wake in S3, S4 and S5).

Figure 27: Power Packages for Intel® RPAT Business

3.14.4　　Features Supported

The Features supported section will determine which features are supported by the system. If a system does not meet the minimum hardware requirements, no error message will be given when programming the image

Figure 28: Features Supported Section

Parameter	Value
Enable Intel?Standard Manageability; Disable Intel?AMT	No
Manageability Application Permanently Disabled?	No
PAVP 1.5 Permanently Disabled?	No
Intel?QST Permanently Disabled?	No
Intel?Identity Protection Technology Permanently Disabled?	Yes
Intel?Remote Wake Technology Permanently Disabled?	Yes
KVM Permanently Disabled?	No
Braidwood Technology Permanently Disabled?	No
TLS Permanently Disabled?	No
Manageability Application Enable/Disable	Enabled
PAVP 1.5 Enable/Disable	Enabled
Intel?QST Enable/Disable	Enabled
Intel?Identity Protection Technology Enable/Disable	Disabled
Intel?Remote Wake Technology Enable/Disable	Disabled

These options control the availability / visibility of firmware features.

In instances where a specific feature is configurable in the MEBx disabling it through the 'Features Supported' section will hide / disable that specific feature in the MEBx.

The ability to change certain options is SKU dependent and some of default values will be grayed out and will not be changeable depending on the SKU Selected.

Note:

The Intel® Manageability Application setting combines several manageability technologies that are related to each other. This setting controls the following manageability technologies:

Intel® Active Management Technology
Intel® Standard Management
Intel® Remote PC Asset Technology for Consumer
Intel® Remote PC Asset Technology for Business
Fast Call for Help
Intel® KVM Remote Assistance Application

Setting "Intel® Manageability Application Permanently Disabled?" to "Yes" will permanently disable all the features listed above without any way to enable them at a later time. The only way to re-enable these features is to completely re-burn the ME region with this setting value set to "No." A firmware update using ***FWUpdLcl.exe*** cannot re-enable features.

All parameters in this section are color-coded as per the key below.

The parameter can be changed.
The parameter is read only and cannot be changed.

Table 5. Feature default settings by SKU

SKU	Feature	Default Value
Intel® Q57	Enable Intel® Standard Manageability; Disable Intel® AMT	No
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	No
	Intel® Identity Protection Technology Permanently Disabled?	Yes
	Intel® Remote Wake Technology Permanently Disabled?	Yes
	KVM Permanently Disabled?	No
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Enabled
	Intel® Identity Protection Technology Enable / Disable	Disabled
	Intel® Remote Wake Technology Enable / Disable	Disabled
Intel® H57	Enable Intel® Standard Manageability; Disable Intel® AMT	Yes
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	No
	Intel® Identity Protection Technology Permanently Disabled?	No
	Intel® Remote Wake Technology Permanently Disabled?	No
	KVM Permanently Disabled?	No
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Enabled
	Intel® Identity Protection Technology Intel® Identity Protection TechnologyEnable / Disable	Enabled
	Intel® Remote Wake Technology Enable / Disable	Enabled
Intel® H55	Enable Intel® Standard Manageability;	Yes

SKU	Feature	Default Value
	Disable Intel® AMT	
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	No
	Intel® Identity Protection Technology Permanently Disabled?	No
	Intel® Remote Wake Technology Permanently Disabled?	No
	KVM Permanently Disabled?	Yes
	Braidwood Technology Permanently Disabled?	Yes
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Enabled
	Intel® Identity Protection Technology Enable / Disable	Enabled
	Intel® Remote Wake Technology Enable / Disable	Enabled
Intel® QM57	Enable Intel® Standard Manageability; Disable Intel® AMT	No
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	Yes
	Intel® Identity Protection Technology Permanently Disabled?	Yes
	Intel® Remote Wake Technology Permanently Disabled?	Yes
	KVM Permanently Disabled?	No
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Disabled
	Intel® Identity Protection Technology Enable / Disable	Disabled
	Intel® Remote Wake Technology Enable / Disable	Disabled
Intel® QS57	Enable Intel® Standard Manageability; Disable Intel® AMT	Yes
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	Yes
	Intel® Identity Protection Technology Permanently Disabled?	Yes
	Intel® Remote Wake Technology Permanently	Yes

Intel Confidential

SKU	Feature	Default Value
	Disabled?	
	KVM Permanently Disabled?	No
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Disabled
	Intel® Identity Protection Technology Enable / Disable	Disabled
	Intel® Remote Wake Technology Enable / Disable	Disabled
Intel® HM57	Enable Intel® Standard Manageability; Disable Intel® AMT	Yes
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	Yes
	Intel® Identity Protection Technology Permanently Disabled?	No
	Intel® Remote Wake Technology Permanently Disabled?	Yes
	KVM Permanently Disabled?	No
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Disabled
	Intel® Identity Protection Technology Enable / Disable	Enabled
	Intel® Remote Wake Technology Enable / Disable	Disabled
Intel® HM55	Enable Intel® Standard Manageability; Disable Intel® AMT	Yes
	Intel® Manageability Application Permanently Disabled?	Yes
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	Yes
	Intel® Identity Protection Technology Permanently Disabled?	Yes
	Intel® Remote Wake Technology Permanently Disabled?	Yes
	KVM Permanently Disabled?	Yes
	Braidwood Technology Permanently Disabled?	Yes
	TLS Permanently Disabled?	Yes
	Intel® Manageability Application Enable / Disable	Disabled
	PAVP 1.5 Enable / Disable	Enabled

SKU	Feature	Default Value
	Intel® QST Enable / Disable	Disabled
	Intel® Identity Protection Technology Enable / Disable	Disabled
	Intel® Remote Wake Technology Enable / Disable	Disabled
Intel® PM57	Enable Intel® Standard Manageability; Disable Intel® AMT	Yes
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	Yes
	Intel® QST Permanently Disabled?	Yes
	Intel® Identity Protection Technology Permanently Disabled?	No
	Intel® Remote Wake Technology Permanently Disabled?	Yes
	KVM Permanently Disabled?	Yes
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Disabled
	Intel® QST Enable / Disable	Disabled
	Intel® Identity Protection Technology Enable / Disable	Enabled
	Intel® Remote Wake Technology Enable / Disable	Disabled
Intel® 3450	Enable Intel® Standard Manageability; Disable Intel® AMT	Yes
	Intel® Manageability Application Permanently Disabled?	No
	PAVP 1.5 Permanently Disabled?	No
	Intel® QST Permanently Disabled?	No
	Intel® Identity Protection Technology Permanently Disabled?	Yes
	Intel® Remote Wake Technology Permanently Disabled?	Yes
	KVM Permanently Disabled?	No
	Braidwood Technology Permanently Disabled?	No
	TLS Permanently Disabled?	No
	Intel® Manageability Application Enable / Disable	Enabled
	PAVP 1.5 Enable / Disable	Enabled
	Intel® QST Enable / Disable	Enabled
	Intel® Identity Protection Technology Enable / Disable	Disabled
	Intel® Remote Wake Technology Enable / Disable	Disabled

3.14.4.1.1 Feature supported for Intel® RPAT Consumer/Business

For both Intel® RPAT Consumer/Business configurations:

- "Manageability Application Permanently disabled?" needs to be set to -> No. (see below).

- "Manageability Application Disable/Enable?" needs to be set to -> Enable. (see below).

Figure 29: Features Supported Intel® RPAT Section

Enable Intel® Standard Manageability; Disable Intel® AMT	Yes	Only applicable for Intel® Q57.
Manageability Application Permanently Disabled?	No	Setting this to Yes permanently disables Intel® AMT, Intel® Standard Manag...
PAVP 1.5 Permanently Disabled?	No	Select whether Protected Audio Video Path (PAVP) 1.5 is permanently disabled.
Intel® QST Permanently Disabled?	No	Select whether Intel® Quiet System Technology is permanently disabled.
Intel® Identity Protection Technology Permanently Disabled?	No	Select whether Sentry Peak is permanently disabled.
Intel® Remote Wake Technology Permanently Disabled?	No	Select whether Intel® Remote Wake Technology is permanently disabled.
KVM Permanently Disabled?	No	Select whether KVM is permanently disabled.
Braidwood Technology Permanently Disabled?	Yes	Select whether Braidwood Technology is permanently disabled.
TLS Permanently Disabled?	No	Select whether TLS is permanently disabled.
Manageability Application Enable/Disable	Enabled	Select whether or not Intel® Active Management Technology is enabled or dis...
PAVP 1.5 Enable/Disable	Enabled	Select whether or not the Protected Audio Video Path 1.5 is enabled or disabled.
Intel® QST Enable/Disable	Enabled	Select whether or not Intel® Quiet System Technology is enabled or disabled.
Intel® Identity Protection Technology Enable/Disable	Enabled	Select whether Sentry Peak is enabled or disabled.
Intel® Remote Wake Technology Enable/Disable	Enabled	Select whether or not Intel® Remote Wake Technology is enabled or disabled.

3.14.5 Setup and Configuration Section

The Setup and Configuration section allows the end user to specify the configuration settings. These values determine the mode of the Intel® AMT device after the system has been configured.

Figure 30: Setup and Configuration Section

Parameter	Value
ODM ID used by Intel?Upgrade Service	0x00000000
System Integrator ID used by Intel?Upgrade Service	0x00000000
Reserved ID used by Intel?Upgrade Service	0x00000000
MEBx Password Policy	0
Provisioning Time Period	0
Remote Configuration Enabled	true
PKI DNS Suffix	
Remote Connectivity Service Capability	true
Remote Connectivity Service Enabler Id	00000000-0000-0000-0000-000000000000
Remote Connectivity Service Enabler Name	
RCS HW Button	0x01
Hash 0 Active	false
Hash 0 Friendly Name	
Hash 0 Stream	

The **Provisioning Time Period** specifies the amount of time (<u>in hours</u>) allowed to configure the Intel® AMTdevice. This time period begins when the ME starts to run on the system and stops when the Intel® AMT system is provisioned.   <u>The provisioning time period will continue to decrement if the ME is running on the system.</u> Once this period of time has elapsed, the system will enter a timed mode. During this timed mode the system only has one hour (per boot) to be configured by the management console.

Remote PC Assist service (formally known as Remote Connectivity Service) allows OEMs, managed service providers (MSP) and IT Outsourcers to connect with enabled systems over the public internet and remotely manage them regardless of system state.

The Hashes specified allow the system to be remotely configured. At least one hash must be active and Remote Configuration Enabled must be set to true to allow remote configuration. The Hash Certificates entered through FITC will be set as default. These Hash certificates will be preserved after a full un-provision.

AMT Configuration Mode — when the mode is **Remote Connectivity Service,** the platform is automatically configured (no additional user configuration required) to interact with Intel Remote Connectivity infrastructure in the Internet.

The Hashes specified allow the system to be remotely configured. At least one hash must be active and Remote Configuration Enabled must be set to true to allow remote configuration. The Hash Certificates entered through FITC will be set as default. These Hash certificates will be preserved after a full un-provision.

ODM ID, system Integrator ID and reserved ID are all used for Intel ® Upgrade service only.

3.14.5.1.1   Setup and Configuration of Intel® RPAT Consumer/Business

Remote Configuration Enabled once set to true to allow remote configuration.

Intel® RPAT code and configuration data are embedded into the ME region. The parameters mentioned below are also available in the FOV section. Please see Appendix A. Setting these parameters correctly greatly eases the effort required by the end user to enable Intel® RPAT.

Remote Connectivity Service capability - Specifies if the platform allows configuration of Remote Connectivity Service (Remote PC Assist Service) capability or not. When the value is "**true**", the platform will have RPAS (formally known as Remote Connectivity service) be enabled on the system and it can start an RPAT session if triggered to do so by MEBX or BIOS.

When the value is "**false**", RPAS (Remote Connectivity Service) code is completely disabled in the Firmware.

 Remote Connectivity Service (Remote PC Assist Service) enabler ID parameter**-** specifies the unique ID of the party (e.g. OEM) which enabled the platform for RPAT (Remote Connectivity Service) mode.

Remote Connectivity Service (Remote PC Assist Service) enabler name parameter **-** specifies the textual description (string) of the party (e.g. OEM) which enabled the platform for Remote Connectivity Service mode.

RCS (Remote PC Assist Service) HW button parameter: This parameter specifies if the system incorporates a hardware button to be used for triggering a RPAT session. Since the hardware button uses the ME section originally used for chassis intrusion sensor - If the parameter is set to 0x001 – ME will treat the signal from the hardware button as chassis intrusion alert. If the parameter is set to 0x002 – ME will treat the signal from the HW button as a "call for help" and will start the RPAT session.

Figure 31: Intel® Remote Connectivity Service Section

Remote Configuration Enabled	true
PKI DNS Suffix	
Remote Connectivity Service Capability	true
Remote Connectivity Service Enabler Id	00000000-0000-0000-0000-000000000000
Remote Connectivity Service Enabler Name	
RCS HW Button	0x01

*Note: When "Remote Connectivity Service capability" = false, the following options are NOT supported values:

1. Change the default value of "Remote Connectivity Service enabler ID" in the FITc tool or FOVS.

2. Change the default value of "Remote Connectivity Service enabler name" in the FITc tool or FOVS.

FPT nor FITC will report errors for the incorrect values stated above.

Table 6 Intel® Remote Connectivity Service (Intel® RPAS) Parameters

Parameter Name	Description	Default value
Remote Connectivity Service (Remote PC Assist Service) capability	Determines if the system supports Remote Connectivity Service RCS supported – 0x01 RCS NOT Supported – 0x00	True
Remote Configuration Enabled	Allow remote configuration.	Enabled:0x01
Remote Connectivity Service (Remote PC Assist Service) enabler ID	Specifies the unique ID of the party (e.g. OEM) which enabled the platform for Remote Connectivity Service mode All 16 byte values between 0x00h - 0xFFFFFFFFh. 0x00h and 0xFFFFFFFFh are not valid values.	None
Remote Connectivity Service (Remote PC Assist Service) enabler Name	Specifies the textual description (string) of the party (e.g. OEM) which enabled the platform for Remote Connectivity Service mode. Textual string. Limited to 60 bytes	None
RCS (Remote PC Assist Service) HW button	Specifies if the HW button is available on the platform. 0x001 – HW button not available (sensor is used for Chassis intrusion detection) 0x002 – HW button is available	0x001

3.15   Building a Flash Image

The flash image can be built using the FITC GUI interface.

To build a flash image using the currently loaded configuration:
1.   Click **Build** on the menu bar.
2.   Select **Build Image**.

—OR—
3.   Specify an XML file with the /b option on the command line.

The FITC uses an XML configuration file and the corresponding binary files to build a McCreary flash image. The following will be produced when building an image:

- Binary file representing the image

- Text file detailing the various regions in the image

- Optional set of intermediate files (see Section 3.6).

- And, if two flash components are specified, multiple binary files containing the image broken up according to the flash component sizes.

The individual binary files can be used to manually program independent flash devices using a flash programmer. However when using FPT, the user should select the single larger binary file.

3.16   Change the region order on the SPI device

The order and placement of the regions in the full SPI image created by FITC can be altered. The location of each region is determined by the order of the region as they are displayed in left hand pane of the FITC window.

Figure 32: Region Order
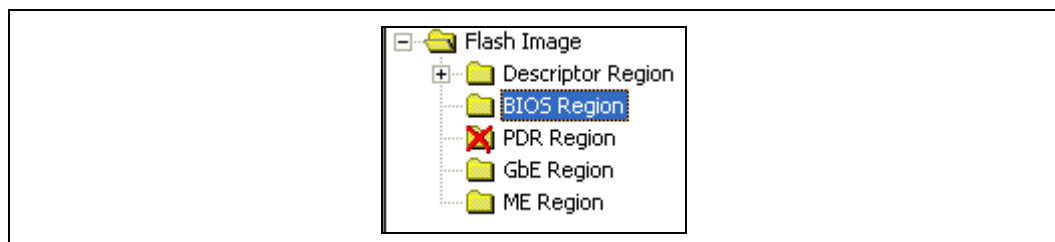
Each region will be added to the full SPI image in the order in which they appear in the list. In Figure 32: Region Order, the Descriptor Region will be the first region in the full image, followed by the BIOS Region. The ME Region will be the last to be added to the full SPI image file.

This can be useful when programming a system with two SPI devices. It is possible to change the order of the regions by clicking and dragging the region to the required location. Figure 24 shows that the BIOS will be placed on the first SPI device and the ME Region will be placed on the second SPI device. The length of each region and the order will determine if that region will be on the first or second SPI device.

3.17   Decomposing an Existing Flash Image

FITC is capable of taking an existing flash image and decomposing it in order to create the corresponding configuration. This configuration can be edited in the GUI just as with any other configuration (see the following sections). A new image can be built from this configuration that is almost identical to the original expect for the changes made by the user.

To decompose an image:
1.   Click **File** on the menu bar.
2.   Select **Open… ,** change the file type filter to the appropriate file type.
3.   Select the required file and click **Open**. The image will automatically be decomposed and the GUI updated to reflect the new configuration.

Alternatively, it is possible to decompose an image by simply dragging and dropping the file onto the main window.

A folder will be created with each of the regions in a separate binary file.

3.18   Command Line Interface

FITC supports command line options. To view all of the supported options, run the application with the -? option. The command line syntax for FITC is:

```
FITC      [<XML_file>]
          [<BIN File>]
          [-?]
          [-H]
          [-B]
          [-O <file>]
          [-ROMBYPASS <true|false>]
          [-ME_disable]
          [-ME <file>]
          [-GBE <file>]
          [-BIOS <file>]
          [-PDR <file>]
          [-CONFIGPARMS <file>]
          [-W <path>]
```

```
[-S <path>]
[-D <path>]
[-U1 <value>]
[-U2 <value>]
[-U3 <value>]
[-I <enable|disable>]
[-FLASHCOUNT <1|2>]
[-FLASHSIZE1 <0|1|2|3|4|5>]
[-FLASHSIZE2 <0|1|2|3|4|5>]
```

<XML_file>—used when generating a flash image file. A sample xml file is provided along with the FITC. When an xml file is used with the /b option, the flash image file will be built automatically.

<Bin File>—decomposes the BIN file. The individual regions will be separated and placed in a folder with the same name as the BIN file name.

-H or -?

displays the command line options.

-B

automatically builds the flash image. The GUI will not be shown if this flag is specified. This option causes the program to run in auto-build mode. If there is an error, a valid message will be displayed and the image will not be built.

If a bin file is included in the command line, this option will decompose the bin file.

-O <file>

path and filename where the image will be saved. This command overrides the output file path in the XML file.

-ROMBYPASS

 Overrides rombyass settings in the XML file.

-ME <file>

overrides the binary source file for the ME Region with the specified binary file.

-GBE <file>

overrides the binary source file for the GbE Region with the specified binary file.

-BIOS <file>

overrides the binary source file for the BIOS Region with the specified binary file.

-CONFIGPARMS <file>

overrides the Configuration Parameters in the XML file with the values in the file specified

-PDR <file>

overrides the binary source file for the PDR Region with the specified binary file.

-FPBA <address>

overrides the flash partition boundary address.

-UBS <value>

overrides the upper block size.

-LBS <value>

overrides the lower block size.

-I <enable|disable>

Enables or disables intermediate file generation.

-W <path>

overrides the working directory environment variable $WorkingDir. It is recommended that the user set these environmental variables first. Suggested values can be found in the OEM Bringup Guide.

-S <path>

overrides the source file directory environment variable $SourceDir. It is recommended that the user set these environmental variables before starting a project.

-D <path>

overrides the destination directory environment variable $DestDir. It is recommended that the user set these environmental variables before starting a project.

-U1 <value>

overrides the $UserVar1 environment variable with the value specified. Can be any value required.

-U2 <value>

overrides the $UserVar2 environment variable with the value specified. Can be any value required.

-U3 <value>

overrides the $UserVar3 environment variable with the value specified. Can be any value required.

-FLASHCOUNT <0, 1 or 2>

overrides the number of flash components in the Descriptor Region. If this value is zero, only the ME Region will be built.

-FLASHSIZE1 <0, 1, 2, 3, 4 or 5>

overrides the size of the first flash component with the size of the option selected as follows:

- 0 = 512KB

- 1 = 1MB

- 2 = 2MB

- 3 = 4MB

- 4 = 8MB

- 5 = 16MB.

-FLASHSIZE2 <0, 1, 2, 3, 4 or 5>

overrides the size of the second flash component with the size of the option selected as follows:

- 0 = 512KB

- 1 = 1MB

- 2 = 2MB

- 3 = 4MB

- 4 = 8MB

- 5 = 16MB.

3.19 Example – Decomposing an Image and Extracting Parameters

The NVARS variables and the current value parameters can be seen by dragging and dropping the 4mb image. The current parameter value will be displayed.

The parameters can also be extracted using the command line method by the following:

Fitc.exe output.bin /b

The above command will create a folder labeled "output". The folder will contain the individual regions (Descriptor, GBE, ME, BIOS), Map file (labeled <FILENAME>_MAP.txt and NVARS.txt file.

The NVARS.txt file will contain the current ME parameters.

The Map file will contain the start, end and length of each region.

3.20   More examples for FITC CLI

NOTE: If using paths defined in the KIT, please be sure to put "" around the path as the spaces will cause issues.

Taking existing (dt_ori.bin) image and creating putting in a new BIOS binary
fitc /b /bios "..\..\..\NVM Image\BIOS\BIOS.ROM" <file.bin or file.xml>

Taking an existing image and putting in a different ME region
fitc /b /me ".\..\..\NVM Image\Firmware\PCH_REL_IGNITION_BYP_ME_UPD_PreProduction_0xB0.BIN" <file.bin or file.xml>

Taking an existing image and putting in a different ME region
Fitc /b /gbe ".\..\..\NVM Image\GbE\82577_A2_IBX_A1_VER0PT21_MOBILE.bin" <file.bin or file.xml>

Taking an existing image and changing ME configuration parameters (NVARs, clock settings, etc)
Fitc /b /configparams Configparams.txt <file.bin or file.xml>

One BKM when dealing with Configparams.txt is to ensure that you write protect (set them as read only) them if using the GUI interface. It is very easy to inadvertently overwrite them when you load up a new Intel ME FW binary.

§

4 *Flash Programming Tool (FPT)*

The Flash Programming Tool (FPT) is used to program a complete SPI image into the SPI flash device(s).

Each region can be programmed individually or all of the regions can be programmed in a single command. The user can perform various functions on the contents of the flash, such as:

- View the contents on the screen.

- Write the contents to a log file.

- Perform a binary file to flash comparison.

- Write to a specific address block.

- Program fixed offset variables

4.1 System Requirements

The DOS version of FPT fpt.exe will run on MS DOS 6.22, DRMKDOS and FreeDOS.

The Windows version fptw.exe requires administrator privilege to run under windows OS. You need to explicitly click on the context menu in Windows " Run as Administrator " under Vista 64/32 and Win7 64/32 bit.

FPT requires an operating system to run on and is designed to deliver a custom image to a computer that is already able to boot, instead of a means to get a blank system up and running. FPT must be run on the system with the flash memory that the user is programming.

One possible flow for using FPT is:
1. Pre-programmed flash with legacy or generic BIOS image is plugged into a new computer.
2. Computer boots.
3. FPT is run and a custom BIOS/ME/GbE image is written to flash.
4. Computer powers down.
5. Computer powers up, boots, and is able to access its ME/GbE capabilities as well as any new custom BIOS features.

4.2 Flash Image Details

A flash image is composed of five regions. The locations of these regions are referred to in terms of where they can be found within the overall layout of the flash memory.

Figure 33: Firmware Image Components

Descriptor	ME/VE				GbE	PDR	BIOS
	Intel® AMT	Intel® RPAT, iRWT, IPT	Braidwood Technology	Intel® QST			
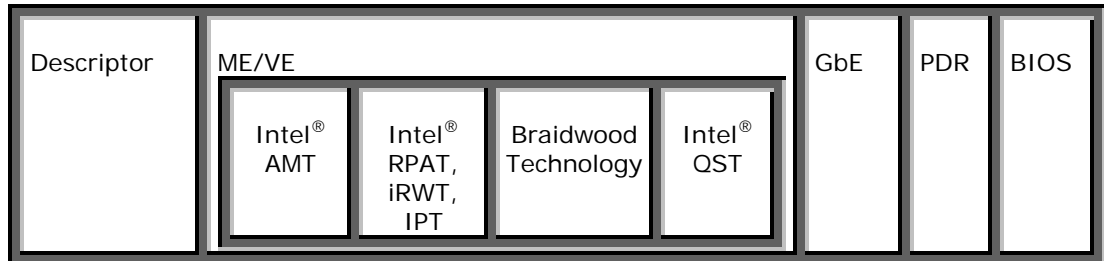
Descriptor—takes up a fixed amount of space at the beginning of flash memory. The descriptor contains information, such as:

- Space allocated for each region of the flash image.

- Read/write permissions for each region.

- A space which can be used for vendor-specific data.

ME—region that takes up a variable amount of space at the end of the descriptor. Contains code and configuration data for ME applications, such as Intel® AMT technology, Intel® NAND, Intel® AT-D, Intel® AT-P, and Intel® Quiet System Technology (Intel® QST).

GbE—optional region that takes up a variable amount of space at the end of the ME Region. Contains code and configuration data for GbE.

BIOS—region that takes up a variable amount of space at the end of flash memory. The BIOS contains code and configuration for the entire platform.

PDR—Platform Descriptor Region that allows system manufacturers to define custom features for the platform.

4.3   Windows* Required Files

The Windows version of the FPT executable is called fptw.exe. The following files must be in the same directory as fptw.exe:

- fparts.txt—contains a comma separated list of attributes for supported flash devices. The text in the file explains each field. An additional entry may be required in this file to describe the flash part which is on the target system. Examine the target board before adding the appropriate attribute values. The file is supplied already populated with default values for SPI devices used with Intel Customer Reference Boards (CRBs).

- fptw.exe—the executable used to program the final image file into the flash.

- sseldrvdll32e.dll—supported library file.

- ssePmxdll32e.dll—supported library file.

- ssepmxdrv.sys—supported system file.

- Fptcfg.ini – contain the FOV that is supported by FPT

4.4   DOS Required Files

The DOS version of the FPT main executable is fpt.exe. The following files must be in the same directory as fpt.exe:

- fpt.exe—the executable used to program the final image file into the flash.

- Fptcfg.ini – contain the FOV that is supported by FPT

- fparts.txt—contains a comma separated list of attributes for supported flash devices. The text in the file explains each field. An additional entry may be required in this file to describe the flash part which is on the target system. Examine the target board before adding in the appropriate attribute values. The file is supplied already populated with default values for SPI devices used with Intel Reference Boards (CRBs).

4.5   Programming the Flash Device

Once the ME has been programmed it will be running at all times. The ME is capable of writing to the flash device at any time, even when the management mode is set to none and it may appear that no writing would occur.

Note: It is important to note that programming the flash device while the ME is running may cause the flash device to become corrupted. The ME SPI accessing should be stopped for any flash accessing before programming the full flash device.

Stopping ME SPI operation use one of the following options:

1. Assert GPIO33 (Flash descriptor override/ME manufacturing jumper) to low while powering on the system. If the parameters are configured to ignore this jumper, this will not be a valid method.
2. Send HMRFPO ENABLE MEI command to ME (detail refer to PCH ME BIOS writer's guide)
3. Temporarily disabling ME through Mebx

Note: Pulling out DIMM from slot 0 or leave empty ME region to stop ME for Ibexpeak platform is not a valid option.

This is **not** a requirement when writing to the fixed offset region.

4.6   Programming fixed offset variables

FPT can program the fixed offset variables. FPT will change the default values of the parameters. The modified parameters will be used by the ME firmware after a Global reset or upon returning from a G3 state. The fixed offset variables can be continuously changed until the *globallocked* bit is set to 0x01. After this bit is set the parameters can **NOT** be modified. To modify the default settings for the parameters, the entire flash device needs to be re-programmed.

The variables can be modified individually or all at once via a text file.

Fpt.exe –FOVs will display a list of the variables supported.

Fpt.exe –EX <Text File> will create a empty text file that will allow the user to update multiple fixed offset variables. The variables will be displayed in the following format:

    [Parmeter name]
    Enabled=0xff
    Value =

In the text file created, variables that NOT enabled (enabled=0xff) will not be modified. Only variables that ARE enabled (enabled=0x1) will be modified.

Fpt.exe –IN <Text file> will update the fixed offset variables with the values as they are entered in the text file.

 A list of all the parameters and their description can be found in the Appendix

4.6.1 Intel® RPAT Consumer/Business, Programming fixed offset variables

In Intel® RPAT both for Consumer/Business there is option to configure several fixed offset variables.

The FOVs related to Intel® RPAT (see below) described above in section13.14.5.1 (Setup and Configuration of Intel® RPAT Consumer/Business)

Table 7 Remote Connectivity Service FOVs Parameters

Parameter Name	Description	Default value
Remote Connectivity Service (Remote PC Assist Service) capability	Determines if the system supports Remote Connectivity Service RCS supported – 0x01 RCS NOT Supported – 0x00	True
Remote Configuration Enabled	Allow remote configuration.	Enabled:0x01
Remote Connectivity Service (Remote PC Assist Service) enabler ID	Specifies the unique ID of the party (e.g. OEM) which enabled the platform for Remote Connectivity Service mode All 16 byte values between 0x00h - 0xFFFFFFFh. 0x00h and 0xFFFFFFFFh are not valid values.	None
Remote Connectivity Service (Remote PC Assist Service) enabler Name	Specifies the textual description (string) of the party (e.g. OEM) which enabled the platform for Remote Connectivity Service mode. Textual string. Limited to 60 bytes	None
RCS (Remote PC Assist Service) HW button	Specifies if the HW button is available on the platform. 0x001 – HW button not available (sensor is used for Chassis intrusion detection) 0x002 – HW button is available	0x001

Parameter Name	Description	Default value
Default Power Package (see Note below)	Default Power Package (Desktop): Pkg1 - ON in S0 Pkg2 - ON in S0, ME Wake in S3, S4-5 Default Power Package (Mobile): Pkg1 - ON in S0 Pkg2 - ON in S0, ME Wake in S3, S4-5 (AC-Only)	Package 1: 0x01 Package 2: 0x02

NOTES:
1. **Note: In all Intel® RPAT Consumer: **Default Power Package** value is 0x01
2. In all Intel® RPAT Business: **Default Power Package** value is 0x02

4.7   Usage

Note: To prevent possible firmware corruption, the user should disable the firmware before programming any SPI flash devices. Refer to the previous section.

Both the Windows version and DOS version of the FPT can run with command line options. To view all of the supported commands, run the application with the -? option. The commands in both the DOS and Windows versions have the same syntax. The command line syntax for fpt.exe and fptw.exe is:

FPT   [-?]
          [-H]
          [-C]
          [-B]
          [-I]
          [-F:<file>]
          [-VERRIFY:<file>]
          [-D:<file>]
          [-ADDRESS:<value>]
          [-LENGTH:<value>]
          [-L]
          [-DESC]
          [-BIOS]
          [-ME]
          [-GBE]
          [-PDR]
          [-Y]
          [-Q]
          [-E]
          [-ERASE]
          [-P:<file>]
          [-LIST]
          [-FOVS]
          [-EX]
          [-U]

```
[-O]
[-IN]
[-N]
[-ID]
[-V]
[-LOCk]
[-DUMPLOCk]
[-PSKFILE]
[-CLOSEMNF]
      [-VERBOSE] <file>
[-GRESET]
[-VER]
[-EXP]
```

-? or –H

   displays the help screen.

-C

asks the user to confirm that the entire flash part will be erased. It is not necessary to erase the flash before a load. The load command will erase the region before a load is performed. If two flash devices are present, both devices will be erased. You need to disable ME before clearing the ME region . More detail please refer to 4.5.

-B

checks to see whether the flash has been erased and generates a message stating whether or not the flash is blank. If there are two flash devices and neither are blank, the program will return with a non-blank message.

-I

Displays information about the flash image. This information includes:

- Start and end of each region

- Read and write permissions

- Whether or not the flash descriptor is valid.

-F

 loads a binary file into the flash starting at address 0x0000. The flash device must be written in 4kB sections. The total size of the flash device must also be in increments of 4kB. After the program process is completed, FPT will automatically verify (-verify) the image was programmed correctly. The verification will always run after using the -f option.

-VERIFY

compare binary file to the image in the flash. If the binary file is not identical to the flash, the address and expected value of the first 5 bytes will be displayed on the screen. The flash device must be written in 4kB sections. The total size of the flash device must also be in increments of 4 KB. This must be performed immediately after programming the SPI flash device. This option is automatically executed after performing the -f option.

-D

dumps the flash contents to a file or to the screen using the STDOUT option. The flash device must be written in 4KB sections. The total size of the flash device must also be in increments of 4 KB.

-ADDRESS or –A

used in conjunction with load, verify or dump, and allows the user to load, verify or dump a file beginning at the specified address. This option cannot be used with the -desc, -bios, -me or -gbe options.

-LENGTH or –L

used in conjunction with the load, verify or dump options, and allows the user to specify the number of bytes to load, verify or dump. This option cannot be used with the -DESC, -BIOS, -ME or -GBE option.

-DESC

used in conjunction with the load, verify or dump options, and allows the user to load, verify or dump to the descriptor region leaving the rest of the flash untouched. This option cannot be used with the -ADDRESS or -LENGTH option.

-BIOS

used in conjunction with the load, verify or dump options, and allows the user to load, verify or dump to the BIOS Region leaving the rest of the flash untouched. This option cannot be used with the -ADDRESS or -LENGTH option.

-ME—used in conjunction with the load, verify or dump options, and allows the user to load, verify or dump to the ME Region leaving the rest of the flash untouched. This option cannot be used with the -ADDRESS or -LENGTH option.

-GBE—used in conjunction with load, verify or dump, and allows the user to load, verify or dump to the GBE Region leaving the rest of the flash untouched. This option cannot be used with the -ADDRESS or -LENGTH option.

-PDR

used in conjunction with load, verify or dump, and allows the user to load, verify or dump to the PDR Region leaving the rest of the flash untouched. This option cannot be used with the -ADDRESS or -LENGTH option.

-Y

do not prompt when a warning occurs. If a warning occurs, the warning will be displayed, however, the specified command will continue to run.

-G

do not display output to the screen.

-E

do not erase any area before writing to the flash.

-ERASE

 Erase the contents of the flash

-P

specifies a different flash part definition file to use instead of the one located within the executable.

-LIST

list all the SPI devices supported

-FOVs

list the names and id numbers of all fixed offset variables (FOVs) supported.

-EX -O <List filename>

extracts list of variables and the current value to the text file specified

-U

updates parameter specified by -N or -ID option.

-IN <FOV filename>

specifies the fixed offset parameter file to update all fixed offset variables.

-N

specifies the name of the variable to update using the -U and -V option

-id

specifies the name of the variable to update using the -U and -V option

-V

specifies the value of the variable. Used with -U and -N or -ID option

-LOCK

locks the descriptor region according to Intel® recommendation. Please see section 4.7.3 Region Access Control for more information.

-DUMPLOCK

displays the current descriptor lock settings

-PSKFILE <PSK filename>

species the name of the PSK file that FPT can read and program PSK value for multiple systems

-CLOSEMNF

Option used at the end of the manufacturing line. Please see Section 4.10 End of Manufacture for more details

-GRESET

Initiate a global reset. A global reset is sufficient to allow the ME to use the update Fixed offset variables. Please check with your BIOS to ensure global resets are permitted for manufacturing purpose.

-VER

show the version of the tools

-VERBOSE <file>

   Display the debug information of the tool or store that in a log file

-EXP

show the examples on how to use the tools

4.8   Update Hash Certificate through FOV

There are 23 certificate hash values that can be stored in ME region. 0-19 are default certificate which are not be deleted by a full un-provisioning process (caused by MEBX, RTC reset or application). Certificates 20-22 are not default certificates and will be deleted after a full un-provisioning. You can change certificates 19-21 by FOV (with FPT or other flash programming methods) or FITC. Certificates 0-18 and certificate 22 are only configurable by FITC. Certificate 19 is the only default certificate that can be configured by FOV.

1)   User must copy the raw hash values from a valid certificate file

Figure 34: Raw Hash value from certificate file
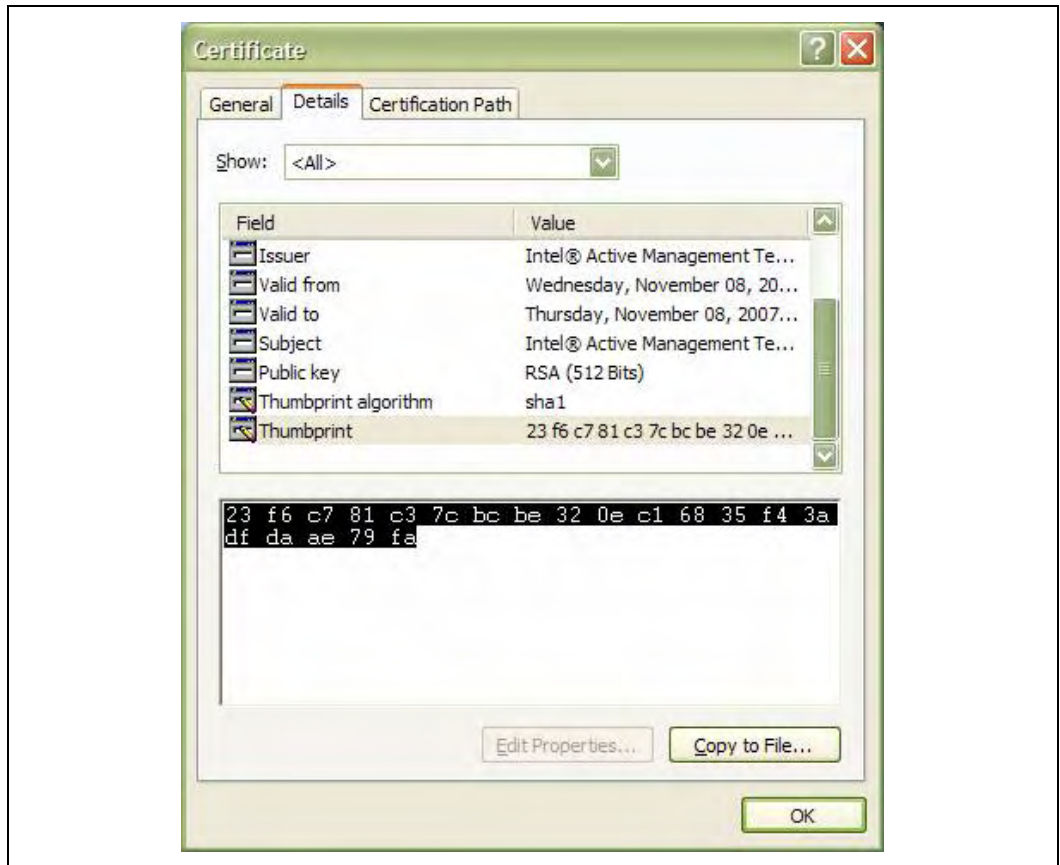
2)   Paste the raw hash contents to a text file and remove all the spaces from there and save that file as hash.txt. This hash file must exist in the same folder as the sampleparm.txt file mentioned in step 3.

Figure 35: Sample Hash BIN file
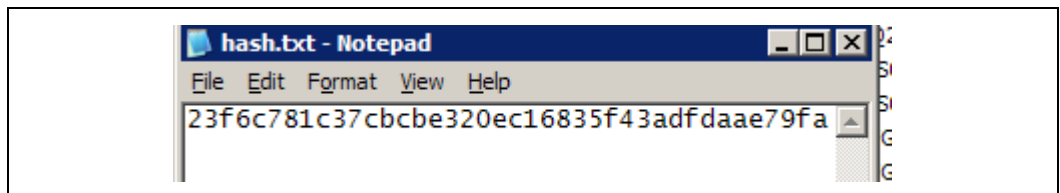
3)   Hash FOV can only be flashed using FPT's   **-u –in**   option like this:

         fpt –u –in sampleparam.txt

Where sampleparam.txt is the file that is used to update multiple FOVs *(fpt.exe /ex /o sampleparm.txt)*. In this case we want to update FOV as well. So user must include following entries to the sampleparam.txt file:

[ZTCEnable]

Enabled = 0x0
Value = 0x00

[Hash1]
Enabled = 0x1
IsActive = 0x1
FriendlyName = myHash3
RawHashFile = hash.txt

[CfgSrvFQDN]
Enabled = 0x0
Value = Intel.com

4.9   fparts.txt File

The fparts.txt file contains a list of all flash devices that the FPT supports. The flash devices listed in this file must contain a 4 KB erase block size. If the flash device is not listed, the user will receive the following error:

Flash Programming Tool.   Version X.X.X
Reading LPC BC register… 0x00000000
BIOS space write protection is enabled
Disabling BIOS space write protection
Reading LPC RCBA register… 0xFED1C001
SPI register base address… 0xFED1F020
Loading the flash definition file
Reading file "fparts.txt" into memory…
Initializing SPI utilities
Reading HSFSTS register… Flash Descriptor: Valid

         --- Flash Devices Found ---
>>> Error: There is no supported SPI flash device installed!

If the device is not located in the fparts.txt file, the user is expected to provide information about their device and insert the values into the file using the same format as the rest of the devices. The device must have a 4KB erase sector and the total size of the SPI Flash device must be a multiple of 4KB. The values are listed in columns in the following order:

- Display name

- Device ID (2 or 3 bytes)

- Device Size (in bits)

- Block Erase Size (in bytes - 256, 4K, 64K)

- Block Erase Command

- Write Granularity (1 or 64)

- Unused

- Chip Erase Command.

4.10   End of Manufacture

Before a platform leaves the manufacturing floor, the descriptor region must be locked, ME region must be locked, the MEManuf counter must be set to 0, and the Global locked bit must be set.

In the past, steps 1 to 3 were performed individually by separate tools.

To end manufacture, perform the following actions:

1.   Set descriptor permissions for each region.
2.   Set MEManuf Counter to zero.
3.   Set Global locked bit.

When used with the *-closemnf* flag, the FPT provides a single command that performs all of these operations. It is possible to lock them at image creation phase. If that is the case, it is not necessary to run this command in manufacturing line.

4.11   Examples

The following examples illustrate the usage of the DOS version Fpt.exe of the tool. The Windows version Fptw.exe will behave in the same manner apart from running in a Windows environment.

4.11.1   Example 1 – Flash SPI flash device with binary file

C:\ fpt.exe –f spi.bin

This usage will write the data in spi.bin file into whole SPI flash from address 0x0x

4.11.2 Example 2 –Program a specific region

Fpt –f –BIOS bios.rom

Flash Programming Tool. Version X.X.X

Reading file "fparts.txt" into memory...
Initializing SPI utilities
Reading HSFSTS register... Flash Descriptor: Valid

     --- Flash Devices Found ---
     SST25VF016B   ID:0xBF2541   Size: 2048KB
(16384Kb)

Using software sequencing.
Reading LPC BC register... 0x00000001
Reading file "BIOS.ROM" into memory...
- Erasing Flash Block [0x101000]... - 100% complete.
- Programming Flash [0x100400]... - 100% complete.
Write Complete

This usage will write the data in bios.bin file into BIOS region of SPI flash and verifies that the operation ran successfully.

4.11.3 Example 3 –Program SPI flash from a specific address

fpt.exe -F image.bin -A 0x100 -L 0x800

This usage loads 2KB of the binary file image.bin starting at address 0x0000. The starting address and the length must be a multiple of 4KB.

4.11.4 Example 4 – Dump Specific Region

fpt.exe –d -desc descdump.bin

Flash Programming Tool. Version X.X.X

Reading file "fparts.txt" into memory...

Initializing SPI utilities
Reading HSFSTS register... Flash Descriptor: Valid

     --- Flash Devices Found ---
     SST25VF016B   ID:0xBF2541   Size: 2048KB
(16384Kb)

Using software sequencing.

- Reading Flash [0x000040]… 4KB of   4KB - 100% complete.
Writing flash contents to file "descdump.bin"…

Memory Dump Complete

This usage writes the contents of the Descriptor Region to the file descdump.bin.

4.11.5   Example 5 – Display SPI information

fptw.exe -i
Flash Programming Tool. Version X.X.X

Reading LPC BC register… 0x00000001
Reading LPC RCBA register… 0xFED1C001
SPI register base address… 0xFED1F020
Loading the flash definition file
Reading file "fparts.txt" into memory…
Initializing SPI utilities
Reading HSFSTS register… Flash Descriptor: Valid

        --- Flash Devices Found ---
        SST25VF016B   ID:0xBF2541   Size: 2048KB
(16384Kb)

Using software sequencing.

        --- Flash Image Information --
        Signature: VALID
        Number of Flash Components: 1
                Component 1 - 2048KB (16384Kb)
        Regions:
                Descriptor - Base: 0x000000, Limit: 0x000FFF
                BIOS   - Base: 0x100000, Limit: 0x1FFFFF
                ME   - Base: 0x001000, Limit: 0x0FDFFF
                GbE   - Base: 0x0FE000, Limit: 0x0FFFFF
        Master Region Access:
                CPU/BIOS - ID: 0x0000, Read: 0xFF, Write: 0xFF
                ME   - ID: 0x0000, Read: 0xFF, Write: 0xFF
                GbE   - ID: 0x0218, Read: 0xFF, Write: 0xFF

This usage displays information about the flash devices present in the computer. The base address refers to the start location of the particular regions and the limit address refers to the end of the region. If the flash device is not specified in fparts.txt, Fpt will return the error message "There is no supported SPI flash device installed".

4.11.6   Example 6 – Verify Image with errors

fpt.exe -verify outimage.bin
Flash Programming Tool. Version X.X.X
Reading LPC BC register… 0x00000001
Reading LPC RCBA register… 0xFED1C001
SPI register base address… 0xFED1F020
Loading the flash definition file
Reading file "fparts.txt" into memory…
Initializing SPI utilities
Reading HSFSTS register… Flash Descriptor: Valid
        --- Flash Devices Found ---
        SST25VF016B   ID:0xBF2541   Size: 2048KB
(16384Kb)
        SST25VF016B   ID:0xBF2541   Size: 2048KB
(16384Kb)
Using software sequencing.
Reading file "outimage.bin" into memory…

RESULT: Data does not match!
        0x00000000: 0x5A - 0x5A
        0x00000001: 0xA5 - 0xA5
        0x00000002: 0xF0 - 0xF0
        0x00000003: 0x0F - 0x0F
        0x00000004: 0x01 - 0x01

This usage compares the ME Region programmed on the flash with the specified
firmware image file outimage.bin. If the -y option is not used, the user will be notified
that the file is smaller than the binary image. This is due to extra padding that is
added during the program process. The padding can be ignored when performing a
comparison. The -y option will proceed with the comparison without warning.

4.11.7   Example 7 –Verify Image successfully

fpt.exe -verify outimage.bin

Flash Programming Tool. Version X.X.X

Reading file "fparts.txt" into memory…
Initializing SPI utilities
Reading HSFSTS register… Flash Descriptor: Valid

    --- Flash Devices Found ---
    SST25VF016B ID:0xBF2541 Size: 2048KB (16384Kb)

Using software sequencing.
Reading file "outimage.bin" into memory…

RESULT: Data does not match!
    [0x000000] Expected: 0x0B, Found: 0x5A
Total mismatches found in 64 byte block: 27

This usage compares the file image.bin with the contents of the flash. Comparing an image should be done immediately after programming the flash device. Verifying the contents of the flash device after a system reset will result in a mismatch because ME will change some data in the flash after a reset.

4.11.8   Example 8 – Program FOV parameter

fpt.exe -u -n "AMTConfigMode" -v 0x03

Flash Programming Tool. Version X.X.X

Reading file "fparts.txt" into memory…
Initializing SPI utilities
Reading HSFSTS register… Flash Descriptor: Valid

   --- Flash Devices Found ---
   SST25VF016B ID:0xBF2541 Size: 2048KB (16384Kb)

Updating software sequencing.
Reading region information from flash descriptor
Reading FOV configuration file "fptcfg.ini"
Updating variable [AMTConfigMode]..

This usage updates the default configuration mode. In this example the Configuration mode was set to **Remote Connectivity Service.** This action is only supported if Remote Connectivity service is supported on the system. FPT will not report dependency errors. Please be sure that the values selected are valid.

§

5 *MEManuf and MEManufWin*

MEManuf validates ME functionality (verifies that all its components have been assembled together correctly) on the manufacturing line.

The Windows version MEMANUFWIN requires administrator privilege to run under windows OS. You need to explicitly click on the context menu in Windows " Run as Administrator " under Vista 64/32 and Win7 64/32 bit.

MEManuf does not check for LAN functionality. The tool assumes that all ME components on the test board have been validated by their respective vendors. The tool verifies that these components have been assembled together correctly.

There are different sets of tests available in ME FW. MEMANUF will call different sets of tests according to FW SKU or the option used on the command.

For the Ibexpeak platform, here is a list of the tests that could be execute by MEMANUF

1. Kernel build-in self test

2. AMT module self test

3. VE self test

4. Host based test including VSCC test, ME FW code region data integrity check

Here is a table to describe the sets of the detail tests available in MEMANUF

Table 8. Tests that are available in MEMANUF

Kernel BIST (AMT disabled)	Flash
	SmBus
	EC (Power Supply)
	Wlan
VSCC Check	Retrieve and check installed flash device JEDEC ID and programmed VSCC value against Intel known good VSCC value (ME and BIOS)

	-S5	-S0	-S4
AMT BIST test (AMT enabled)	Flash	Flash	Flash
	Mlink	Mlink	Mlink
	BIOS	BIOS	BIOS
	SmBus	SmBus	SmBus
	ME-EC - Power Supply	ME-EC - Power Supply	ME-EC - Power Supply
	Kedron – Wlan (This test is skipped if -nowlan is provided)	Kedron – Wlan (This test is skipped if -nowlan is provided)	Kedron – Wlan (This test is skipped if -nowlan is provided)
	Power Reset	N/A	Hibernation
AMT extended BIST test (AMT enabled)	Sprite Device (this test is skipped if Intel® integrated graphics card is disabled/not present)		
	KVM related tests (Sampling, Comparator and Compressor Engine, sampling test is skipped if Intel® integrated graphics card is disabled/not present)		
	USBR HW		

* Note: KVM feature only work with Intel integrated graphic engine.

5.1 Windows* PE requirements

For Intel® AMT the following drivers are required:

- The ME Interface driver must be installed in the Windows PE image.

- The Windows PE image must be WMI enabled.

5.2 Firmware test Counter

The AMT built in self test will include a reboot cycle, and for security reasons, the firmware counter tracks the number of times the AMT manufacturing test command has been sent to the Host Interface. When the counter reaches zero, any AMT manufacturing test command issued to the Host Interface is no longer acknowledged.

Use of the MEManuf test with reboot on AMT enabled platform decrements this counter with each run. This limits the number of times a test system can be repaired in order to have it pass the manufacturing test.

Once the counter has reached zero, the image needs to be reprogrammed into the SPI flash device or the counter must be changed using the FOV mechanism . If the CPU does not have write access to the Descriptor Region or the global locked bit is set, the counter can only be reset by reprogramming the image using the flash override jumper (GPIO 33) if needed.

5.3   How to use MEMANUF

MEMANUF will check the FW SKU and run the proper tests accordingly unless there is an option specified to select tests. If AMT is enabled on the platform, it will cause a reboot automatically. Because tool can not store the test result over the reboot cycle, there is another command to retrieve the result.

MEMANUF

MEMANUF -R

By running this, you don't have to know the detail of the platform/firmware SKU. It will automatically detect the FW SKU and return the test result for MEMANUF –R.

If you have run MEMANUF with an option the first time, you need to run exactly same option with –R for the next MEMANUF run. Otherwise tools may give you error message or empty test result.

For example:

If you run MEMANUF –NAND , you will need to run MEMANUF –NAND –R to retrieve the result. If you run MEMANUF –AMT –R to retrieve the result, MEMANUF will report failure on the test and give detail failure information in –verbose mode.

AMT Sx (S4, S5) test is the test that will have to include a reset. These test will test the ME behavior when system in Sx state. It is highly recommended to run this at least once in you manufacturing process to make sure your HW component is working properly to support AMT working at Sx.

5.4   Usage

The DOS version of the tool can be operated using the same syntax as the Windows version. The Windows version of the tool can be executed by:

MEMANUF [-S5/-S4/-S0][-AMT/-NOAMT][-block][-counter][-NETOFF/-NETON][-R][-NOWLAN][-VERBOSE <file>][-EXP] [-H][-?]

No option

The tool will request the FW to run a complete hardware built in self test which includes a power reset (S5) at the end of the test. If AMT is disabled, this will run the BIST that is provided by the ME kernel, which executes only a subset of tests that AMT BIST covers. The user will be notified when the program is run on an AMT-disabled system. The BIST that kernel provides does not decrement the run counter.

For AMT systems this includes a power cycle (S5) at the end of the test. The tool will need to query the FW when it starts if it is the first or second run (before or after power down).

In addition, the tool will run some host based tests that are listed in the previous section. VSCC Table validation and Code integrity check are the host tests that will run for all test cases. The KVM enablement check will be run only when AMT is enabled.

-S5

Same as No option, except if AMT is not available on the platform it will report an error message saying "AMT is not enabled to run the tests for the system" will be displayed.

-S4

Same as –S5 option, except that the system will try to hibernate (S4) instead of power reset (S5).

Note: This option is only available for the Windows version of MEManuf, and the test will not run successfully if the system cannot go into hibernate mode or the power package selected does not support the ME running in the S4 state.

-S0

do ME selftest without reset-hibernate if AMT is enabled

-AMT

This option indicates that a user wants to run AMT only built in self test with a power reset (S5). If this option is used on a system with AMT disabled, an error message saying "AMT is not enabled to run the tests for the system" will be displayed.

-NOAMT

do non-AMT tests without reset-hibernation

-BLOCK

The tool will block future AMT test with Sx support runs of the test for security reasons. This is meant to be the last line in a manufacturing script. It however continues allowing ME self test without a reboot to be run. For other SKUs/ AMT is disabled, an error "Error 9527: The specified test cannot be executed since Intel $^{®}$ AMT is not found or the Manageability mode is not Intel $^{®}$ AMT" is returned to the user.

-COUNTER

The tool will return the value of the run counter in a message like this "Full Test Counter: XXX". For other SKUs with AMT is disabled, an error "Error 9527: The specified test cannot be executed since Intel $^{®}$ AMT is not found or the Manageability mode is not Intel $^{®}$ AMT" is returned to the user.

-NETOFF

This option will re-enable the integrated Gbe wired/wireless LAN interface so that network traffic can go in/out of it. If AMT is disabled, an error "Error 9257: Cannot run the command since Intel® AMT is not available" is returned to the user

-NETON

This option will block any network traffic that goes in/out of the integrated Gbe wired/wireless LAN interface. If AMT is disabled, an error "Error 9257: Cannot run the command since Intel® AMT is not available" is returned to the user.

-R

Get the test result stored on the platform. More detail in Example

-NOWLAN

This option only applies to AMT test so that a user can skip wireless LAN NIC test if he/she has no wireless LAN NIC attached the hardware, otherwise tools will report a false failure about wireless LAN NIC.

-VERBOSE <file>

Display the debug information of the tool or store that in a log file

-H or -?

Display help screen

-S4 will only be available if OS support hibernate and can enter S4 within 60 seconds

-S5, -S4 ,-S0 only can be used on AMT enabled platform with AMT test

Table 9. MEMANUF test Matrix

VSCC test is called for all option other than the hidden –NoVSCC option.

	Intel® vPro™ SKU	8M Consumer SKU	4M SKU
No option	AMT Extend + VE test+ AMT S5	Run Kernel test +VE test if Intel ® NAND is enabled	Run Kernel Test +VE test if Intel ® NAND is enabled
-S4	AMT Extend + VE test + AMT S4I	Error: -S4 option is only available with AMT test	Error: -S4 option is only available with AMT test
-S5	AMT Extend +VE test + AMT S5	Error: -S4 option is only available with AMT test	Error: -S4 option is only available with AMT test

	Intel® vPro™ SKU	8M Consumer SKU	4M SKU
-S0	AMT Extend + VE test + AMT S0	Error: -S0 option is only available with AMT test	Error: -S0 option is only available with AMT test
-AMT	AMT Extend + AMT S5	Error: AMT is not available on this FW	Error: AMT is not available on this FW
-NAND	VE test	Error: NAND is not available on this FW	Error: NAND is not available on this FW
-NoAMT	Kernel test +VE test	Run Kernel test +VE test if Intel ® NAND is enabled	Run Kernel Test +VE test if Intel ® NAND is enabled
-NoNAND	AMT Extend + AMT S5	Run Kernel test	Run Kernel Test
-AMT –S4	AMT Extend + AMT S4	Error: AMT is not available on this FW	Error: AMT is not available on this FW
-AMT –S5	AMT Extend + AMT S5	Error: AMT is not available on this FW	Error: AMT is not available on this FW
-AMT –S0	AMT Extend + AMT S0	Error: AMT is not available on this FW	Error: AMT is not available on this FW

5.5 Examples

5.5.1 Example 1

MEManufWin.exe –s4

This usage runs the AMT test with reboot if AMT is enabled on this platform , however, instead of a hard power cycle, MEManuf will send Windows into the S4 hibernate mode and then bring the system back to the S0 state. This command should be used again to view the test results. If the power package selected does not support the ME in the S4 state, MEManuf will not run and will return the following error message:

"Intel® AMT power policy prevents ME from bringing the system back from hibernation, so hibernation will not be performed. All other tests ran successfully."

You need to use MEMANUFWIN.exe –S4 –R to retrieve the test result.

Intel(R) MEManuf Version: 6.0.0.9348

Copyright(C) 2005 - 2009, Intel Corporation. All rights reserved.

MEManuf Test Passed

5.5.2 Example 2

MEManufWin.exe -block

This usage sets the MEManuf test counter to 0 (zero) and prevents any more –S4, -S5 tests from being executed. Other tests will still be allowed. If the user needs to run additional full or graceful tests, the complete SPI image must to be reprogrammed.

5.5.3 Example 3

MEManufWin.exe –S5

This usage will immediately send the computer into an S5 state and then power back on if AMT is enabled on the platform. To view the results, the user must run the MEMANUF –S5 -R option after previous MEMANUF –S5 test. If this command is invoked on Windows, the user may lose unsaved data.

5.5.4 Example 4: Consumer Platform

MEManufWin.exe –NOAMT

This usage will execute Kernel and VE tests on a consumer platform. There system will not power cycle at the end of this test.

§

6   *MEInfo*

MEInfoWin and MEInfo provide a simple test to check whether the ME firmware is alive or not. Both tools perform the same test, query the Intel® ME firmware including AMT and QST, and retrieve data. Table 8 is a list of the data that each tool will return.

The Windows version MEINFOWIN requires administrator privilege to run under windows OS. You need to explicitly click on the context menu in Windows " Run as Administrator " under Vista 64/32 and Win7 64/32 bit.

6.1   Windows* PE requirements

For Intel® AMT the Intel ME Interface driver must be installed in the Windows PE image.

The Windows PE image must be WMI enabled.

MEInfo will report an LMS error. This is expected behavior as the LMS driver cannot be installed on Windows PE.

6.2   Usage

The executable can be invoked by:

MEINFO.exe
MEInfo.exe [-feat <name> -value <value>]
MEInfo.exe [-feat <name>]

MEINFO.exe -FWSTS

MEINFO.exe [-H]

MEINFO.exe [-?]

MEINFO.exe –verbos <filename>

No option:

If tool is invoked without parameters, the tool will report information for all components listed in Table 10 below for full SKU firmware. On Ignition firmware, the tool simply dumps its firmware status register into a human readable format and then exits.

Table 10. List of components for which version information will be retrieved

Component	Common for all Common Services SKUs	Specific SKU difference	Field value
Tools version	X		A version string
PCH version	X		A version string
FW version	X		A version string
BIOS version	X		A version string
GbE version	X		A version string
MEBx Version	X		A version string
VendorID	X		A number (in Hex)
Wireless Driver/Hardware Version	X		A version string
Link status	X		Link up/ down

FW Capabilities	X		Intel® AMT, TPM and other SKUs, and their possible combinations (in Decimal value and its bits definition breakdown)
Cryptography Support	X		Enabled/ Disabled
Flash Lockdown	X		Enabled/ Disabled/Unknown
Host Read Access to ME	X		Enabled/Disabled/Unknown
Host Write Access to ME	X		
Last ME reset reason	X		Power up/ Firmware reset/ Global system reset
Intel® AMT State		Not available on consumer SKU	Enabled/Disabled
Intel® Standard Manageability State		Not available on consumer SKU	Enabled/Disabled
BIOS boot State	X		Pre Boot/ In Boot/ Post Boot
FW_STS	X		Hexadecimal number and its bit definition breakdown
System UUID	X	Not available on consumer SKU	UUID of the system

Configuration state	X	Not for CP	Not started/ In process/ Completed
Provisioning Mode	X	Not available on consumer SKU	PKI/PSK/Remote Connectivity Service/None
FW behavior on Flash Descriptor Override Pin-Strap	X		Continue / Halt Note: '1': 'Ignore' Continue '0': 'Normal' Halt
MAC Address	X		A MAC address (in Hex separated by "–")
Wireless MAC address	X		A MAC address (in Hex separated by "-")
IPv4 Address (Wired and Wireless)	X		IPv4 IP address (in decimal separated by ".")
IPv6 Address (Wired and Wireless)	X	Not available on Consumer SKU	All IPv6 IP addresses
IPv6 Enabled	X	Not available on Consumer SKU	Enabled/ Disabled
FWU Override Counter	X		(A number)/ Always/ Never
FWU Override Qualifier	X		Never/Always/Restricted
Local FWUpdate	X		Enabled/ Disabled

Secure FWUpdate			Enabled/Disabled
MEI Driver version*	X		A version string
LMS version*	X		A version string
UNS version*	X		A version string
Wireless Driver Version*	X		A version string
SPI Flash ID	X		An SPI Flash ID
VSCC register value	X		VSCC Register value for SPI on system (both JEDECID and VSCC values in Hex)
Identity Protection Technology	X	Not available on Corporate SKU	Enabled/Disabled
Identity Protection Technology Version	X	Not available on Corporate SKU	A version string
Identity Protection Technology Status	X	Not available on Corporate SKU	Disabled/Not Configured/Running/Unknown
Capability Licensing Service	X		Enabled/Disabled

Capability Licensing Service Status	X		Permit info not available/Upgraded/Not Upgraded/Not Upgradable
Remote PC Assist Service Registered	X		True/False
Remote PC Assist Service Enabler ID	X		Format in UUID. All values between "0x00 - 0xffffffff" are valid except for the boundary cases - all zeros or all 0xFFs are invalid.
Remote PC Assist Service Enabler Description	X		A human readable string to describe the party represented by Enabler ID
Tools version	X		A version string
PCH version	X		A version string
BIOS version	X		A version string
GbE version	X		A version string
MEBx Version	X		A version string
Netstack Version	X		A version string
Managgeability Mode	X		A string

FT Version	X		A version string Note: FT is 'Fault Tolerant'
FT Build Number	X		A number Note: FT is 'Fault Tolerant'
Non-FT Version	X		A version string Note: FT is 'Fault Tolerant'
Non-FT Build Number	X		A version string Note: FT is 'Fault Tolerant'
VendorID	X		A number (in Hex)
Wireless Driver/Hardware Version	X		A version string
Link status	X		Link up/ down
FW Capabilities	X		Intel® AMT, TPM and other SKUs, and their possible combinations (in Decimal value and its bits definition breakdown)
Cryptography Support	X		Enabled/ Disabled
Flash Lockdown	X		Enabled/ Disabled/Unknown
Host Read Access to ME	X		Enabled/Disabled/Unknown

Host Write Access to ME	X		
Last ME reset reason	X		Power up/ Firmware reset/ Global system reset
BIOS boot State	X		Pre Boot/ In Boot/ Post Boot
FW_STS	X		Hexadecimal number and its bit definition breakdown
System UUID	X		UUID of the system
Configuration state	X	Not for CP	Not started/ In process/ Completed
Enrollment State	---	Enrollment State for CP only	Enrollment Status Enrolled Provider
Provisioning Mode	X		PKI/PSK/Remote Connectivity Service/None
FW behavior on Flash Descriptor Override Pin-Strap	X		Continue / Halt Note: '1': 'Ignore' → Continue '0': 'Normal' → Halt
MAC Address	X		A MAC address (in Hex separated by "–")
Wireless MAC address	X		A MAC address (in Hex separated by "-")

IPv4 Address (Wired and Wireless)	X		IPv4 IP address (in decimal separated by ".")
IPv6 Address (Wired and Wireless)	X	Not available on Consumer SKU	All IPv6 IP addresses
IPv6 Enabled	X	Not available on Consumer SKU	Enabled/ Disabled
FWU Override Counter	X		(A number)/ Always/ Never
FWU Override Qualifier	X		Never/Always/Restricted
Local FWUpdate	X		Enabled/ Disabled
Secure FWUpdate			Enabled/Disabled
MEI Driver version*	X		A version string
LMS version*	X		A version string
UNS version*	X		A version string
Wireless Driver Version*	X		A version string

	X		An SPI Flash ID
SPI Flash ID			An SPI Flash ID
VSCC register value	X		VSCC Register value for SPI on system (both JEDECID and VSCC values in Hex)
Remote Connectivity Service Registered	X		True/False
Remote Connectivity Service Capability	X		True/False
Remote Connectivity Service Enabler ID	X		Format in UUID. All values between "0x00 - 0xffffffff" are valid except for the boundary cases - all zeros or all 0xFFs are invalid.
Intel® Identity Protection Technology Version		Consumer SKU only	A string containing the revision information of the Intel IPT firmware
Intel IPT Status		Consumer SKU only	A string containing the status of IP Firmware: Enabled Disabled Not Configured Error
Remote Connectivity Service Enabler Description	X		A human readable string to describe the party represented by Enabler ID.

-feat < name> -value <value>—compares the value of the given feature name with the value in the command line. If the feature name or value is more than one word, the entire name or value must be enclosed in quotation marks. If the values are identical, a message will display indicating success. If the values are not identical, the actual value of the feature will be returned. Only one feature may be requested in a command line.

-feat <name> - retrieves the current value for the specified feature. If the feature name is more than one word, the entire feature name must be enclosed in quotation marks. The feature name entered must be the same as the feature name displayed by MEInfo

MEInfo can retrieve all of the information detailed below, however, depending on the SKU selected, some information may not appear.

MEINFO –FWSTS

This option will decode the ME firmware status register value field, and break it down into the following bit definitions for easy readability:

FW Status Register: 0x00000245

CurrentState: *Normal*

ManufacturingMode: *Disabled*

FlashPartition: *Valid*

OperationalState: *M0 with UMA*

InitComplete: *Complete*

BUPLoadState: *Success*

ErrorCode: *No Error*

ModeOfOperation: *Normal*

Phase: *Unknown (8)*

-Verbose <filename>

Turn on additional information about the operation for debugging purpose. This option has to be used together with the above mentioned option(s). Fail to do so will generate an error "Error 9254: Invalid command line option".

This option will works with no option, -feat.

-H or -?:

Display the list of command line options supported by MEInfo tool.

6.3   Examples

6.3.1   Example 1

This is a simple test that indicates whether the firmware is alive and if so, will return device specific parameters. The output is from the Windows version. The DOS version will not display the UNS version, Intel Management Engine Interface or LMS version numbers.

MEINFO.exe

Intel(R) MEInfo Version: 6.0.0.7084

Copyright(C) 2005 - 2009, Intel Corporation. All rights reserved.

Intel(R) Manageability and Security Application code versions:

BIOS Version: 4.6.3

MEBx Version: 6.0.3.3

Gbe Version: 7.16.0

VendorID: 8086

PCH Version: 400004

FW Version: 6.0.0.7080

FW Capabilities: 6741605

   Intel(R) Active Management Technology

   Intel(R) Anti-Theft Technology PC Protection

   Intel(R) Remote PC Assist Technology

   Intel(R) Capability Licensing Service

   Intel Braidwood Technology

   Protect Audio Video Path

Intel(R) AMT State: Enabled

Link Status: Link down

Cryptography Support: Enabled

Last ME reset reason: Power up

System UUID: 03000200-0400-0500-0006-000700080009

MAC Address: 88-88-88-88-87-88

Configuration state: Not started

IPv4 Address: 0.0.0.0

IPv6 Enablement: Disabled

BIOS and GbE Config Lock: Disabled

Host Read Access to ME: Enabled

Host Write Access to ME: Enabled

SPI Flash ID #1:   1F4700

SPI Flash ID VSCC #1:   20152015

SPI Flash ID #2:   1F4700

SPI Flash ID VSCC #2:   20152015

BIOS boot State: Post Boot

Provisioning Mode: PKI

FWU Override Counter: Never

FWU Override Qualifier: Always

Local FWUpdate: Disabled

Secure FWUpdate: Enabled

OEM Id: 00000000-0000-0000-0000-000000000000

Remote PC Assist Service Registered: False

Remote PC Assist Service Enabler ID: 00000000-0000-0000-0000-000000000000

Remote PC Assist Service Enabler Description:

Capability Licensing Service: Enabled

Capability Licensing Service Status: Permit info not available

Override to RPAT-c SKU: Not Set

FW behavior on Flash Descriptor Override Pin-Strap: Halt

ntel(R) MEInfo Version: 6.0.0.1115

Copyright(C) 2005 - 2009, Intel Corporation. All rights reserved.

Intel(R) Manageability and Security Application code versions:

BIOS Version: 4.6.3

MEBx Version: 6.0.3.3

Gbe Version: 6.32.0

VendorID: 8086

PCH Version: 400004

FW Version: 6.0.0.1115

UNS Version: Not Available

LMS Version: 6.0.0.1095

MEI Driver Version: 6.0.0.1095

Wireless Hardware Version: Not Available

Wireless Driver Version: Not Available

FW Capabilities: 8314615

    Intel(R) Active Management Technology

    Intel(R) Anti-Theft Technology PC Protection

    Intel(R) Quiet System Technology

    Intel(R) Remote PC Assist Technology

    Intel(R) Capability Licensing Service

Intel(R) AMT State: Enabled

Link Status: Link up

Cryptography Support: Enabled

Last ME reset reason: Power up

System UUID: 03000200-0400-0500-0006-000700080009

MAC Address: 88-88-88-88-87-88

Configuration state: Completed

IPv4 Address: 10.10.10.62

IPv6 Enablement: Disabled

BIOS and GbE Config Lock: Disabled

Host Read Access to ME: Enabled

Host Write Access to ME: Enabled

SPI Flash ID #1:   1F4700

SPI Flash ID VSCC #1:   20152015

SPI Flash ID #2:   1F4700

SPI Flash ID VSCC #2:   20152015

BIOS boot State: Post Boot

Provisioning Mode: PKI

FWU Override Counter: Never

FWU Override Qualifier: Always

Local FWUpdate: Disabled

Secure FWUpdate: Enabled

Capability Licensing Service: Enabled

Capability Licensing Service Status: Permit info not available

FW behavior on Flash Descriptor Override Pin-Strap: Halt

6.3.2 Example 2

This example retrieves the current value of the Flash version

C:\ MEInfo.exe -feat "Local FWUpdat"

Disabled

6.3.3 Example 3

This example checks whether the computer has completed the setup and configuration process. If the parameter name or the value has a space, the value or name should be entered in quotes.

C:\ MEInfo.exe -feat "Setup and Configuration" –value "Not Completed"

Success. The values are identical

§

7　　*Firmware Update (FWUpdLcl)*

FW Update allows an end user, such as an IT administrator, to update the ME firmware without having to reprogram the entire flash device. It then verifies that the update was successful.

FWUpdate does not update the BIOS, GbE or Descriptor Region. It only updates the firmware code portion that Intel® provides on the OEM website. FWUpdate will update the entire ME code area.

The image file that the tool uses for the update is not the image file used to create the complete SPI firmware image file. A sample firmware image file for updating, Base_Corporate_ME_UPD.BIN, is located in the kit's NVM image folder. It only contain the ME code region and it can not be used to generate whole SPI image with FITC tool.

Please be aware that firmware update takes approximately 1-4 minutes to complete depending on the flash device on the system.

After FW update a host reset is needed to complete FW update. You can use – FORCERESET option to do this automatically

7.1　　Requirements

FWUpdLcl is a command-line executable that can be run on an Intel® AMT- enabled system that needs updated firmware.

Firmware can only be updated when the system is in an S0 state. Firmware updates are NOT supported in the S3/S4/S5 state.

Please be aware that firmware update takes approximately 1-4 minutes to complete, based on flash device.

If Intel® Anti-theft technology is enabled, a system restart must occur to complete the firmware update process.

Note: FWUpdate only supports upgrading firmware. Downgrading firmware is not supported. FWupdate tool does not work for Ignition firmware SKU.

7.2　　Dos Requirements

ME Firmware Local Update must be enabled in the MEBx.

7.3 Non-Secure Windows Requirements

ME Firmware Local Update must be enabled in the MEBx.

In the MEBx, Intel® AMT must be selected in the Manageability Feature Selection menu.

The ME Interface driver must be installed.

7.4 Secure Windows Requirements

ME Firmware Local Update must be enabled in the MEBx.

In the MEBx, Intel® AMT must be selected in the Manageability Feature Selection menu.

The ME Interface driver must be installed.

The Intel® AMT LMS must be installed.

7.5 Windows* PE Requirements

The ME driver must be installed in the Windows PE image.

The Windows PE image must be WMI-enabled.

7.6 Enabling and Disabling Local Firmware Update

Disabling Firmware Local Update in the MEBx prevents any updating of the firmware. However, even if Firmware Local Update is disabled, you can still enable updating the firmware for a limited number of times which can be done during manufacturing. To do this, configure the two variables Local FWU Override Counter and Local Firmware Override Qualifier to temporarily override the MEBx settings. These parameters can be modified by using FITC or FPT.

When Local FWU Override Counter has a value between 1 and 255, firmware updates are allowed even if updates are disabled in the MEBx settings. After the flash is programmed, each time the computer restarts it causes Local FWU Override Counter to be decremented. When Local FWU Override Counter reaches 0, firmware updates are no longer allowed if they are not enabled in the MEBx settings.

Note: The restart that takes place after the flash memory has been programmed also causes Local FWU Override Counter to be decremented. Therefore if you want to enable updating the firmware N times, you need to assign Local FWU Override Counter the initial value N+1.

If the Local FWU Override Counter is set to -1 and the Local Firmware Override Qualifier is set to 0, firmware updates are always allowed regardless of the settings in the ME BIOS extension

The following table shows the possible value combinations for the two variables. To enable local firmware updates, make sure both variables are assigned the correct values.

Table 11. Firmware Override Update Variables

	Local FWU Override Qualifier = 0 (zero)	Local FWU Override Qualifier = 1 (one)	Local FWU Override Qualifier = 2 (two)
Local FWU Override counter = 0 (zero)	Local Firmware Updates NOT Allowed	Local Firmware Updates NOT Allowed	Local Firmware Updates NOT Allowed
Local FWU Override Counter = -1 (minus one)	Local Firmware Updates Allowed	Local Firmware Updates NOT Allowed	Local Firmware Updates Allowed only until ME is configured
Local FWU Override Counter = 0<n<255	Local Firmware Updates Allowed	Local Firmware Updates Allowed	Local Firmware Updates Allowed

7.7 Usage of DOS Version

Note: In this section, <Image File> refers to an Intel-provided image file of the section of the firmware to be updated, not the image file used in FITC to program the entire flash memory.

To differentiate between the image files used for updating and those used for programming the entire flash memory, files used for FWUpdate include the string UPD in their file names.

Please be aware that firmware update takes approximately 1-4 minutes to complete, based on flash device.

FWUpdLcl.exe [-H|?] [-VER] [-EXP] [-VERBOSE] [-ALLOWSV]

      [-FORCERESET] [-OEMID] [-USER] [-PASS] [-WSMAN] [-DASH] [-EOI]

      [-TLS] [-GENERIC] [-HOST] [-CERT] [-HALTRCFG]

-H|?   Displays help screen.

-VER   Displays version information.

-EXP   Displays example usage of this tool.

-VERBOSE <file>   Display the debug information of the tool.

-ALLOWSV   Allows same version firmware updates.

-FORCERESET   Automatically Reboots system after update (if needed).

-OEMID <UUID>   OEM ID needed to perform firmware update.

-HALTRCFG   Halts remote configuration.

Image File—image file of the firmware to be updated. This image file is not the same image file used by FITC.

   -HaltRCFG – Halts all remote configuration network traffic and prevents remote configuration. The system can not be remotely configured until a local agent, such as Activator or ZTCLocalAgent, is run to initiate delayed provisioning mode. Only valid with firmware 4.1.3 and greater. The haltRCFG command can NOT be used as a command line argument while performing firmware update.

Image File—image file of the firmware to be updated. This image file is not the same image file used by the FITC.

7.8   Usage of Windows* Version

In this section, <Image File> refers to an Intel-provided image file of the section of the firmware to be updated, not the image file used in FITC to program the entire flash memory.

To differentiate between the image files used for updating and those used for programming the entire flash memory, files used for FWUpdate include the string UPD in their file names.

Please be aware that firmware update takes approximately 1-4 minutes to complete, based on flash device.

The executable can be invoked by:

FWUpdate.exe <Image File> – [options]

Image File—image file of the firmware to be updated. This image file is not the same image file used by the FITC.

Options—these options are only valid if the system has Intel® AMT selected in the MEBx. The options can be one or more of the following:

FWUpdLcl.exe [-H|?] [-VER] [-EXP] [-VERBOSE] [-ALLOWSV]

        [-FORCERESET] [-OEMID] [-USER] [-PASS] [-WSMAN] [-DASH] [-EOI]

        [-TLS] [-GENERIC] [-HOST] [-CERT] [-HALTRCFG]

-H|?   **Displays help screen.**

-VER   **Displays version information.**

-EXP   **Displays example usage of this tool.**

-VERBOSE <file>   **Display the debug information of the tool.**

-ALLOWSV   **Allows same version firmware updates.**

-FORCERESET   **Automatically Reboots system after update (if needed).**

-OEMID <UUID>   **OEM ID needed to perform firmware update.**

-USER <name>   **Admin user name. Must be used with the '-pass' option.**

-PASS <pass>   **Admin password. Must be used with the '-user' option.**

-WSMAN   **Optional. WSMAN is default even without this option.**

-DASH   **Optional. For DASH support   (otherwise WSMAN is used).**

-EOI   **Optional. For legacy support (otherwise WSMAN is used).**

-TLS   **Optional. to be used in TLS work mode.**

-GENERIC   **Perform the update through MEI without credentials.**

-HOST <name>   **The hostname of the AMT firmware.**

-CERT <cert>   **User certificate for Mutual Authentication.**

-HALTRCFG   **Halts remote configuration.**

-HaltRCFG – Halts all remote configuration network traffic and prevents remote configuration. The system can not be remotely configured until a local agent, such as Activator or ZTCLocalAgent, is run to initiate delayed provisioning mode. Only valid with firmware 4.1.3 and greater. The haltRCFG command can NOT be used as a command line argument while performing firmware update.

7.9 Examples

7.9.1 Example 1

FWUpdLcl.exe Base_Corporate_BYP_ME_UPD.BIN –user Admin –pass Admin@98 –TLS –host Cert_Name

The above will update the local firmware using a TLS connection to the firmware. The certificate name Cert_Name matches the certificate name provided in the firmware.

7.9.2 Example 2

FWUpdLcl.exe Base_Corporate_BYP_ME.BIN –user admin –pass Admin@98
Error: Bad seek
Error: failed to parse image file

The above error message is seen if the wrong firmware binary file is used. When updating the firmware, the correct file for this tool name contains the string UPD in the filename.

7.9.3 Example 3

FWUpdLcl.exe -haltRCFG

Calling the haltRCFG option will halt all remote configuration traffic and prevent remote configuration. The haltRCFG command can NOT be used as a command line argument while performing firmware update.

§

8   Update parameter tool (UPdParam)

8.1 Purpose of the tool

UpdateParam tool is used to change certain ME firmware parameters (both AMT and Kernel) even after the global valid bit being set and description region is locked. This tools will work only when BIOS does not send EOP(End Of Post) message.

8.2 Usage of the tool

UpdParam.exe [-?] [-h] [-f] [-v] [-r] [-u] [-verbose]

H|? :   Display help screen

F: Input USB file name

V: Override MEBx Admin password

R :   Global reset

U: Partial Unprovisioning

VERBOSE   Display the debug information of the tool

This tool uses a binary file as an input. This binary file gets created using USB Utility. Below is a list of parameter that could be set in a USB file utility command to generate a binary file.
Syntax:

To create a USB file:

```
 USBfile -create <usb output file name> <current MEBx password>
        <new MEBx password> [-v 1|2|2.1] [-amt]
        [-v1file <version 1 outfile>]
        [-dns <DNS suffix>] [-fqdn <prov server fqdn>]
        [-ztc 0|1]
        [-dhcp 0|1]
        [-fwu 0|1]
        [-pm 0|1]
        [-fwuq 0|1|2]
        [-pp <16 byte GUID>
        [-pspo <port number>
```

```
[-psadd <ip addr>
[-ito <4 byte of idle time out>
[-gen <num of records>]
[-xml <xml file name>]
[-pid <pid> -pps <pps>]
[-hash <cert file name> <friendly name>]
[-redir <n>]
[-s4p <StacticIPv4Params>]
[-hostname <hostname>]
[-domname <domain name>]
[-vlan <0|1-VlanTag>]
[-passPolicyFlag    <0|1|2>]
```

Where,

-v 1|2|2.1: the setup file version, 2.1 by default
-v1file <version 1 outfile>: creates a version 1 setup file
-amt: this will set the manageability selection value to AMT
-dns <DNS suffix>: sets the PKI dns suffix name (up to length 255)
-fqdn <prov server fqdn>: string up to length 255
-ztc 0|1: disable/enable PKI Configuration
-dhcp 0|1 disable/enable DHCP
-fwu 0|1: disable/enable Firmware local update
-pm 0|1: Enterprise/SMB provisioning mode
-fwuq 0|1|2: Always|Never|Restricted Firmware Update Qualifier
-pp <GUID> set the power package ,GUID should be in network order
-pspo <port number> provision server port number
-psadd <ip addr> :ip address for provision server e.g 123.222.222.121
-ito <4 byte of idle time out> : 4 char of idle time out
-gen <n> : number of records to create
-xml <xml file name> : configuration xml file
-pid <pid> -pps <pps>: a psk pair - this is ignored if -gen was chosen
-hash <certificate file name> <friendly name>: to compute and add the
       hash of the given root certificate file. Up to three certficate
       hashes may be specified.
-redir <n>:
       This is an integer that is calculated as follows:
        bit 0 : 1 (Enable) or 0 (Disable) - SOL feature
        bit 1 : 1 (Enable) or 0 (Disable) - IDER feature
        bit 2 : 1 (Enable) or 0 (Disable) - Username/password
               authentication type of the SOL/IDER in the ME FW
-s4p <localHost:SubnetMask:GatewayAddr:DNSaddr:SecondaryDNSaddr>
 :e.g 10.0.0.1:255.255.255.0:10.0.0.2:10.0.0.3:10.0.0.4
 Note: DHCP flag should be disable.
-hostname <hostname> :ASCII reprasentation of host name max length 63
-domname <domain name> : max length of domain name is 255
-vlan <0|1-VlanTag(1-4096)> : VlanStatus enable/disable e.g 0-4011

-passPolicyFlag <0|1|2> : Default/block in post/always open

More details on how to use the USB file utility can be found using the help command in the USB file utility. Once all the parameters that the user needs to change are set (along with the current MEBx password) usbfile.exe creates a binary file.

For example if the user sets the command as

Usbfile.exe –create test.bin Admin Admin@98 (Supposing the System current MEBx password is Admin)

Running the Usbfile.exe, the above command will create a binary file named "test.bin" setting the new password for MEBx to Admin@98

Once the binary file is created it is used by the UpdateParam tool as in input.

There are few requirements the user needs to follow in order to use the binary file created by this usbutility.exe.

- This binary file needs to contain the current MEBx password.
- For this tool (UpdateParam tool) to run it needs to be in either pre-boot or in-boot mode

  Pre boot: just flashed the image, not changed the password yet

  In boot: once the user have changed the password and entered MEBx

- User also needs to make sure that BIOS will not send End Of Post to ME

8.3 Output

If the binary file contains the right MEBx password, binary file proceeds further to make appropriate changes to the settings returning (SUCCESS/ FAIL) status for each of the parameters that are in the binary file otherwise the tool will exit out with the error code and error message. (Below is a screenshot of the tool when the pwd entered is incorrect)

```
------------------------------------------------
Intel(R) UpdParam Version:        6.0.0.9290
Copyright (c) 2007-2009, Intel Corporation. All rights reserved.

------------------------------------------------
Chipset: Ibexpeak.
Validating Password... Failed.

Error 3037: The CurrentMEBx password is invalid.
```

Once the password validation is successfully completed tool changes the rest of the parameters as listed in the bin file. If there is a failure changing/updating any of the parameter there is an error code and error message returned associated with the failure.

```
------------------------------------------------
Intel(R) UpdParam Version:        6.0.0.9290
Copyright (c) 2007-2009, Intel Corporation. All rights reserved.

------------------------------------------------
Chipset: Ibexpeak.

Validating Password... Success.
Updating Local Firmware Update Qualifier... Success.
Updating PID/PPS... Success.
Note: No change in ZTC status required.  Same as input.
Updating PID/PPS... Success.
Updating PKI DNS Suffix... Failed..
Error 3: Command is not permitted in current operating mode
Updating Config Server FQDN... Failed..
Error 1: AMT device internal error
Updating SOL/IDER Configuration... Success.
Setting FW update Parameter... Failed.
Setting Host Name... Success.
Setting Domain Name... Success.
Setting Idle Timeout... Success.
Setting Provisioning Mode... Success.
Setting ProServer Port Parameter... Success.
Setting IPv4 Parameters... Success.
Changing Password... Success.
|
```

Note: all the error messages are displayed in Red and any warnings are displayed in yellow.

This tool uses MEI communicate to different components of the ME. Therefore, the tool also returns MEI status.

A log file is also created with details on all the steps run. The log file can be found in the same location where the executable is run from.

8.4 ME parameters that can be changed by UpdParam tool:

- Current MEBx password
- New MEBx password
- Manageability Feature selection
- Firmware Local update
- Firmware update qualifier
- Power package
- PID
- PPS
- PKIDNSSuffix
- ConfiServerFQDN
- ZeroTouchSetupEnabled
- HostName
- DomainName
- DHCP
- Idle Timeout
- Provisioning Mode
- Provisioning Server Address
- Provisioning server port
- StaticIPv4Parameters
- VLAN
- Host IF SOL enabled
- Host If IDER enabled
- Updating Pre Installed Certificate settings
- Updating Customized certificate hash entries
- Adding Custom certificate entries.

8.5 Examples:

UpdParam –f <filename>

    This usage inputs the binary file and updates the parameter

UpdParam –f <filename> -v <CurrentMebxPwd>

    This usage inputs a binary file with mebx current password entered at the command prompt

UpdParam –f <filename> -v <CurrentMebxPwd> -u

This usage input binary file with mebx current password entered at the command prompt and also an option is entered to do partial unprovisioning.

Updparam –r

This usage performs a global reset

Updparam –h

This usage displays the help screen

§

Appendix A   Fixed offset Variables

This appendix only covers fixed offset variables that are directly available to FPT and FPTW. A complete list of fixed offset variables can be found in the Firmware Variable Structures for Intel® Management Engine (Document number 24571). All of the fixed offset variables have an id and a name. The "-fov" option will display a list of the ID and their respective name. The variable name must be entered exactly as displayed below.

*Note: This table is for reference use only and will be updated later.

Table 12. Fixed Offset Item Descriptions

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value
Non-Application Specific Fixed Offset Item Descriptions				
ME State Lock	0x0001	Locks ME State to enable. Cannot be used to lock the state to ME disabled. This value will only result in the State being locked if the current state is set to enabled or ME State Control FOV is set to Enabled. **The end-user can no longer disable Manageability Engine in MEBx.**	1	Default: **0xFF** Locked: **0x02** UnLock: **0x00**
ME State Control	0x0002	Sets ME state to disable or enable. Two ME resets are required for this item to take effect. Once ME is disabled the only way to enable it is to use MEBx. Setting the FOV to enable when ME is in the disabled state will not cause ME to become enabled.	1	Disabled: **0x00** Enabled: **0x01**

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value
MEBx Password	0x0003	Overrides the MEBx default password. It must be at least eight characters and not more than 32 characters in length. All characters must meet the following: ASCII(32) < = char < = ASCII(126) Cannot contain these characters: , : " Must contain for complexity: a. At least one Digit character (0 - 9) b. At least one 7-bit ASCII non alpha-numeric character above 0x20 (e.g. ! $;) c. Both lower-case and upper case Latin d. underscore and space are valid characters but are not used in determination of complexity **See section 2.7 for format and strong password requirements.**	8<=N<=32	
ME Power Features Lock	0x0004	Lock ME Power Features to current state	1	Locked: **0x02** UnLock: **0x00**
Default Power Package	0x0005	Default Power Package (Desktop): Pkg1 - ON in S0 Pkg2 - ON in S0, ME Wake in S3, S4-5 Default Power Package (Mobile): Pkg1 - ON in S0 Pkg2 - ON in S0, ME Wake in S3, S4-5 (AC-Only)	1	Package 1: 0x01 Package 2: 0x02
Local FW Update Enable	0x0006	Enable or disable Local FW Update	1	Disabled: **0x00** Enabled: **0x01**
Local FW Update Override Qualifier	0x0007	Value of the Qualifier	1	Always: **0x00** Never: **0x01** Restricted: **0x02**
Local FW Update Override Counter	0x0008	Value of the Counter	1	Never: **0x00** Allow N boot cycles: **0<N<255** Always: **0xFF**

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value
AMT MEBx Password Rule Flag	0x0009	Controls Password Manager ability to allow AMT to set MEBx password from remote.	1	0 = Do not allow the MEBx password to be from remote 1 = Allow the MEBx password to be set from remote
OEM Permanent Disable	0x000A	UINT32 (little endian) value. This controls what features are permanently disabled by OEM. See 3.6.1 andTable 5 for more details. If a feature is grayed out in Table 5 for that target HW SKU, then the firmware will disregard that selection. Notes: 1. User must set all non-reserved bits to the value they want. There is NO ability to change features one at a time. This FOV sets OEM Permanent Disable for ALL features. 2. This will not enable functionality that is not capable of working in the target hardware SKU. Example: This will not make Intel⁰ Q57 capable of using Intel® IPT. Please see the respective Firmware Bringup Guide for a list of what features are capable with what firmware bundle and Hardware SKU of Intel 5 Series Chipset. Examples: • Intel Q57 with Intel® AMT, Intel® QST, KVM and PAVP 1.5 enabled: Bits: 0,2,4,12, 18,21 set to '1' (0x241015) • Intel QM57 with disabling Intel® AMT, PAVP 1.5 enabled: Bits: 12 set to '1' (0x0) • Intel HM57 with Intel® IPT, RPAT-C, PAVP 1.5 and Intel® QST Enabled. KVM and TLS disabled: Bits: 2,4,12, 13 (0x3014)	4	Feature Capable: 1 Feature Permanently disabled: 0 See bit table below: 1. For corporate SKUs (Intel® Q57, Intel® QM57, Intel® QS57 and Intel® 3450) bits 0 and 2 need to be both set to '1' to allow for Intel® AMT to work. 2. KVM (bit 18) should only be set to '1' when Manageability Application (bit 2) is set to '1'. If using a Corporate SKU, then Intel® AMT (bit 0) must also be set to '1'. Reserved bits should be set to 0.

Bit table within Expected Value column (OEM Permanent Disable):

Bit	Description	Notes
31:22	Reserved	
21	TLS	
20:19	Reserved	
18	KVM	2
17:14	Reserved	
13	Intel IPT	
12	PAVP 1.5	
11:5	Reserved	
4	Intel® QST	
3	Intel RWT	
2	Manageability Application	1
1	Reserved	
0	Intel® AMT	1

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value
Feature Ship State	0x000B	UINT32 (little endian) value. This controls what features are enabled or disabled. These features may be enabled/disabled by mechanisms such as MEBx or provisioning. This setting is only relevant for features NOT permanently disabled by the OEM Permanent Disable. Notes: 1. User must set all non-reserved bits to the value they want. There is NO ability to change features one at a time. 2. This will not enable functionality that is not capable of working in the target hardware SKU. Example: This will not make Intel® Q57 capable of using Intel® IPT. Please see the respective Firmware Bringup Guide for a list of what features are capable with what firmware bundle and Hardware SKU of Intel 5 Series Chipset. Examples: • Intel Q57 with Intel® AMT, Intel® QST, ship enabled: Bits: 2,4 set to '1' (0x14) • Intel QM57 with disabling Intel® AMT, Bits: none set to '1' (0x0) • Intel HM57 with Intel® IPT, RPAT-C, Enabled. Bits: 2, 13 (0x2004)	4	Feature Enabled: 1 Feature Disabled: 0 | Bit	Description	Notes
---	---	---
31:14	Reserved	
13	Intel IPT	
11:5	Reserved	
4	Intel® QST	
3	Intel RWT	
2	Manageability Application	
1:0	Reserved		 All other bits are reserved. Reserved bits should be set to 0.

Intel® AMT Related Fixed Offset Item Descriptions

PID	0x2001	A 64 bit quantity made up of ASCII codes of some combination of 8 characters – capital alphabets (A–Z), and numbers (0–9). **Must be set along with PPS**.	8	Please see the PSK algorithm section on how to generate a valid PID.
PPS	0x2002	A 256 bit quantity made up of ASCII codes of some combination of 32 characters – capital alphabets (A–Z), and numbers (0–9). **Must be set along with PID**.	32	Please see the PSK algorithm section on how to generate a valid PPS.
Manufacturing Test Counter	0x2006	MeManuf Full Test Counter	1	Valid: between **0x00 and 0xFF**

Fixed offset Variables

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value
Idle Timeout – ME	0x2008	UINT16 representing the time in minutes for the Idle Timeout	2	Value **0x0000 < n <:0xFFFF**
Remote Configuration Enabled	0x2009	Remote Configuration Enable setting	1	Enabled:**0x01**
Config Server FQDN Name	0x200A	Provisioning Server Domain Name. Null terminated sting	1<=N<= 255	Valid Domain Name
Certificate Hash Entry 1	0x200B	Cert Hash Data. See Certificate Hash Entry Structure definition **Note: If the platform is un-configured the Certificate Hash will be deleted.**	55 => 83	Valid Certificate Hash Entry
Certificate Hash Entry 2	0x200C	Cert Hash Data. See Certificate Hash Entry Structure definition **Note: If the platform is un-configured the Certificate Hash will be deleted.**	55 => 83	Valid Certificate Hash Entry
Default Hash Entry 19	0x200D	Cert Hash Data. See Certificate Hash Entry Structure definition **Note: If the platform is un-configured the Certificate Hash will be reverted to 'active'.**	55 => 83	Valid Certificate Hash Entry
MEBx Password Change Policy	0x200E	The policy that controls MEBx password change over the network interface. Policy 0 – change allowed only if the password is still default Policy 1 – change allowed only during Setup and Configuration Policy 2 – change always allowed	1	Policy 0: **0x00** Policy 1: **0x01** Policy 2: **0x02**

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value

Remote Connectivity Service Enabler ID	0x200F	A unique identifier which will be used by Intel® AMT to indicate to Remote Connectivity Service who the reseller of the PC is	16	"The following structure definition is used to set a 'RCS Enabler Id':
				
				/** Defines a globally unique ID (GUID) */
				
				typedef struct
				
				{
				
				  UINT32 Data1; /**< DWORD 1 */
				
				  UINT16 Data2; /**< WORD 2 */
				
				  UINT16 Data3; /**< WORD 3 */
				
				  UINT8 Data4[8]; /**< BYTE 4 Array */
				
				} GUID, *PGUID;
				
				All zeros is invalid
				
				All ff's is invalid

Remote Connectivity Service Enabler Name (OEM Description)	0x2010	A human readable string to describe the party represented by uiEnablerId.	60	Remote Connectivity Service Enabler Name (OEM Description)
Remote Connectivity Service Capability	0x2012	Determines whether the platform is RCS capable.	1	Intel Remote Connectivity Service Capability flag RCS Capable : **0x01** RCS Not-Capable **0x00**
KVM settings	0x2014	KVM feature settings	1	b11 – Enabled b10 - Disabled Bit mask: Bits 7:0 Bit 0..1 - Kvm Host I/F enabled (KvmHostIFEnabled) Bit 2..3 - Opt in (user consent) policy for change from PTNI (OptinPTNIEnabledPolicy) Bit 4..5 - Opt in (user consent) enabled (OptinPTNIEnabledPolicy) Bit 6..7 - reserved
Remote Connectivity Service HW Button	0x2015	RCS HW Button. If enabled - chassis intrusion alerts will be translated as RCS trigger.	1	Chassis Intrusion:0x01 RCS Trigger: 0x02
Provisioning Period	0x2016	Provisioning Period setting	1	Time value between 0 to 255 hours. 0 - delayed provisioning

Fixed Offset Name	Fixed Offset ID	Description	Data Length (inBytes)	Expected Value
USBr Settings	0x2017	USBr feature settings	1	b11 – Enabled b10 - Disabled Bit mask: Bits 7:0 Bit 0..1 - EHCI 1 enabled (EHCI1Enabled) Bit 2..3 - EHCI 2 enabled (EHCI2Enabled) Bit 4..7 - reserved At least one of the EHCIs should be enabled. This is not required but recommended.
CLS Related FOV Item Descriptions				
Unlocking MTP	0x5001	8 byte key value used to unlock the Manufacturing Test Permit after Global Lock is Set	8	A valid 8 byte MTP unlocking key. Cannot be all 0x00 or 0xFF.
Unlocking SMTP	0x5002	8 byte key value used to unlock the Service Manufacturing Test Permit after Global Lock is Set	8	A valid 8 byte SMTP unlocking key. Cannot be all 0x00 or 0xFF.
AT-p Related FOV Item Descriptions				
AT-p FW Flash Protection Override Policy Hard GPIO33	0x6001	Indicates whether Hard-GPIO-33 is allowed, and under what conditions.	1	Always Allowed: **0x01** Allowed when AT-p NOT provisioned: **0x02**
AT-p FW Flash Protection Override Policy Soft GPIO33	0x6002	Indicates whether Soft-GPIO-33 is allowed, and under what conditions.	1	Always Allowed: **0x01** Allowed when AT-p NOT provisioned: **0x02**

Appendix B Tool Error message

Error code and message in this document will be changed after Alpha

B.1 Common Error code for all tools

Error Code	Error Message	Explanation	Suggestion
SPI Common errors			
7	Invalid descriptor region.		
8	%s Region does not exist. Where %s can be the followings: DESCRIPTOR BIOS ME GBE PDR		
154	Hardware sequencing failed. Make sure that you have access to target flash area!		
202	The host CPU does not have read access to the target flash area. To enable read access for this operation you must modify the descriptor settings to give host access to this region.		
255	An attempt was made to read beyond the end of flash memory		
405	There are no supported SPI flash devices installed. Please check connectivity and orientation of SPI flash device.		
Communicatio n Common errors			
8196	MEI message recieve buffer memory allocation failed		

Error Code	Error Message	Explanation	Suggestion
8199	Could not issue %s command message	Where %s can be the followings: Get FWU Version Get FWU Info Get FWU Feature State Block LAN Unblock LAN Intel(R) ME Kernel Test Intel(R) AMT Extended Test Intel(R) AMT Partial Test Intel(R) AMT Full Test Intel(R) AMT Graceful Test Intel(R) AMT Test Result Intel(R) ME Kernel Test Result Block Intel(R) AMT Full Test Get Intel(R) AMT Test Counter	
8203	Unexpected result in %s command response	Where %s can be the followings: Get FWU Version Get FWU Info Get FWU Feature State Block LAN Unblock LAN Intel(R) ME Kernel Test Intel(R) AMT Extended Test Intel(R) AMT Partial Test Intel(R) AMT Full Test Intel(R) AMT Graceful Test Intel(R) AMT Test Result Intel(R) ME Kernel Test	

Error Code	Error Message	Explanation	Suggestion
		Result Block Intel(R) AMT Full Test Get Intel(R) AMT Test Counter	
8213	Requesting MEI receive buffer size is too small		

B.2 Firmware Update Errors

Error Code	Error Message	Explanation	Suggestion
0	Success		
1	An internal error to the AMT device has occurred	haltrcfg related	
2	AMT Status is not ready	haltrcfg related	
3	Invalid AMT Mode	haltrcfg related	
4	An internal error to the AMT device has occurred	haltrcfg related	
8193	Intel ® ME Interface : Cannot locate ME device driver		
8199	Intel ® ME Interface : ME Device not ready for data transmission		
8204	Intel ® ME Interface : Unsupported message type		
8703	PLEASE REBOOT YOUR SYSTEM. Firmware update cannot be initiated without a reboot.	You may try to update firmware twice without a reboot	Reboot the system

Tool Error message

Error Code	Error Message	Explanation	Suggestion
8704	Firmware update operation not initiated due to a SKU mismatch		
8705	Firmware update not initiated due to version mismatch		
8706	Firmware update not initiated due to integrity failure or invalid FW image		
8707	Firmware update failed due to an internal error		
8707	Firmware update failed due to an internal error. Firmware returns SAL notification error, Please try after ME-reset or re-flashing the ME image.		
8707	Firmware update failed due to an internal error. Firmware returns Audit policy error, Please try after ME-reset or re-flashing the ME image		
8707	Firmware update failed due to an internal error. Firmware failed to create fault tolerant partition, Please try after ME-reset or re-flashing the ME image		
8708	Firmware Update operation not initiated because a firmware update is already in progress		
8710	Firmware update tool failed due to insufficient memory		
8710	Firmware update failed due to insufficient memory		
8712	Firmware update failed due to authentication failure		
8713	Firmware update not initiated due to an invalid FW image		
8713	Firmware update not initiated due to an invalid FW image header		
8714	Firmware update not initiated due to file <file> open or read failure		
8714	Firmware update not initiated due to file open or read failure		
8715	Firmware update tool failed to connect iAMT through LMS, due to a HTTP operation failure		
8715	Firmware update tool failed to connect iAMT through LMS, due to a HTTP operation failure, Please verify the inputs (host, user, password, certificate, work mode etc).		

Error Code	Error Message	Explanation	Suggestion
8716	Invalid usage		
8716	Invalid usage, -allowsv switch required to update the same version firmware		
8717	Firmware update not initiated due to invalid hostname specified		
8718	Update operation timed-out; cannot determine if the operation succeeded		
8719	Firmware update cannot be initiated because Local Firmware update is disabled		
8720	Firmware update cannot be initiated because Secure Firmware update is disabled		
8722	Cannot receive the current version from the firmware after update		
8723	No Firmware update is happening		
8724	Update finished but version mismatch after the update		
8725	Failed to receive last update status from the firmware		
8727	Firmware update tool failed to get the firmware parameters		
8728	Firmware update iAMT communication failed, Failed to find certificate <certName> in certificate store		
8728	Firmware update iAMT communication failed, Failed to set HTTP certificate options <lastError>: <errMsg>		
8728	Firmware update iAMT communication failed, Failed to find certificate names		
8728	Firmware update iAMT communication failed, Failed to open system certificate store <lastError>: <errMsg>		
8728	Firmware update iAMT communication failed, HTTP request failed: Certificate rejected		
8734	Firmware update iAMT communication failed, WSMAN not supported		
8740	Unsupported Operating System		
8743	Unknown or Unsupported Platform		
8744	OEM ID verification failed		
8745	Invalid UUID provided with the OEMID switch		

Error Code	Error Message	Explanation	Suggestion
8745	Firmware update cannot be initiated because the OEM ID provided is incorrect		
8746	Firmware update not initiated due to invalid image length		
8747	Firmware update not initiated due to an unavailable global buffer		
8748	Firmware update not initiated due to invalid firmware parameters		

B.3   MEManuf Errors

Error Code	Error Message	Explanation	Suggestion
9249	Intel® ME internal communication error (AMT)	Execution error for Intel® ME firmware	1) G3 2) RTC reset and G3 3) Reflash ME region and G3
9250	Communication error between application and Intel® ME	Application is trying to communicate with Intel® ME firmware, and get an error result because Intel® ME is disabled or not available	Verify Intel® ME is enabled, and communication between application and Intel® ME is working
9251	Fail to create verbose log file %s Where %s is the log file name user specified	No write access to the media, or out of disk space	Make sure enough space on the media and/or have write access
9252	Invalid command line option(s)	Invalid command line option(s)	See MEManuf -h or -exp for detail
9254	Unsupported OS	Run application on a wrong OS environment	Change the tool version or change OS environment
9255	Cannot run the command since Intel® AMT is not available	These options are not available with AMT disabled or not avaiable	Make sure Intel® AMT module is supported and enabled

Error Code	Error Message	Explanation	Suggestion
9256	Communication error between application and Intel® ME module (FW Update client)	Application is trying to communicate with Intel® ME firmware, and get an error result because Intel® ME is disabled or not available	Verify Intel® ME is enabled, and communication between application and Intel® ME is working
9257	Internal error (Could not determine FW features information)	Firmware failed to return an application query	Flash Intel® ME region and perform G3
9261	Hibernation isn't supported by the OS, Intel(R) AMT test cannot run	S4 option is not supported if OS doesn't support hibernation	Running this option in Windows with hibernation feature on
9262	Intel® ME kernel test result has error code %d	Intel® ME Built-in Self Test failed. Detail error is broken down in human readable format	See detail errors above, and apply correction method accordingly
9263	Intel(R) AMT extended test result has error code %d	Intel® AMT Extended Built-in Self Test failed. Detail error is broken down in human readable format	See detail errors above, and apply correction method accordingly
9264	Intel® AMT test reports %s Where %s can be the followings: an internal error a flash read/write error an error sending commands on the SMBus a power down error a BIOS error a wireless link related error a MC error	Same as the error message	See detail errors, and apply correction method accordinly
9265	No Intel® AMT test result to retrieve	Before use MEManuf -r to retrieve test result, user needs to run MEManuf first with Intel® AMT enabled	Run MEManuf before run MEManuf -r
9266	Internal error (Block full test command could not be performed)	Internal error	

Error Code	Error Message	Explanation	Suggestion
9267	Fail to establish a communication with SPI flash interface	Missing (Windows) driver	Check (Windows) driver is functioning
9268	Fail to load vsccommn.bin	File cannot be found, or corrupted	Make sure vsccommn.bin is in the same folder as MEManuf.exe
9269	Zero flash device found for VSCC check	Internal error	
9270	Fail to load driver (PCI access for Windows) Tool needs to run with an administrator priviledge account.	Fail to load driver	Copy the <third party driver files> to the tools folder, and reboot
9271	Flash ID 0x%06X Intel® ME VSCC mismatch Programmed value of 0x%X doesn't match the recommended value of 0x%X See PCH SPI programming Guide for more details	Wrong VSCC configured value	Refer to PCH SPI programming guide
9272	Flash ID 0x%06X ME VSCC value didn't find recommanded value	Unrecognized SPI flash part	Make sure user use the latest version of vsccommn.bin
9273	Intel (R) VE is disabled by PCH SoftStrap	VE is disabled by softstrap	Make sure to enable VE softstrap
9275	Current Intel® ME power policy doesn't support S4 option	S4 option only available at power policy that supports Intel® ME working at system sleep state	Modify Intel® power policy accordingly
9276	Fail to read FW Status Register value 0x%X	Intel® ME is disabled or not available	1. Verify Intel® ME is properly enabled in BIOS 2. If error persists, perform G3 3. If error persists, clear CMOS and perform G3 4. If error still persists, flash Intel® ME region and perform G3

Error Code	Error Message	Explanation	Suggestion
9277	Intel (R) VE internal error	VE internal error	1) G3 2) RTC reset and G3 3) Reflash ME region and G3
9278	Cannot locate hardware platform identification This program cannot be run on the current platform. Unknown or unsupported hardware platform or A %s hardware platform is detected This program cannot be run on the current platform. Unknown or unsupported hardware platform Where %s is the offical name of the hardware platform		
9285	Unsupported command line option(s) for Ignition firmware	Ignition related	
9286	Ignition firmware check was not successful	Ignition related	
9291	Access Ignition firmware flash device failure	Ignition related	
9292	The SKU does not have any test assigned to be run -AMT is not available on this firmware -NAND is not available on this firmware -S5 option is only available with Intel(R) AMT test -S4 option is only available with Intel(R) AMT test -S0 option is only available with Intel(R) AMT test -S4 Inte(R) AMT test only runs under Windows	Same as the error	Invalid option for currently configuration
9295	Intel® AMT BIST test failed, error 0x%x returned	Intel® AMT module test failed	
9296	MEManuf Test Failed Use <VERBOSE> option for more details	There are MEManuf test failures	Use -verbose to see more detail
9297	Intel(R) NAND needs to be enabled to perform the test	Same as the error	Make sure Intel® NAND is available and enabled on the platform

Error Code	Error Message	Explanation	Suggestion
9298	This command line option is only available for Ignition firmware	This option is only available for Intel® ME Ignition SKU	Use this command on Intel® ME Ignition SKU

B.4   MEInfo Errors

Error Code	Error Message	Explanation	Suggestion
9250	Communication error between application and Intel® AMT module (Manageability client)	Application is trying to communicate with Intel® AMT firmware, and get an error result because Intel® AMT is disabled or not available	Verify Intel® AMT is enabled, and communication between application and Intel® ME is working
9251	Communication error between application and Intel® AMT module (PTHI client)	Application is trying to communicate with Intel® AMT firmware, and get an error result because Intel® AMT is disabled or not available	Verify Intel® AMT is enabled, and communication between application and Intel® ME is working
9254	Invalid command line option(s)	Invalid command line option(s)	See MEInfo -h or -exp for detail
9255	Fail to read FW Status Register value 0x%X	Intel® ME is disabled or not available	1. Verify Intel® ME is properly enabled in BIOS 2. If error persists, perform G3 3. If error persists, clear CMOS and perform G3 4. If error still persists, flash Intel® ME region and perform G3
9256	Unsupported OS	Run application on a wrong OS environment	Change the tool version or change OS environment

Error Code	Error Message	Explanation	Suggestion
9257	Fail to create verbose log file %s Where %s is the log file name user specified	No write access to the media, or out of disk space	Make sure enough space on the media and/or have write access
9258	Communication error between application and Intel® ME module (FW Update client)	Application is trying to communicate with Intel® ME firmware, and get an error result because Intel® ME is disabled or not available	Verify Intel® ME is enabled, and communication between application and Intel® ME is working
9259	Internal error (Could not determine FW features information)	Firmware failed to return an application query	Flash Intel® ME region and perform G3
9260	Cannot locate hardware platform identification This program cannot be run on the current platform. Unknown or unsupported hardware platform or A %s hardware platform is detected This program cannot be run on the current platform. Unknown or unsupported hardware platform Where %s is the offical name of the hardware platform		
9267	Cannot use zero as SPI Flash ID index number	Zero index number is invalid	Type MEInfo to see the correct SPI Flash IDs
9268	Couldn't find a matching SPI Flash ID	Invalid SPI Flash ID index number has been supplied in feature name	Type MEInfo to see the correct SPI Flash IDs
9269	Access to SPI Flash device(s) failed	Communication between application and SPI device(s) failed	Verify hardware connection between CPU and SPI device(s) is working properly
9270	Fail to load driver (PCI access for Windows) Tool needs to run with an administrator priviledge account.	Fail to load driver	Copy the <third party driver files> to the tools folder, and reboot

Error Code	Error Message	Explanation	Suggestion
9271	Invalid feature name XXXXX Where XXXXX is the feature name	XXXXX is not a valid feature name	Please refer to Tools User Guide for supported feature names
9272	XXXXX feature was not available Where XXXXX is the feature name	XXXXX is disable or not available	1) Verify the feature is in the firmware SKU 2) Verify the feature is enabled 3) Verify the driver is properly installed
9273	XXXXX actual value is - YYYYY Where XXXXX is the feature name Where YYYY is the feature value	Wrong value with the specified feature	Please refer to Tools User Guide for supported feature name/value

B.5 FPT Errors

Error Code	Error Message	Explanation	Suggestion
0	Success		
1	Memory allocation error occurred	Make sure there is enough memory in the system	
200	Invalid parameter value specified by the user. Use -? Option to see help.	Check the command line arguments supported by using the "-?"	
201	FPT.exe cannot be run on the current platform. Please contact your vendor.		
202	Confirmation is not received from the user to perform operation.	User input required	
203	Flash is not blank. Data <data> found at address <address>.	Attempt to erase the device again	
204	Data verify mismatch found at address <address>.	Reprogram the device	
205	Failure. Unexpected error occurred	Please file a sighting	

Error Code	Error Message	Explanation	Suggestion
206		PDR region exists	
240	Access was denied opening the file <file>	Check the permissions for the file	
241	Access was denied creating the file <file>	Check the permissions for the file	
242	An unknown error occurred while opening the file <file>	Verify the file is not corrupt	
243	An unknown error occurred while creating <file>	Verify the file is not corrupt	
244	<name> is not a valid file name.	Check the filename	
245	<file> file not found	Check file location	
246	Failed to read the entire file into memory. File: <file>	Check system memory. Verify the file is not corrupt	
247	Failed to write the entire flash contents to file	Check system memory	
248	<file> file already Exists	Delete the file that already exist	
249	The file is longer than the flash area to write	Check file size	
250	The file is smaller than the flash area to write	Check file size	
251	Length of image file extends past the flash area	Check file size	
252	Image file <file> not found	Check filename	
253	<file> file does not exist	Check filename	
254	Not able to open the file <file>	Check filename	
255	Error occurred while reading the file <file>.	Check filename	
256	Error occurred while writing to the file <file>	Check filename	
280	Failed to disable write protection for the BIOS space!	Verify BIOS does not have write protection enabled	
281	The Enable bit in the LPC RCBA register is not set. The value of this register cannot be used as the SPI BIOS base address		
282	Failed to get information about the installed flash devices	Check descriptor region access settings	

Error Code	Error Message	Explanation	Suggestion
283	Unable to write data to flash. Address <address>.	Check descriptor region access settings	
284	Fail to load driver (PCI access for Windows). Tool needs to run with an administrator priviledge account.		
320	General Read failure	Attempt the command again. If symptom persists file a sighting	
321	The address <address> is outside the boundaries of flash area	Check address	
360	Invalid Block Erase Size value in <file>.	Check fparts.txt or its equivalent file	
361	Invalid Write Granularity value in <file>	Check fparts.txt or its equivalent file	
362	Invalid Enable Write Status Register Command value in <file>	Check fparts.txt or its equivalent file	
363	Invalid Chip Erase Timeout value in <file>	Check fparts.txt or its equivalent file	
400	Flash descriptor does not have correct signature	Verify file is not corrupt	
401	An error occurred reading the flash mapping data	Check SPI device	
402	An error occurred reading the flash components data	Check SPI device	
403	An error occurred reading the flash region base/limit data	Check SPI device	
404	An error occurred reading the flash master access data	Check SPI device	
405	An error occurred reading the flash descriptor signature	Check SPI device	
406	System booted in Non-Descriptor mode, but the flash appears to contain a valid signature	Check SPI device	
407	User provided Chip Erase Timeout has been reached. If the timeout value was set incorrectly the chip erase may still occur.	Check fparts.txt or its equivalent file	
440	Invalid Fixed Offset variable name	Check Variable name	
441	Invalid Fixed Offset variable Id	Check Variable ID	
442	Param file <file> is already opened	Close parameter file	
444	Invalid name or Id of FOV	Check variable name or ID	

Error Code	Error Message	Explanation	Suggestion
445	Invalid length of FOV value. Check FOV configuration file for correct length	Check length of FOV parameter in parmeter file	
446	Password does not match the criteria	Password does not meet strong password requirements	
447	Error occurred while reading FOV configuration file		
448	Invalid hash certificate file	Check hash certificate file	
449	Valid PID/PPS/Password records are not found in	Check PID/PPS/Password records and ensure that all 3 values exist	
450	Invalid Global locked value entered	Globallocked value is incorrect. Value should be 0x01 when modifying FOV parameters is no longer desired	
451	Unable to get master base address from the descriptor	Check file integrity	
452	Verification of End Of Manufacturing settings failed	Attempt command again. If problem persists, file a sighting	
453	End Of Manufacturing Operation failure - Verification failure on Global Locked settings	Verify global locked bit has not been previously set	
454	End Of Manufacturing Operation failure - Verification failure on ME Manuf counter	Verify MEManuf counter is valid	
455	End Of Manufacturing Operation failure - Verification failure on Descriptor Lock settings	Verify Descriptor region is present and not corrupt	
456	Invalid hexadecimal value entered for the FOV	Check value for FOV supplied	
480	The setup file header has an illegal UUID	UUID must be valid before ME is turned on	
481	The setup file version is unsupported	Check setup file integrity	
482	A record encountered that does not contain an entry with the Current MEBx password	Current MEBX password must be supplied	

Error Code	Error Message	Explanation	Suggestion
483	The given buffer length is invalid	Check buffer length value	
484	The record chunk count cannot contain all of the setup file record data	Setup file number exceeded	
485	The setup file header indicates that there are no valid records	Setup file has no valid records. Check setup file integrity	
486	The given buffer is invalid	Check buffer value	
487	A record entry with an invalid Module ID was encountered	Check record values. Check Setup file Integrity	
488	A record was encountered with an invalid record number	Check record values. Check Setup file Integrity	
489	The setup file header contains an invalid module ID list	Check record values. Check Setup file Integrity	
490	The setup file header contains an invalid byte count	Check record values. Check Setup file Integrity	
491	The setup file record id is not RECORD_IDENTIFIER_DATA_RECORD	Check record values. Check Setup file Integrity	
492	The list of data record entries is invalid	Check record values. Check Setup file Integrity	
493	The CurrentMEBx password is invalid	MEBX password does not meet strong password requirements	
494	The NewMEBx password is invalid	MEBX password does not meet strong password requirements	
495	The PID is invalid	Check to see if value is valid. Check file integrity	
496	The PPS is invalid	Check to see if value is valid. Check file integrity	
497	The PID checksum failed	Check to see if value is valid. Check file integrity	
498	The PPS checksum failed	Check to see if value is valid. Check file integrity	

Error Code	Error Message	Explanation	Suggestion
499	The data record is missing a CurrentMEBx password entry	Missing value is required	
500	The data record is missing a NewMEBx password entry	Missing value is required	
501	The data record is missing a PID entry	Missing value is required	
502	The data record is missing a PPS entry	Missing value is required	

B.6 UPDPARAM errors:

Error Codes	Description
0	Success
3001	Invalid arguments specified
3002	Invalid Parameter value
3003	Error occurred while opening image file
3004	Parsing of image file failed
3005	MEI communication failed
3006	File does not exist
3007	Operating system is not supported
3008	AMT Internal error occurred
3009	User defined certificate hash table is full
3010	Unable to start MEI
3011	Invalid input file name
3012	Chipset not supported by the tool
3013	PID value is NULL
3014	PPS value is NULL
3015	Configuration Server FQDN value is NULL
3016	PKI DNS Suffix value is NULL
3017	Host Name value is NULL
3018	Domain Name value is NULL
3019	The setup file header has an invalid UUID
3020	The setup file version is unsupported
3021	A record has been encountered that does not contain an entry with the Current MEBx Password

Tool Error message

Error Codes	Description
3022	The given buffer length is invalid
3023	The header chunk count cannot contain all of the setup file header data
3024	The record chunk count cannot contain all of the setup file record data
3025	The requested index is invalid
3026	The setup file header indicates that there are no valid records
3027	The given buffer is invalid
3028	A record entry with an invalid Module ID was encountered
3029	A record was encountered with an invalid record number
3030	The setup file header contains an invalid module ID list
3031	he setup file header contains an invalid byte count
3032	The setup file record id is invalid
3033	The list of data record entries is invalid
3034	Failed to write to the given file
3035	Failed to read from the given file
3036	Failed to create random numbers
3037	The CurrentMEBx password is invalid
3038	The NewMEBx password is invalid
3039	The PID is invalid
3040	The PPS is invalid
3041	The data record is missing a CurrentMEBx password entry
3042	The data record is missing a NewMEBx password entry
3043	The data record is missing a PID entry
3044	The data record is missing a PPS entry
3045	The data record is missing a PKI DNS Suffix entry.
3046	The data record is missing a Config Server FQDN entry
3047	The data record is missing a ZTC entry
3048	The data record is missing a Pre-Installed Certificate enabled entry
3049	The data record is missing a User defined certificate config entry
3050	The data record is missing a User defined certificate Add entry
3051	The data record is missing a SOL/IDER enable entry
3052	Manageability Mode data missing in USB File

Appendix C ME Variable changes

C.1   FOV changes:

C.1.1   Removed variables

Variable name	Reason removed	comments
QST Lock (QST state control)	According to the SKU manager implementation	
Manageability mode (FOV 0x2004)		can be controlled through OEM SKU rule (Permanently enable/disable of manageability application)
Manageability Mode Lock (FOV 0x2003)		
Intel ® iQST EN (FOV 0x002)		
Intel ® iQST Lock (FOV 0x1001)		
IRWT state control (FOV 0x4001) (Wox)		
IRWT lock (FOV 0x4002)		
RCS Capability setting removed		can be controlled through OEM SKU rule (Permanently enable/disable of manageability application)
IRWT state control removed		

These setting are now controlled by SKU manager logic through the "permanently enable/disable" and "Shipping enable/disable" settings

1) Intel® AT-d removal:
   - All AT-d related settings

C.1.2   Changed values/structure

Variable name	Change	Reason	comments
Default Power Package	Changed from including 7 possible PP to including 2 PP	Power package reduction	
OEM SKU rule	structure changes	According to the SKU manager implementation	No backward compatibility for FPT. This relates to the permanently disabled settings in FITc. OEM SKU rule will be changed to OEM capabilities and will control permanently disabled features.

C.1.3   Name changes
- "Full tests counter name" changed to "Manufacturing test counter"
- "ZTC enabled" changed to "Remote configuration Enabled"
- "Enabler ID" name changed to "Remote PC Assist Technology ID"
- "OEM description" name changed to "Remote PC Assist Technology Enabler Name"

C.1.4   New settings:

1) General settings:
   - Shipment time state setting – controls shipment enabled state
     - Intel$^®$ QST Enable/Disable
     - PAVP 1.5 Enable/Disable
     - Manageability Application Enable/Disable
     - Intel$^®$ Identitiy Protection Technology Enable/Disable
     - Intel Remote Wake Technology (Corwin Springs) Ship State
   - Permanently Disabled settings: If set to yes, the technology cannot be turned on on the platform:
     - Intel QST Permanently Disabled
     - PAVP 1.5 Permanently Disabled
     - Intel$^®$ Identitiy Protection Technology Permanently Disabled
     - Manageability Application Permanently Disabled
     - KVM Permanently Disabled
     - Intel Remote Wake Technology Permanently Disabled

2) KVM settings :
   - " KVM Host I/F Enabled"
   - "KVM Opt-In Enabled Policy": Opt-in Configurable from Remote IT: Opt in (user consent) policy for change from PTNI
   - "KVM Opt-In PTNI Editable Policy": Opt in (user consent) en/dis
   - USBr Settings: (Controls USB redirection for mouse and keyboard input for the KVM feature. OEM should choose which EHCI controller is physically connected on the board)
     - EHCI 1 enabled
     - EHCI 2 enabled

3) RPAT:
   - RPAT HW Button

4) AMT:
   - Provisioning Period
   -
5) ICLS setting:
   - Unlocking SMTP: used to unlock the Service Manufacturing Test Permit after Global Lock is Set

C.2 FITc changes:

C.2.1 Removed settings

Variable name	Reason removed	comments
Manageability mode	According to the SKU manager implementation	can be controlled through OEM SKU rule (Ship State enable/disable of manageability application)
Manageability mode lock		can be controlled through OEM SKU rule (Permanently Disabled yes/No of Manageability Application)
Intel® AMT Supported		
Intel ® Remote Wake Technology Supported		can be controlled through OEM SKU rule Intel® Remote Wake Technology Permanently Disabled
Intel ® AMT Configuration Mode	Replace SMB Mode with Manual Configuration Mode	
Flash Descriptor Override Pin-Strap Ignore		
Intel ® iQST Supported	According to the SKU manager implementation	can be controlled through OEM SKU rule (QST Permanently Disabled and shipping enable/disable)
Intel ® iQST Lock		
ME Visual LED Indicator Enabled	setting is no longer supported by FW and due to chipset changes.	Can controlled through OEM SKU rule (Permanently enable/disable of QST)

Variable name	Reason removed	comments
ME Flash Protection Override Enabled		Not used by OEMs.
ASF2 Supported	there is no longer support for ASF	
Intel ® Standard Manageability Supported	According to the SKU manager implementation	Can be controlled through OEM SKU rule (Disable Intel® AMT; Enable Intel® Standard Manageability, Manageability Applicatoin shipping and Permanently enable/disable settings)
Intel ® Remote PC Assist Technology Supported		
Remote Connectivity Service Capability	According to the SKU manager implementation	Manageability Application permanently disabled on Consumer platforms only
Intel ® Intel® Identity Protection Technology Technology Supported		can be controlled by Intel ® Intel® Identity Protection Technology permanently disabled
Intel ® TPM Supported	Removal of iTPM support	
Intel ® iTPM FIPS 140-2		
Intel ® iTPM Physical Presence Lifetime Lock		
Intel ® iTPM Physical Presence Hardware Enabled		
Intel ® iTPM Physical Presence Command Enabled		
Authentication Fail Threshold		
Initial Lockout Time		
Lockout Increase Factor		
Local Firmware Update Enabled	Reduce Manufacturing Parameters	Unused by OEMs / hard coded
PET Language Code		
PET OEM Custom Fields 00 - 15		
PET OEM Custom Fields 16 - 31		
PET OEM Custom Fields 32 - 47		
PET OEM Custom Fields 48 - 63		
PET OEM Custom Fields Length		

ME Variable changes

Variable name	Reason removed	comments
PET Community String		
Configuration Server Port		
Configuration Server Name		
Configuration Server IP		
Intel ® AMT Host Name		
Intel ® AMT Domain Name		
DHCP Enabled		
Intel ® AMT Static IP Address		
Intel ® AMT Static IP Subnet Mask		
Intel ® AMT Static IP Default Gateway Address		
Intel ® AMT Static IP Primary DNS Address		
Intel ® AMT Static IP Secondary DNS Address		
IDER Boot Capable		
SOL Boot Capable		
Intel ® AMT VLAN Local Configuration Blocked		
Config Server FQDN		
Intel ® AMT Legacy Provisioning Mode Supported	FW no longer support legacy provisioning mode	
Intel ® Anti-Theft (AT-d) Technology	Removal of AT-d support	
Enable Intel ® Anti-Theft Technology		
Platform Repurpose Disable Allowed		
Single Sign On		
Third Party Configuration Policy		
Platform and Device Metadata Cipher Policy		
Escrow of Migration Package on USB		
Runtime Device Limit		
Device Host Data Region Cipher Policy		

Variable name	Reason removed	comments
Local Operator Primary Authentication Mode		
Local Operator Recovery Authentication Mode		
Local Admin Primary Authentication Mode		
Local Admin Recovery Authentication Mode		
Credential Rotation Frequency		
Remote Admin Policy		
Remote Unlock Policy		
Platform Token Storage		

C.2.2   Name chages:

LAN power well changed to "LAN power well config"

"Remote Connectivity Service Enabler Name" changed to "Remote Connectivity Service Enabler Name (OEM Description)"

"Intel ® Remote Wake Technology Enabled" changed to "Intel® Remote Wake Technology Enabled/Disabled" (Shipment state)

"Intel ® Anti-Theft (AT-p) Technology" changed to "Intel® Anti-Theft (AT-p) permanently "disabled

ICC Supported changed to ICC permanently disabled

Protected Audio Video Path Supported changed t0 "Protected Audio Video Path Permanently Disabled"

C.2.3   Changed values/structure

- o   LAN power well changed to "LAN power well config" and there are more values to choose

- • WLAN power well values changed and now there are more values to choose from
- • Power Packages – change in values
- • Default Power Package - change in values

C.2.4 New settings

- o   Shipment time state setting – controls shipment enabled state
  - o   Intel QST Ship State
  - o   PAVP 1.5 Ship State
  - o   Manageability Application Enable/Disable
  - o   Enable Intel® Standard Manageability; Disable Intel® AMT
  - o   KVM Enable/Disable
  - o   Intel Remote Wake Technology (Corwin Springs) Enable/Disable

- o   Permanently Disabled settings: If set to yes, the technology cannot be turned on on the platform:
  - o   Intel QST Permanent Disabled
  - o   PAVP 1.5 Permanent Disabled
  - o   Manageability Application Permanent Disabled
  - o   KVM Permanent Disabled
  - o   Intel Remote Wake Technology Permanent Disabled

- o   KVM settings:
  - o   KVM Enable/Disable: KVM Host I/F enabled
  - o   Opt-in Configurable from Remote IT: Opt in (user consent) policy for change from PTNI (OptinPTNIEnabledPolicy)
  - o   User Opt-in Enable/Disable: Opt in (user consent) enabled (OptinPTNIEnabledPolicy)
  - o   USBr Settings: (Controls USB redirection for mouse and keyboard input for the KVM feature. OEM should choose which EHCI controller is physically connected on the board)
    - o   EHCI 1 enabled

    - o   EHCI 2 enabled

- •   KVM settings:
  - o   Selectable ICC records count
  - o   1st MGPIO for ICC Record Sel
  - o   2nd MGPIO for ICC Record Sel
  - o   3rd MGPIO for ICC Record Sel

- •   Other settings:
  - o   HMRFPO Enabled – for FW downgrade
  - o   RPAT HW Button
  - o   LAN controller is for ME's usage of Intel LAN.
  - o   M3 Power Rails availability tells the kernel that M3 is there. CRBs need this set to yes. Customer boards will need to have to populate this based on if M3 rail is there (even if using it for 4 MB SKU).

Appendix D SKU features

	H55	H57	HM57	PM55	PM57	QS57	QM57	P55	P57	Q57
System Defense	N	N	N	N	N	Y	Y	N	N	Y
3rd Party Data Store (3PDS)	N	N	N	N	N	Y	Y	N	N	Y
Remote Control Power Operations	N	N	N	N	N	Y	Y	N	N	Y
Intel® Management Engine Interface (MEI)	N	N	N	N	N	Y	Y	N	N	Y
Local Manageability Service	N	N	N	N	N	Y	Y	N	N	Y
Intel® AMT	N	N	N	N	N	Y	Y	N	N	Y
Standard Manageability	N	N	N	N	N			N	N	
Setup and Configuration	N	N	N	N	N	Y	Y	N	N	Y
Event Manager Interface	N	N	N	N	N	Y	Y	N	N	Y
Network Administration Interface	N	N	N	N	N	Y	Y	N	N	Y
Intel® Quiet System Technology	Y	Y	N	N	N	Y	Y	N	Y	Y
Alert Standard Format (ASF)	Y	Y	N	N	N			N	N	
FW Update	Y	Y	N	N	N	Y	Y	N	N	Y
System for Asset Management	N	N	N	N	N	Y	Y	N	N	Y
Serial over LAN / IDE Redirect (SOL/IDER)	Y	Y	N	N	N	Y	Y	N	N	Y
Agent Presence	N	N	N	N	N	Y	Y	N	N	Y
Security Administration	N	N	N	N	N	Y	Y	N	N	Y
Intel® Remote Wake Technology	Y	Y	N	N	N			N	N	
Kerberos	N	N	N	N	N	Y	Y	N	N	Y
Linux Support	N	N	N	N	N	N	N	N	N	N
Windows Vista Support	N	N	N	N	N	Y	Y	Y	Y	Y
Intel® Management & Security Status Icon/App	N	N	N	N	N	Y	Y	N	N	Y
User Notification	N	N	N	N	N	Y	Y	N	N	Y
ME Hardening (Blob data protection)	N	N	N	N	N	Y	Y	N	N	Y

	H55	H57	HM57	PM55	PM57	QS57	QM57	P55	P57	Q57
ME Wake on LAN	N	N	N	N	N	Y	Y	N	N	Y
Mfg Graceful Shutdown	N	N	N	N	N	Y	Y	N	N	Y
OEM Ctrl of Flash Programming	Y	Y	N	N	N	Y	Y	Y	N	Y
OEM Manufacturing Audit Utility	N	N	N	N	N	Y	Y	N	N	Y
DNS Environment Detect	N	N	N	N	N	Y	Y	N	N	Y
ISV Local access to Event Log	N	N	N	N	N	Y	Y	N	N	Y
Dual Interface	N	N	N	N	N	Y	Y	N	N	Y
802.1x (wired or wireless)	N	N	N	N	N	Y	Y	N	N	Y
Power Package Support	N	N	N	N	N	Y	Y	N	N	Y
Wireless use of Wired MAC address	N	N	N	N	N	Y	Y	N	N	Y
Asset Inventory	N	N	N	N	N	Y	Y	N	N	Y
Host VPN for Intel AMT	N	N	N	N	N	Y	Y	N	N	Y
Cisco NAC Posture Support	N	N	N	N	N	Y	Y	N	N	Y
EAP TLS Support & Cisco* Certification support for Cisco* NAC Embedded Trust Agent	N	N	N	N	N	Y	Y	N	N	Y
PXE Support over 802.1x/NAC	N	N	N	N	N	Y	Y	N	N	Y
ME-Unconfigure w/o password	N	N	N	N	N	Y	Y	N	N	Y
802.11i Wireless Security	N	N	N	N	N	Y	Y	N	N	N
Wireless Agent Presence*	N	N	N	N	N	Y	Y	N	N	N
Wireless Asset Inventory*	N	N	N	N	N	Y	Y	N	N	N
Wireless System Defense	N	N	N	N	N	Y	Y	N	N	N
Wireless Agent Presence	N	N	N	N	N	Y	Y	N	N	N
Wireless SOL/IDER	N	N	N	N	N	Y	Y	N	N	N
Wireless 3rd Party Data Store	N	N	N	N	N	Y	Y	N	N	N
Wireless Remote Control	N	N	N	N	N	Y	Y	N	N	N
Wireless Network Administration	N	N	N	N	N	Y	Y	N	N	N
Wireless FW Update	N	N	N	N	N	Y	Y	N	N	N
Wireless support in S0/H0 w/out ME profiles	N	N	N	N	N	Y	Y	N	N	N
Wireless ME access in Sx	N	N	N	N	N	Y	Y	N	N	N
Remote Configuration	N	N	N	N	N	Y	Y	N	N	Y
WS-MAN Support	N	N	N	N	N	Y	Y	N	N	Y

	H55	H57	HM57	PM55	PM57	QS57	QM57	P55	P57	Q57
Enhanced System Defense Filters	N	N	N	N	N	Y	Y	N	N	Y
DASH Profiles Compliance	N	N	N	N	N	Y	Y	N	N	Y
Measured Intel AMT	N	N	N	N	N	Y	Y	N	N	Y
MSFT NAP Support	N	N	N	N	N	Y	Y	N	N	Y
Fast Call for Help	N	Y	N	N	N	Y	Y	N	N	Y
Remote Scheduled Maintenance	N	Y	N	N	N	Y	Y	N	N	Y
Remote Alerts	N	Y	N	N	N	Y	Y	N	N	Y
Intel® Access Monitor (previously Audit Log)	N	N	N	N	N	Y	Y	N	N	Y
Intel® Access Monitor Default Setting	N	N	N	N	N	Y	Y	N	N	Y
Intel® Anti-Theft Technology - PC Protection			Y	N	Y	Y	Y	N	N	Y
Intel® WiFi/WiMax			N	N	N	Y	Y	N	N	N
Manageability and Encryption ISV Interoperability requirements	N	N	N	N	N	Y	Y	N	N	Y
VE Manufacturing Tool	N	N	N	N	N	Y	Y	N	N	Y
vPro Bare Metal Configuration Enable/Disable	N	N	N	N	N	Y	Y	N	N	Y
Transfer Soft Creek upgrade for in-field service replacement platform	N	N	N	N	N	Y	Y	N	N	Y
Simplified Power Packages	N	N	N	N	N	Y	Y	N	N	Y
Support for Special Characters in AMT Hostname	N	N	N	N	N	Y	Y	N	N	Y
Intel® Remote PC Assist Technology for Consumer (Castle Peak)	Y	Y	Y	N	Y	N	N	N	N	N
Intel® Remote PC Assist Technology for Business	N	N	N	N	N	Y	Y	N	N	Y
Intel® Management and Security Status Icon/App (IMSS) Updates	N	N	N	N	N	Y	Y	N	N	Y
PCH ME ROM Hardening	N	N	N	N	N	Y	Y	N	N	Y
IPv6 Support	N	N	N	N	N	Y	Y	N	N	Y
SHA-256 Support	N	N	N	N	N	Y	Y	N	N	Y
Intel® ME Ignition MPC Support	N	N	N	Y	N	N	N	Y	N	N
Intel® ME Ignition ICC Support	N	N	N	Y	N	N	N	Y	N	N
OEM VE SKU Selection via Flash Programming Tool	N	N	N	N	N	Y	Y	N	N	Y
Support for Presence WS-Event Notification	N	N	N	N	N	Y	Y	N	N	Y
KVM Redirection	N	N	N	N	N	Y	Y	N	N	Y
vPro™ "Out of the Box"	N	N	N	N	N	Y	Y	N	N	Y

	H55	H57	HM57	PM55	PM57	QS57	QM57	P55	P57	Q57
Discovery										
Intel® Identity Protection Technology (IPT)	Y	Y	Y	N	Y	N	N	N	N	N
Switchable Graphics support for KVM-r and Sprite	N	N	N	N	N	Y	Y	N	N	N
Softcreek Upgrade Service (Std Manageability to full Manageability on Ibex Peak Q55)	N	N	N	N	N	Y	Y	N	N	Y
CILA Support added to IMSS	N	N	N	N	N	Y	Y	N	N	Y
PAVP	Y	Y	Y	N	N	Y	Y	N	N	Y
Integrated NAND (Braidwood)	N	Y	Y	N	Y	Y	Y	N	Y	Y
Local wake and update (Alarm Clock)	N	N	N	N	N	Y	Y	N	N	Y

§