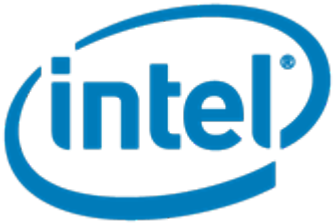


Manufacturing with Intel® Management Engine Ignition Firmware, on Ibex Peak based Platforms

January 2009/Revision 1.0

This and other MAS documents are available at http://learn.intel.com site.

Legal Disclaimer

- INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.
- UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.
- Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.
- The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.
- Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.
- Code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees, and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or service and any such use of Intel’s internal code names is at the sole risk of the user.
- Intel, the Intel logo, and Intel vPro are trademarks of Intel Corporation in the U.S. and other countries.
- *Other names and brands may be claimed as the property of others.
-

Introduction

- *All* Q3'09 desktop and notebook platforms have a new requirement related to firmware, and cannot ship without it.

- High level design tasks:
  - Check flash architecture against requirements.
  - Add 1M bits (128K bytes) of Intel-supplied firmware to flash image.

- High level manufacturing tasks:
  - Use the correct flash image & flash parts from R&D.
  - Use an Intel-supplied test at functional test.

- This document addresses how to manufacture using Intel® Management Engine Ignition Firmware.

Module Table of Contents

- Technology Overview
  - What it does
  - Applicable chipsets/platforms
  - How it works
- Implementing in Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

What It Does

- Intel® ME Ignition Firmware performs some new platform initialization steps, *before* the BIOS boots.
- These steps allow:
  - Adjustment (in firmware) of clock parameters that would otherwise require board or silicon redesign.
    - Previously unavailable clock tune-ability.
  - Faster resolution (workaround) of some types of silicon issues.
  - 3rd party fan solutions to work with more information.
  - Fewer processor SKUs to be stocked for assembly.

Applicable Chipsets/Platforms

	Piketon (Desktop) **Also Kings Creek and Fox Hollow**	
PCH (chipset) Platform Control Hub	Ibex Peak	
Processor	Lynnfield (Q3'09)	Lynnfield/Havendale (Q1'10)
Available Firmware (FW) SKUs	**ME Ignition FW (required)**	**ME Ignition FW** or vPro FW ** or Home IT FW **

	Calpella (Notebook) **Business and Consumer**	
PCH (chipset) Platform Control Hub	Ibex Peak	
Processor	Clarksfield (Q3'09)	Clarksfield/Auburndale (Q1'10)
Available Firmware (FW) SKUs	**ME Ignition FW (required)**	**ME Ignition FW** or vPro FW ** or Home IT FW **

**** vPro and Home IT FW SKUs will be covered in a separate MAS.**

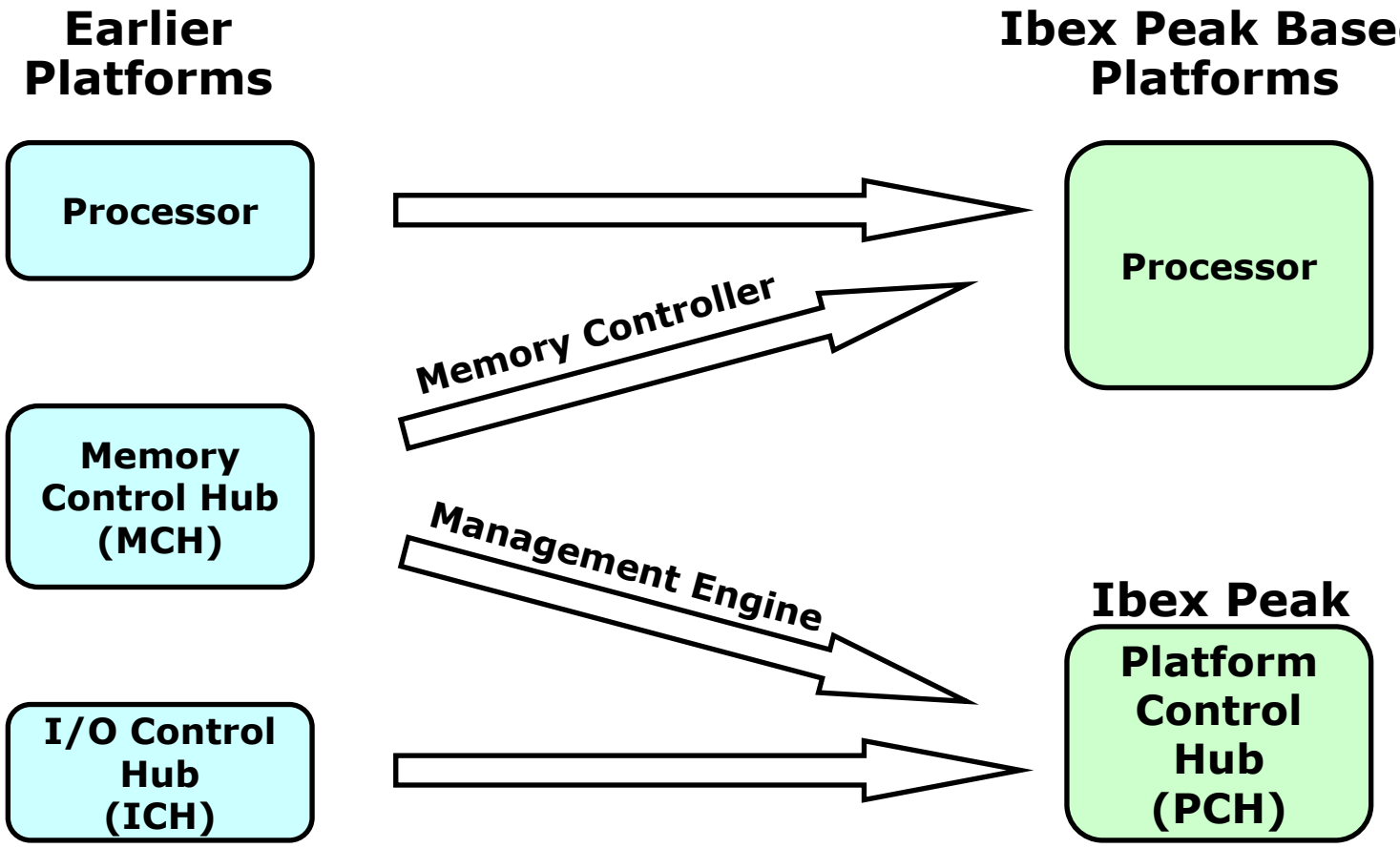
Intel® ME Ignition FW Features

Configure Platform Clocks	Provides drive strength & slew rate controls for EMI management. Adds debug & tuning capabilities, and power management. Allows up to 8 board designs, with different clock configurations, to use the same ME FW without modification.	
Silicon Workaround Capability (SWC)	Provides potential workaround capability for certain limited types of silicon issues, should they arise.	This allows faster resolution of some issues that would otherwise require processor or chipset replacement.
Thermal Reporting (TR)	Makes thermal and power information available to host-accessible registers or to the Embedded Controller (EC).	This allows third party temperature control solutions to monitor temperatures of CPU core, graphics core, chipset and DIMMs.
Configuration Feature (CF)	With Havendale or Auburndale, auto enables/disables processor ECC and integrated graphics, and sets PCI Express Graphics lane width, based on capabilities of the attached Ibex Peak.	This reduces the number of Intel component SKUs that must be stocked for board & system assembly.

Examples of Ignition FW Actions

Configure Platform Clocks	If it's found during board design that a certain clock signal is surpassing regulatory limits for electromagnetic emissions...	...R&D can adjust the clock profile to increase the transition time of this signal, in order to reduce emissions, *without* changing the HW design.
Silicon Workaround Capability (SWC)	If Intel finds that a register is being set incorrectly in the chipset HW design...	...then ME, while running Ignition FW during startup, could set the register to the correct value.
Thermal Reporting (TR)	If temperature of CPU core exceeds a preset value...	...ME Ignition FW sends an alert to the EC, so that it can increase fan speed.
Configuration Feature (CF)	If the Ibex Peak SKU does not support Error Correction Code (ECC)...	...ME Ignition FW will automatically turn off ECC in the processor (Havendale/Auburndale only).

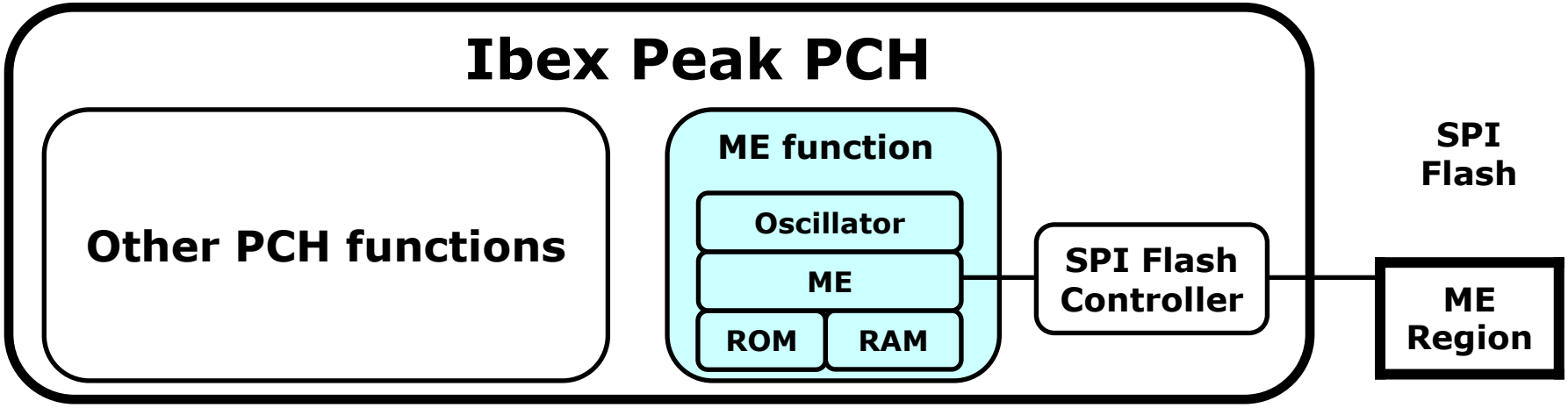
Repartitioning of Platform Functions



Platform functions from 3-chip implementations (Processor/MCH/ICH) have been distributed to 2 packages (Processor/PCH).

Intro to Intel® Management Engine (ME)

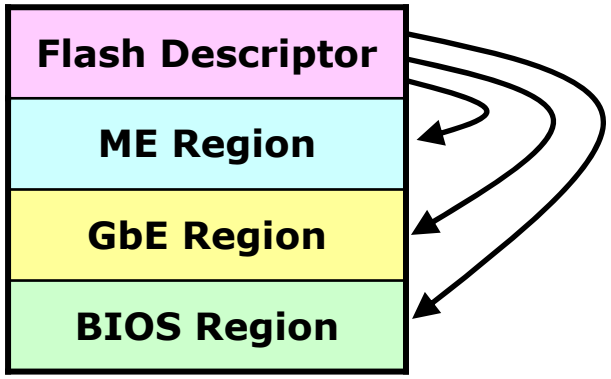
- The ME is an independent processor core embedded *inside* the PCH.
  - Has its own internal clock, ROM, and RAM.
  - Executes from its internal ROM, internal RAM, or from the ME Region of SPI flash *(will be explained shortly)*.
  - Firmware in the ME Region is like the Operating System for the ME.



- On Ibex Peak based platforms with Intel® ME Ignition Firmware (scope of this document), the ME is ONLY used to support ME Ignition FW.
  - On *other* platforms, the ME executes firmware to implement features such as:
    - Intel® Active Management Technology, and Intel® Quiet System Technology.

SPI Flash Management by PCH

- SPI flash is *required* for all Ibex Peak platforms.
  - SPI (Serial Peripheral Interface) is a four-wire serial bus.
- SPI flash must be connected *directly* to the PCH.
  - This allows the PCH to manage access to flash.
- Flash is partitioned into **regions**, allowing different masters (Host/BIOS, ME, GbE) to have different read/write permissions for *each* region.
- The first 4KB is designated as the **Flash Descriptor**.



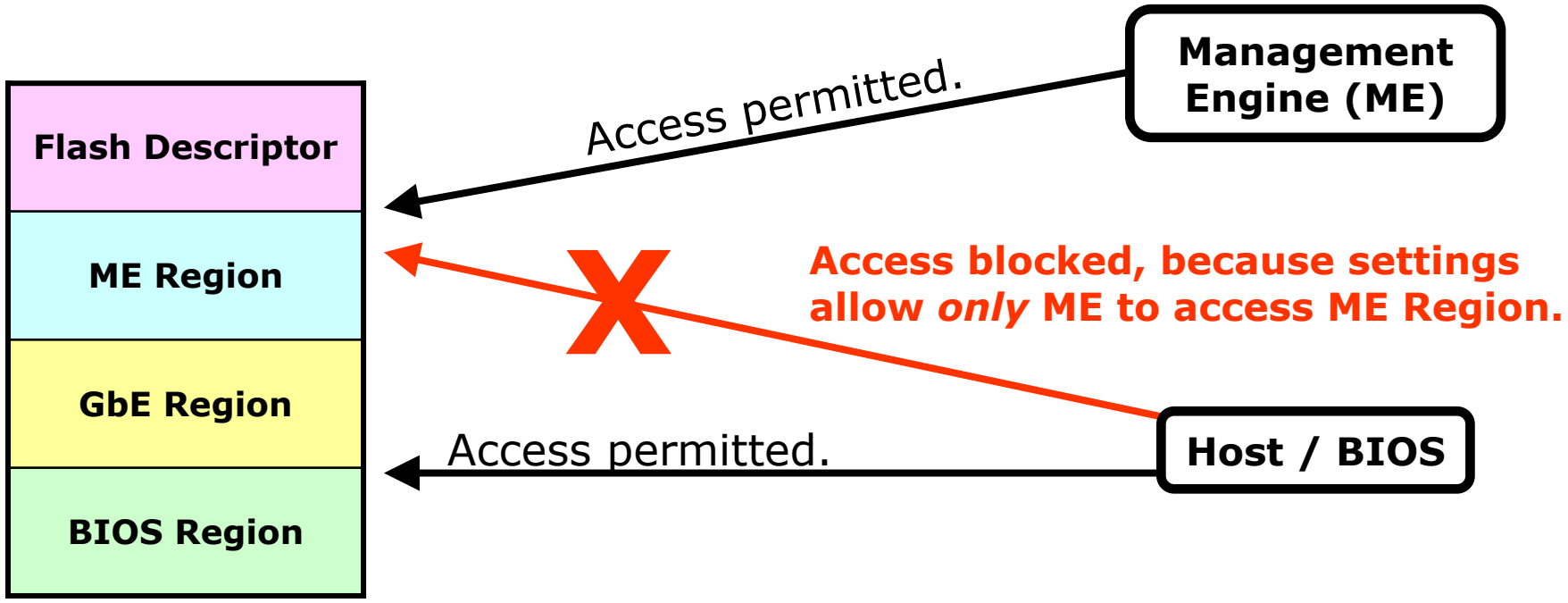
The Flash Descriptor contains:

- Pointers to other regions.
- Access permissions to all regions (including Descriptor Region) for all masters.
- 'Soft Straps' for clock profile selection for Ignition FW.
- Other configuration information for platform HW and FW configuration.

Note: The GbE Region is only needed on designs with Intel Gigabit Ethernet.

SPI Flash Master Access Example

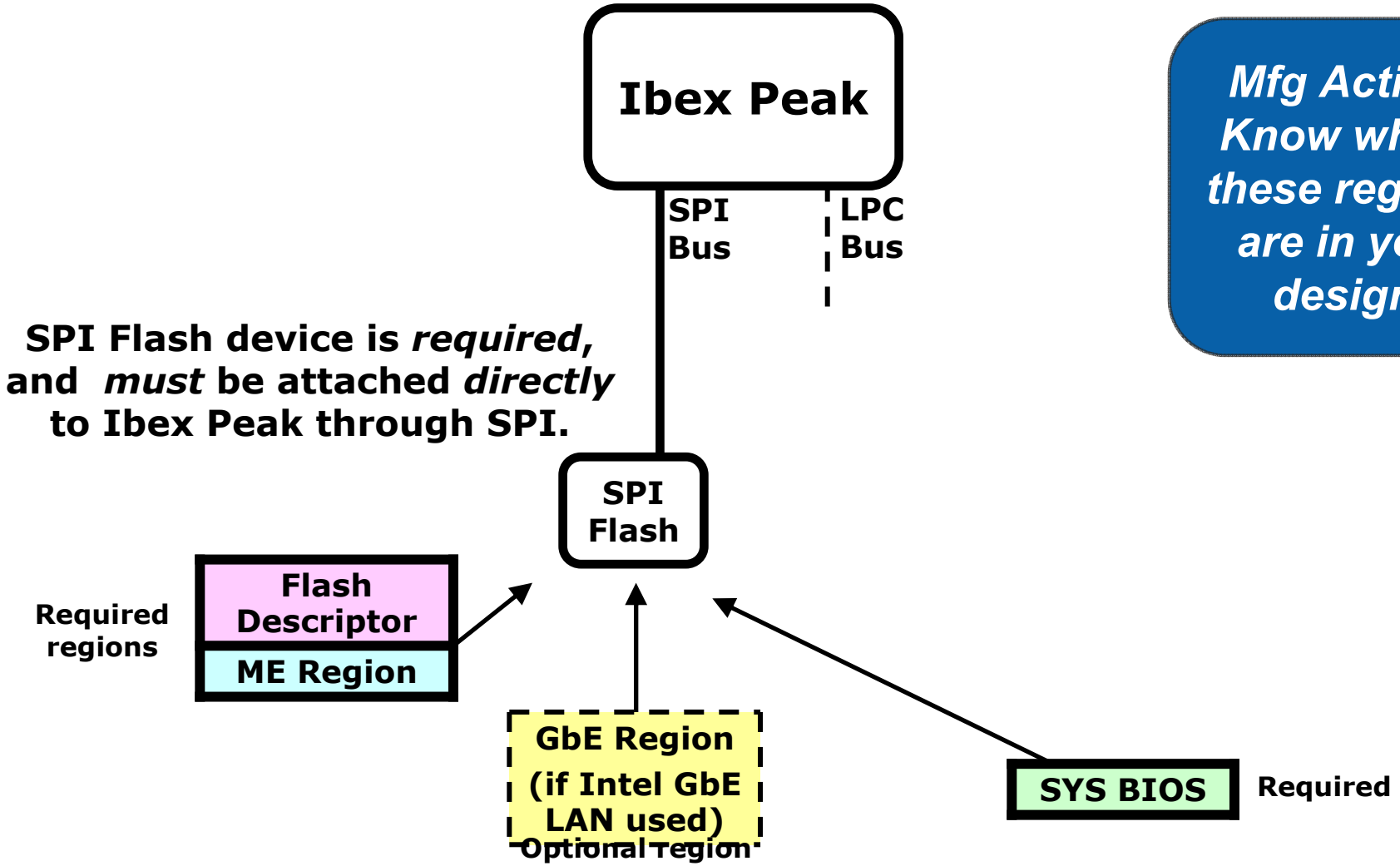
- For example, in the Flash Descriptor:
  - The ME Region can be set to be accessible *only* by the ME, and *not* by the Host/BIOS.
  - The BIOS Region can be set to be accessible *only* by the host/BIOS and *not* by the ME.



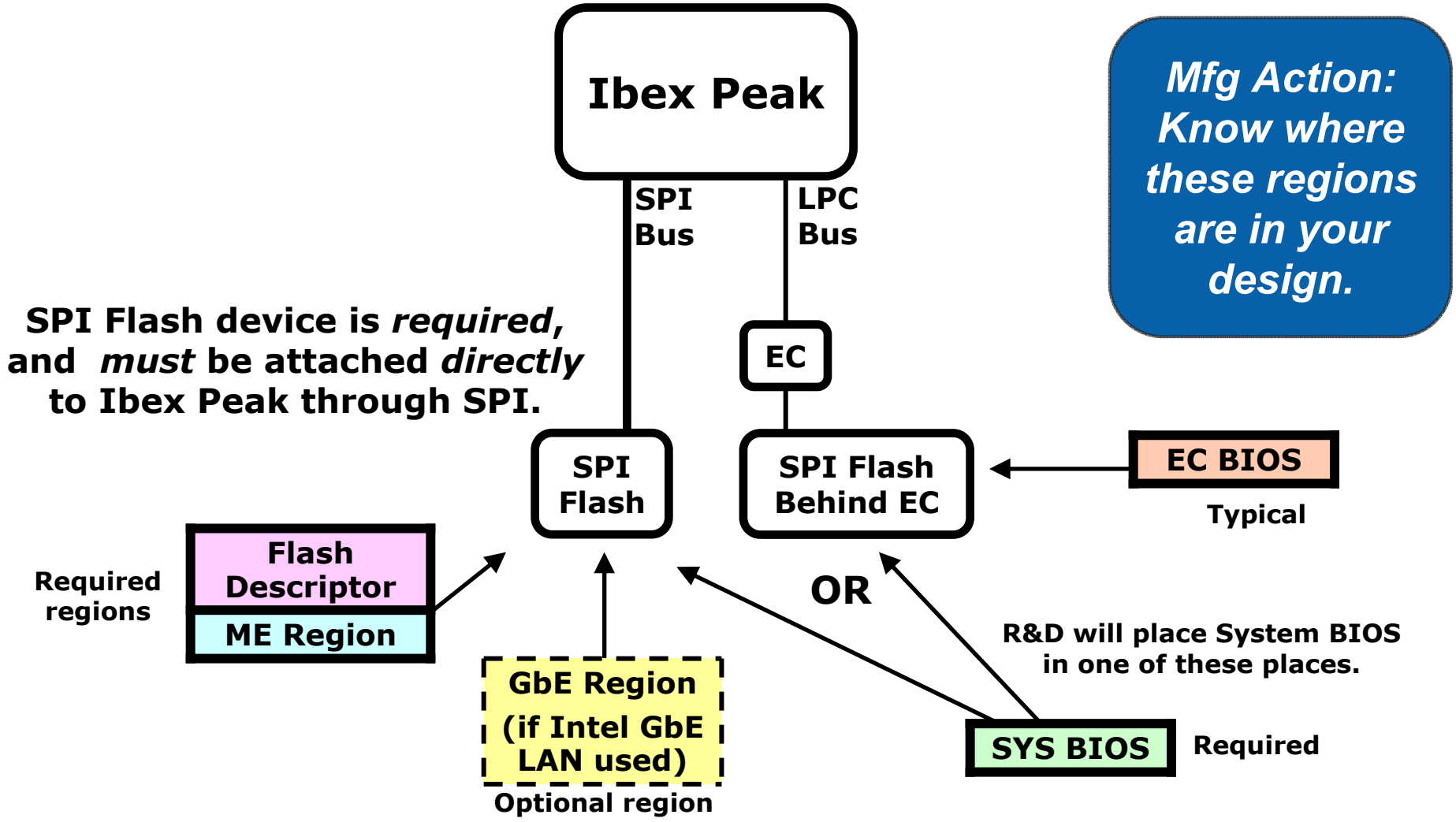
Flash Memory Regions

<table>
<tr><th></th><th colspan="2">Previous platforms</th><th colspan="2">Ibex Peak based platforms</th></tr>
<tr><th></th><th>Required/
Optional</th><th>Location</th><th>Required/
Optional</th><th>Location</th></tr>
<tr><td>Flash Descriptor Region</td><td rowspan="2">These two regions were not used on most boards.
But both were required for FW SKUs with Intel® AMT.</td><td rowspan="3">In ICH-attached SPI flash.</td><td rowspan="2">Both regions are required for all Ibex Peak boards, regardless of ME FW SKU:
Ignition
vPro
Home IT</td><td rowspan="3">In PCH-attached SPI flash.</td></tr>
<tr><td>ME Region</td></tr>
<tr><td>GbE Region</td><td>Required only for Intel GbE LAN.</td><td>Required only for Intel GbE LAN.</td></tr>
<tr><td>BIOS</td><td colspan="2">In ICH-attached SPI flash
(as BIOS Region),
OR behind embedded controller.</td><td colspan="2">In PCH-attached SPI Flash
(as BIOS Region),
OR behind embedded controller.</td></tr>
</table>

Flash Architecture for Ignition FW - Desktop



Flash Architecture for Ignition FW - Mobile



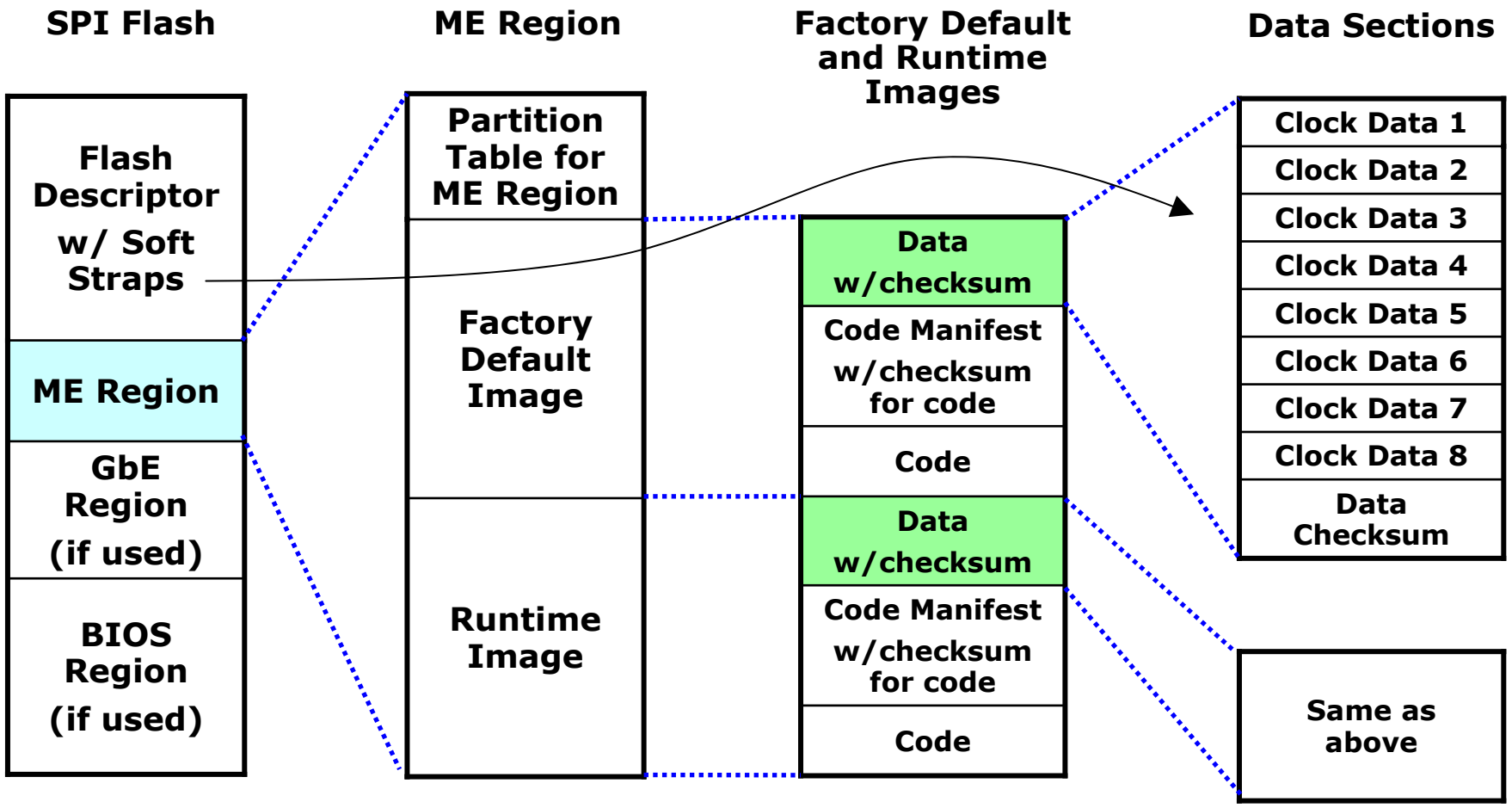
SPI Flash Content – ME Region

- Small addition to firmware, only 1M bit (128K bytes)
  - Compared to 32M bit (4M bytes) ME Region on some previous platforms.
- Includes "Runtime Image" and "Factory Default" Image.
  - **Runtime Image** (this is the normal image):
    - Written during flash programming.
    - **Can be updated anytime using BIOS update.**
      - BIOS must include this new capability.
  - **Factory Default Image** (this is the backup image):
    - Written during flash programming.
    - **Not intended to be updated.** More information is provided on a later slide.

Mfg Action: Make sure that your BIOS include ME update capability.

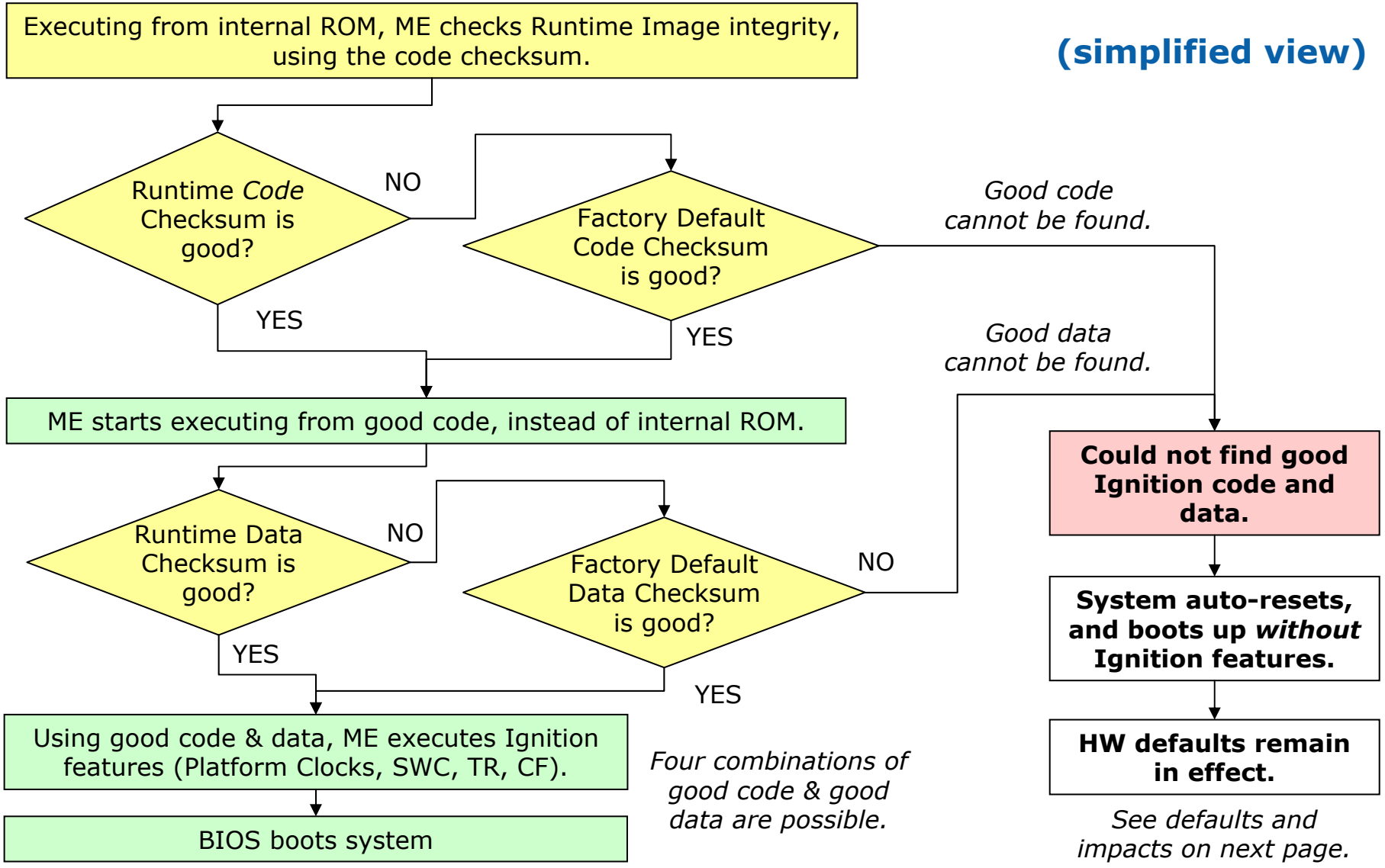
- Fault tolerance: Both images contain checksums, so that ME can detect corruption.
  - ME executes the Runtime Image, unless corruption is detected, then it switches to the Factory Default Image.
  - ME will not run corrupted code or data.
  - If good code and data *cannot* be found, the system still boots using hardwired default values to allow for debug.
    - Exception: Dual processor platforms without good code and data may not boot.

SPI Flash Contents



Soft Straps in Flash Descriptor indicate which Clock Data Set applies to a given board, allowing the *same* ME Region FW image to support 8 *different* board designs.

What Happens at Startup (and every exit from a sleep state)



If Ignition Features Can't Be Initialized

- If the ME is *unable* to initialize Ignition features...
  - ...such as if good Ignition FW code and data cannot be found:
    - The system should ***still boot***...
      - ...but with some features non-operational...
      - ...and others set to default values (contained in the HW).

If this feature can't be initialized...	
Configure Platform Clocks	HW clock defaults in PCH are used. No power management, no EMI-related settings. No board-specific configuration.
Silicon Workaround Capability	No SWC is possible.
Thermal Reporting	No Thermal Reporting is possible.
Configuration Feature (Havendale/Auburndale only)	These default values are used: Integrated Graphics = ON. DDR3 ECC = ON. PCI Express Graphics lane width = x16.

Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
  - Summary
  - Pre-Manufacturing
  - During Manufacturing
  - Post Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

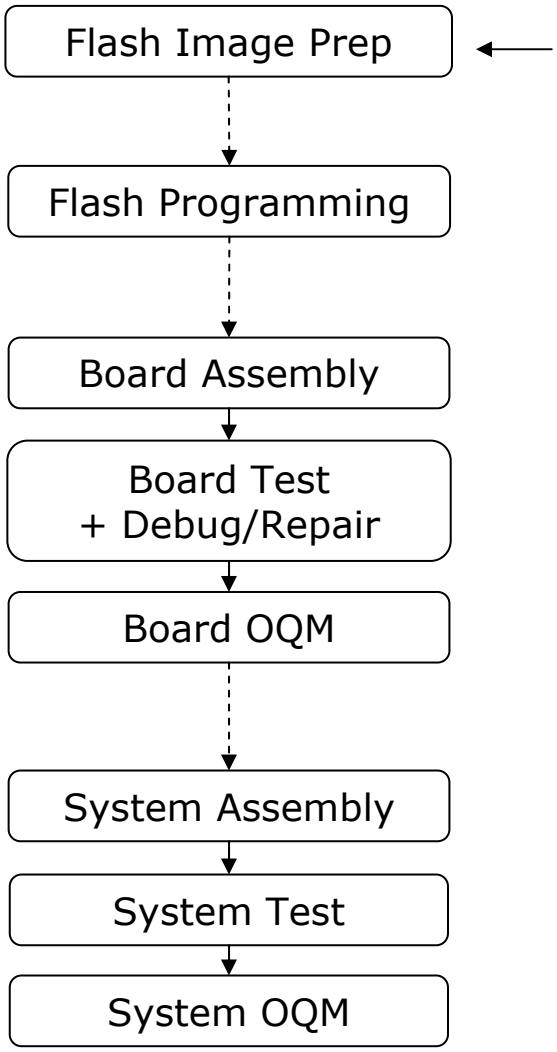
Manufacturing Summary

- Top manufacturing considerations:
  - Include all regions and locking in initial flash image.
  - Don’t create or edit Flash Descriptor or ME Region during manufacturing.
    - Exception: Runtime Image may be updated if needed.
  - Use MEManuf at all test and OQM stations.
  - See that design includes a way to bypass Descriptor locking.
    - Or plan to replace flash parts when they need reflashing.

Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
  - Summary
  - Pre-Manufacturing
  - During Manufacturing
  - Post Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

Pre-Manufacturing Recommended Process Flow



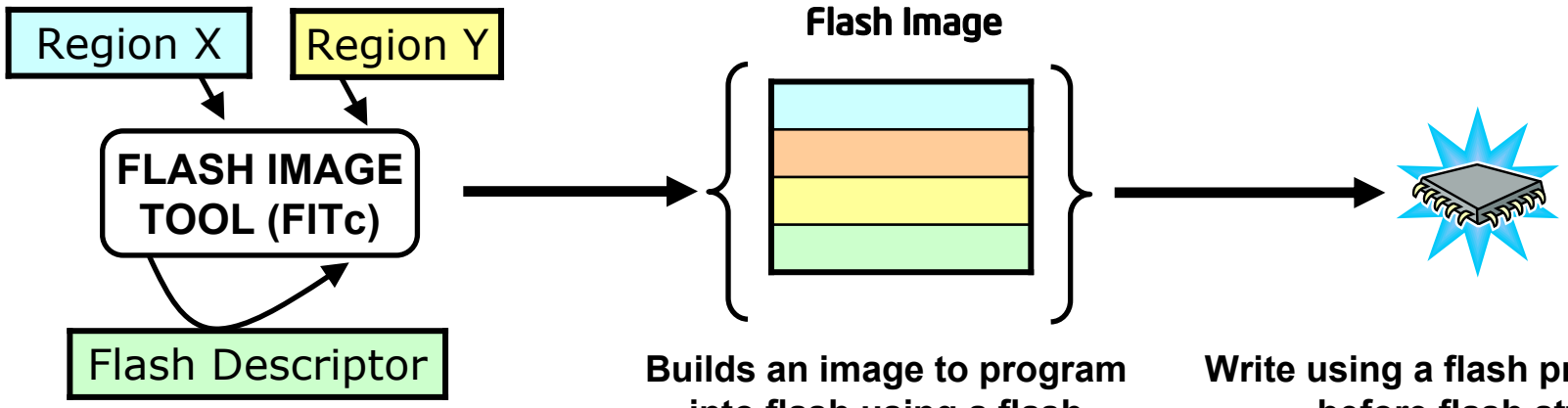
Flash image prep to include:

- **Flash Descriptor**
  - **Set access permissions to all regions.**
  - **Set Soft Straps (incl Clock Data Set selector).**
  - **Lock the Descriptor & ME Region.**
- **ME Region (Runtime & Factory Default images).**
- **BIOS Region (if in same flash as ME Region).**
- **Gigabit Ethernet Region (if needed).**

- **Build the entire flash image, including locking appropriate regions, prior to initial flash programming.**
- **Don't modify the Flash Descriptor during manufacturing.**
- **Except for possibly updating the Runtime Image, don't modify the ME Region during manufacturing.**

FITc – Flash Image Tool (previously Ftoolc)

- FITc combines binaries for different regions (BIOS, ME, etc), and creates a Flash Descriptor for an image to be written to SPI flash.
- Sets configuration options in flash descriptor for ME Firmware.
  - Such as Clock Data Set selector (Soft Strap) for Intel® ME Ignition FW.
- Calculates and inserts checksums for Ignition code and data areas.
- FITc *must* be used for:
  - Flash Descriptor and ME Regions (on all designs).
  - BIOS region, only if BIOS is in the same flash device.
  - GbE Region, only if this region is needed.



Create/edit Flash Descriptor, and combine with other region binaries.

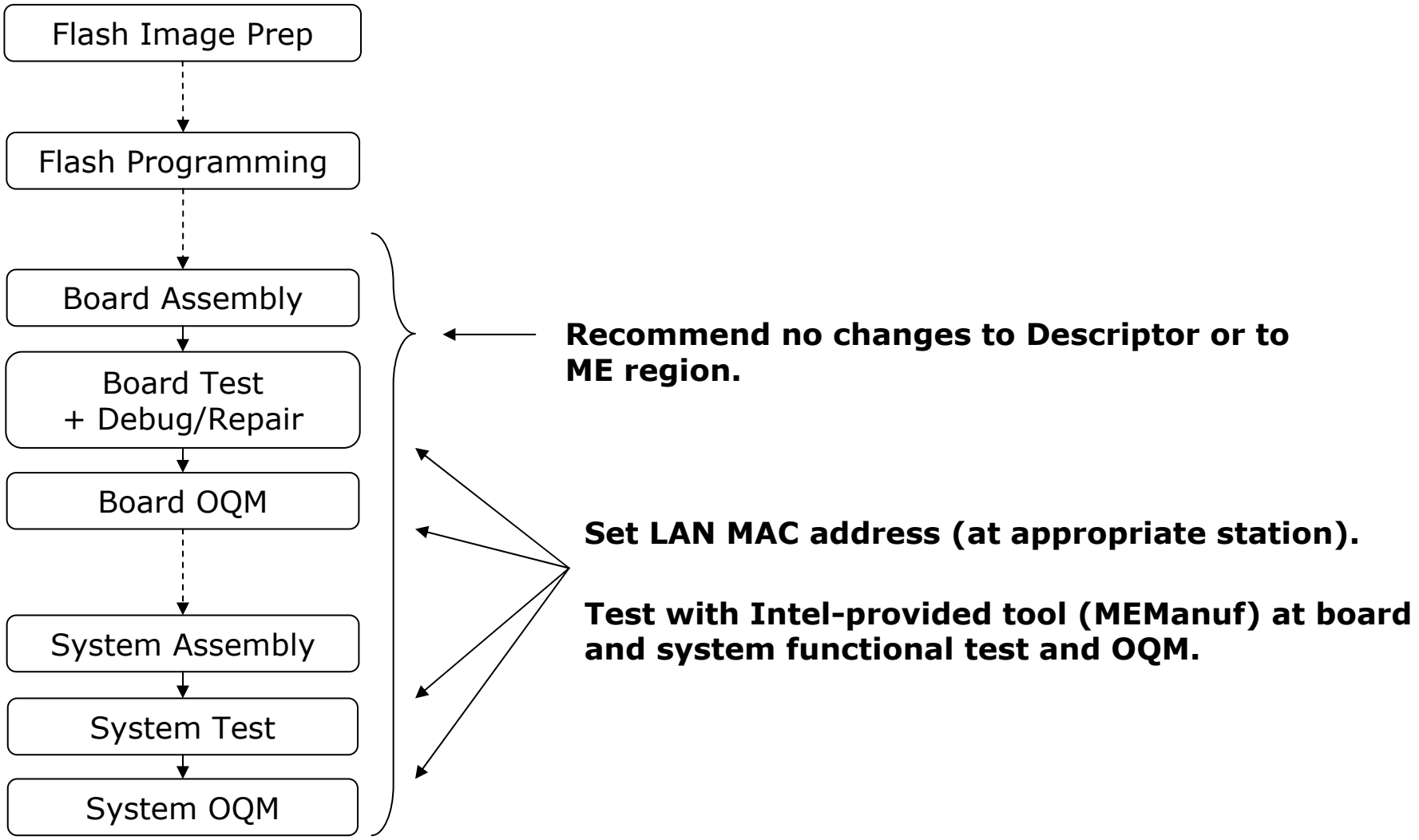
Builds an image to program into flash using a flash programmer. FITc does not write to flash.

Write using a flash programmer before flash attach (or at ICT after flash attach).

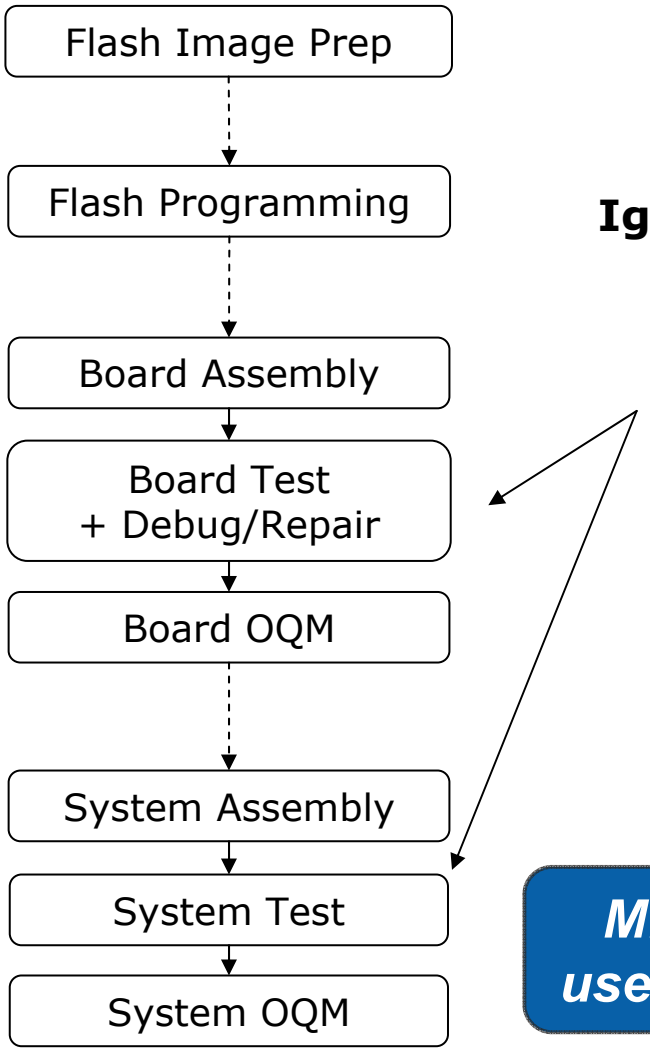
Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
  - Summary
  - Pre-Manufacturing
  - During Manufacturing
  - Post Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

Manufacturing Recommended Process Flow



If Ignition Runtime Image *must* be updated *after* flash programming…



Ignition Runtime Image CAN be updated, but:

It should *only* be done using the BIOS capability.

BIOS has the ability to unlock the ME Region for the update, and re-lock it when finished.

Mfg Action: To update Runtime Image, only use BIOS update capability, not other methods.

If Ignition Factory Default Image *Must* be updated *after* flash programming…

- Example: In the factory, both images could be found to be corrupted.
  - Updating a non-corrupted Factory Default Image is *not recommended.*
- BIOS *cannot* update the Factory Default Image.
- Because the ME Region is locked, the Factory Default Image cannot be updated with software tools alone, such as Intel Flash Programming Tool (FPT).
- Two options are:
  - Override Flash Descriptor permissions using GPIO33 (see next page).
  - Remove flash device from board, and replace with a new one.

Mfg Action: Know how to reflash entire flash device on each design.

GPIO33 for Flash Permission Override

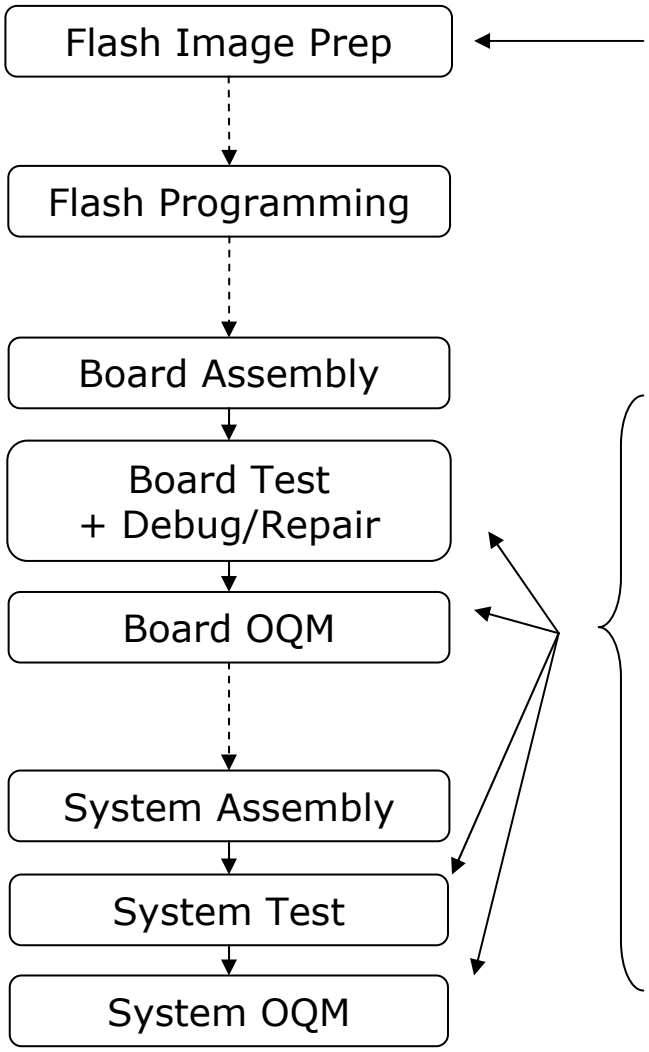
- The PCH GPIO33 pin can be used in the factory, to temporarily override flash access permissions in the Flash Descriptor...
- ...However, the board design must support this use.
- This can be done with a jumper, or with some other method designed into the board to drive PCH GPIO33 to GND during PWROK assertion.
- After permissions are overridden:
  - Use FPT, or other flash programming software, to reprogram the flash.
  - Be aware that after the update (and until there is an ME Reset), ME will still be running the old FW (resident in PCH RAM), and not the FW newly written to flash.
  - ME will boot from new code after a cold reset (host partition reset with power cycle).

If GPIO33 is required for reflash, know how to activate it.

NON-Recommended Practices

NON-Recommended Practice	Better Practice	Reason
Leaving Descriptor *unlocked* at image prep.	Always lock Descriptor at image prep.	Could ship with Descriptor or ME Region unlocked (security issue). Also, no need to leave unlocked.
Leaving ME Region *unlocked* at image prep.	Always lock ME Region at image prep.	
Leaving ME Region *empty* at image prep.	Always program all of the ME region at image prep.	Could ship without Ignition features, since most boards will still boot.
Using *FPT* (or other non-BIOS method) to update ME Region Runtime Image.	Only use the BIOS capability to update Runtime Image.	BIOS is the only supported method for Runtime Image update. Also, BIOS method unlocks/relocks the ME Region.
Using FPT to change soft straps (e.g. to select Clock Data Set)	Only set soft straps in initial flash image.	Clocks may be misconfigured. Potential flash corruption.
Editing clock configuration parameters in manufacturing.	Only set clock parameters in initial flash image.	Clocks may be misconfigured. Potential flash corruption.

Manufacturing Tools



Flash Image Tool (FITc)

Can be used to prepare flash image, or to decompose & analyze an existing image.

"Press key to continue" warning at BIOS screen.

Warns if ME is using Factory Default Code and/or Data.

EEUpd LAN, EEUpd WLAN to set MAC address.
If using Intel LAN and/or WLAN.

MEManuf

Confirms that Runtime Image is being used.

Confirms that Ignition features initialized properly.

BIOS Warning Message

- During boot, if the BIOS finds that ME is running Factory Default Code and/or Data, it will stop and display a message such as:

```
Flash corruption detected. Reflash SPI. Press any key to continue.
```

This wording is preliminary, and may change.

- If ME is running Runtime code and data, boot will *not stop*, and no message will be displayed.

MEManuf

- Use Intel-provided MEManuf tool for manufacturing test of platforms using ME Ignition Firmware.
  - In the Ibex Peak Ignition FW kit, available on VIP database.
- Command line for Windows: MEManufWin.exe
- Command line for DOS: MEManuf.exe
- MEManuf determines:
  - If Runtime or Factory Default *Data* is being used.
  - If Runtime or Factory Default *Code* is being used.
  - If Integrated Clock Control setup executed.
  - If Configuration Feature executed.
  - If Silicon Workaround Capability executed.
  - What state ME is in (e.g. running, sleeping, exception).
- In addition to the individual results listed above, MEManuf also provides a single-line fail indicator at the end of the output (see next page).

No options are needed in the command lines.

MEManuf – Sample Pass & Fail Outputs

Sample 'Pass' Output

```
Ignition firmware is detected on the system.
  Ignition FW Status(FAD_IDX):     Using Data from Runtime Image
  Ignition FW Status(BOOT_IMAGE): Using Code from Runtime Image
  Ignition FW Status(EVENT_LOG):   Event Log is Empty
  Ignition FW Status(ICC_STS):     ICC was executed successfully
  Ignition FW Status(CF):          CF was executed successfully
  Ignition FW Status(SWC):         SWC was executed successfully
  Ignition FW Progress:            Intel® ME is active
```

Note NO fail summary line.

This wording is preliminary, and may change.

Sample 'Fail' Output

```
Ignition firmware is detected on the system.
  Ignition FW Status(FAD_IDX):     Using Data from Runtime Image
  Ignition FW Status(BOOT_IMAGE): Using Code from Factory Default Image
  Ignition FW Status(EVENT_LOG):   Event Log is Empty
  Ignition FW Status(ICC_STS):     ICC was executed successfully
  Ignition FW Status(CF):          CF was executed successfully
  Ignition FW Status(SWC):         SWC was executed successfully
  Ignition FW Progress:            Intel® ME is active
Error 9281: Ignition firmware check was not successful.
```

Note fail summary line in red.

MEManuf – Feature Status Outputs

	Feature Output Message	
FAD_IDX	Using Data from Runtime Image	OK
	Use Data from Factory Default Image	Error
BOOT_IMAGE	Using Code from Runtime Image	OK
	Use Code from Factory Default Image	Error
EVENT_LOG	Event Log is Empty	OK
	Event Log is not Empty	Error
ICC_STS	ICC was executed successfully	OK
	ICC did not execute successfully	Error
CF	CF was executed successfully	OK
	CF did not execute successfully	Error
SWC	SWC was executed successfully	OK
	SWC did not execute successfully	Error
Progress	Intel® ME is active	OK
	Intel® ME is sleeping	OK
	...all other responses...	Error

MEManuf Notes

- MEManuf is a general purpose tool, for all Ibex Peak ME Firmware SKUs, not just Ignition FW.
- It has many options which are not used with Ignition FW.
- When invoked without options, MEManuf detects that Ignition FW is installed, and runs *only* the Ignition-related tests.
- If you invoke MEManuf.exe -? you will see all these options. However, *only* the following options are used for Ignition FW:

-DBG <file>	Display the debug information of the tool or store that in a log file.
-EXP	Show an example of how to use the tool.
-H or -?	Display help screen

Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
  - Summary
  - Pre-Manufacturing
  - During Manufacturing
  - Post Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

Post Manufacturing: OQM/Audit

- MEManuf should be included in OQM/Audit at board and system levels.
- This is important, since the board will boot, and could potentially ship, even if Ignition features are non-operational.

OQM = Outgoing quality monitoring

Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

Post Sales (Field Repair)

- Remember that the Factory Default Image in the ME Region of SPI flash *cannot* be updated using the BIOS.
- If boards returned from the field *must* have their entire flash reprogrammed, remember to consider:
  - Using GPIO33 to override flash protection,
  - or replacing the flash part.
  - If using GPIO33 is the preferred method, make sure that the design supports it.
- Refer to earlier slide (Titled *If Factory Default Image Must be Updated...)* for more information.
- Be sure to use MEManuf after repair.

Action: Know how to reflash entire flash device on each design. If GPIO33 is required, know how to activate it.

Summary – Call to Action

- R&D, Pre-Manufacturing
  - Ensure that SPI flash is connected *directly* to PCH.
  - Only use Flash Image Tool (FITc) to create the entire SPI flash image.
    - Include Flash Descriptor and ME Regions.
    - Lock the Descriptor and ME Regions in flash image.
  - Ensure that full SPI flash erase capability is designed into board (such as using GPIO33), to allow for factory and post sales debug and repair.
    - Without this, parts needing reflashing must be replaced.

- Manufacturing
  - Be aware of flash architecture and flash region location.
  - To update Runtime Image, only use BIOS capability.
  - Use MEManuf to confirm that ME and firmware are OK.
  - Know how to reflash entire flash device on each design.
    - If GPIO33 is required, know how to activate it.
  - Include MEmanuf in Audit/OQM, to catch process changes.
  - Equip field repair sites with full flash erase capability.

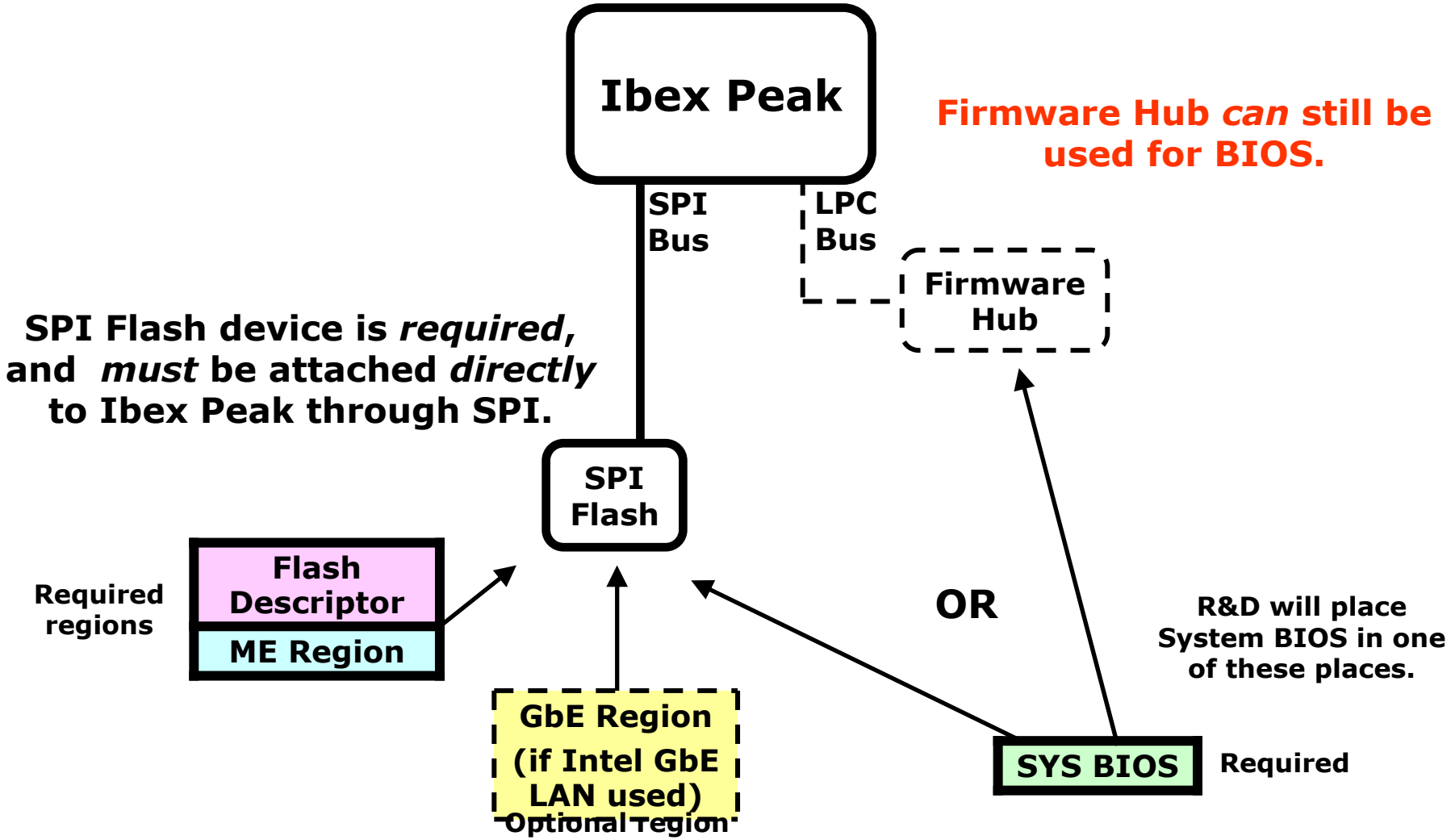
Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

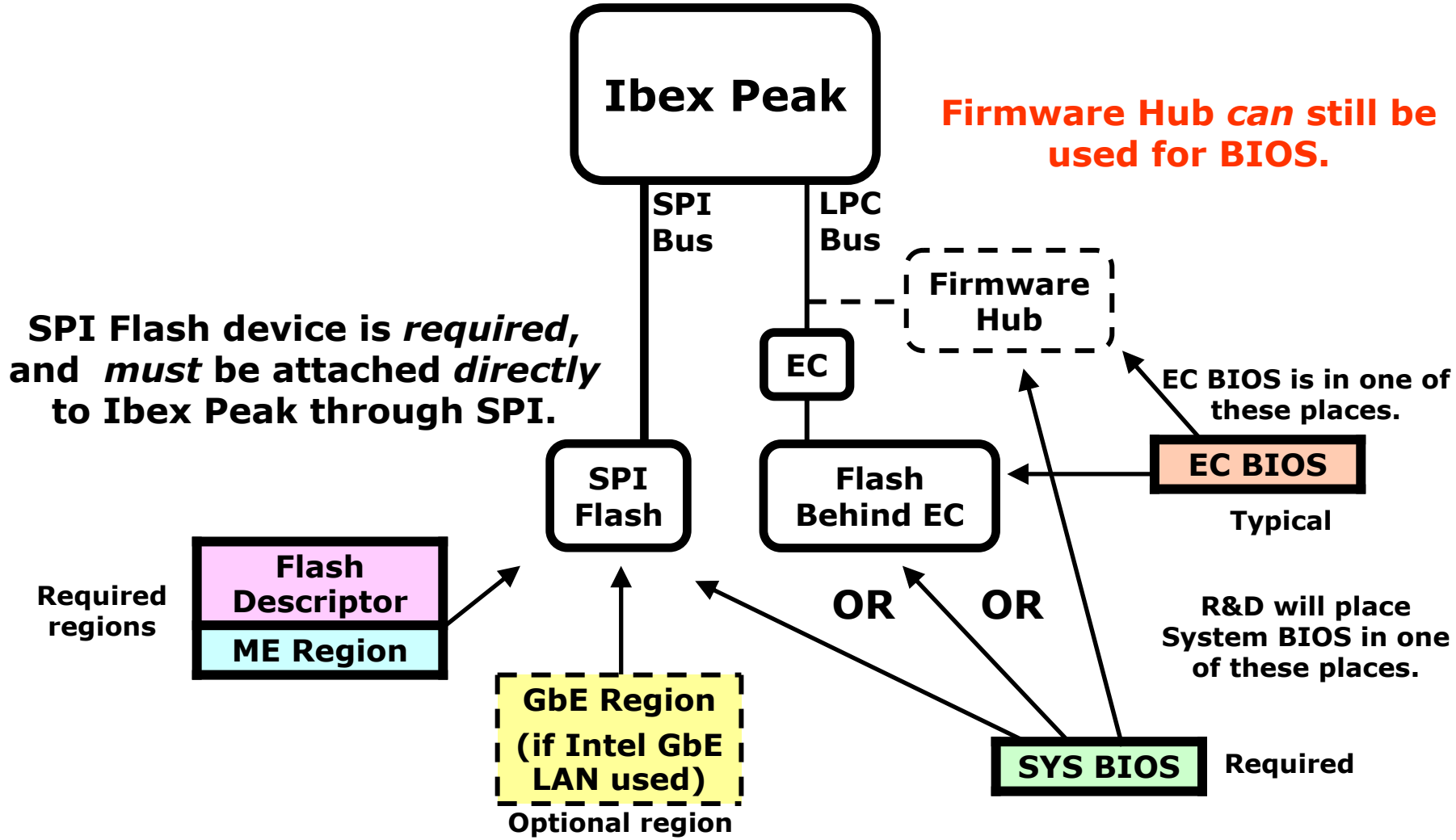
Previous ME FW vs Ignition FW

Previous Platforms with ME FW	Ibex Peak based platforms with Ignition FW
ME uses DRAM in DIMM0.	Uses NO system RAM.
ME reads frequently from SPI ME region.	Reads SPI ME region only before BIOS starts.
ME writes regularly to ME region.	NEVER writes to ME Region.
Up to 32M bit (4M bytes) ME Region size.	1M bit (128K bytes) ME Region size.
HECI	NO HECI once BIOS boots.
Testing requires rebooting.	NO reboots required.
CloseMfg (lock ME) delayed until all ME configuration complete.	Pre-lock ME in FW image with FITc. NO need for CloseMfg.
MEmanuf test, many options depending on situation. Output requires interpretation.	NO options needed to test. Output includes single line fail message.
Must disconnect power (not just shutdown) to reset ME.	ME resets with normal power cycling.
ME code can be updated in field. Requires special tool.	Only Runtime Image can be updated in field, by BIOS update. Factory Default image remains untouched.
FOVs, with need for setting during Mfg.	NO FOVs, all settings done in FITc.
Mfg requires extensive ME/vPro experience/skills.	Less experience/skills required.
Complex OQM process to check ME applications.	Simple OQM process. Same tool as Functional test.

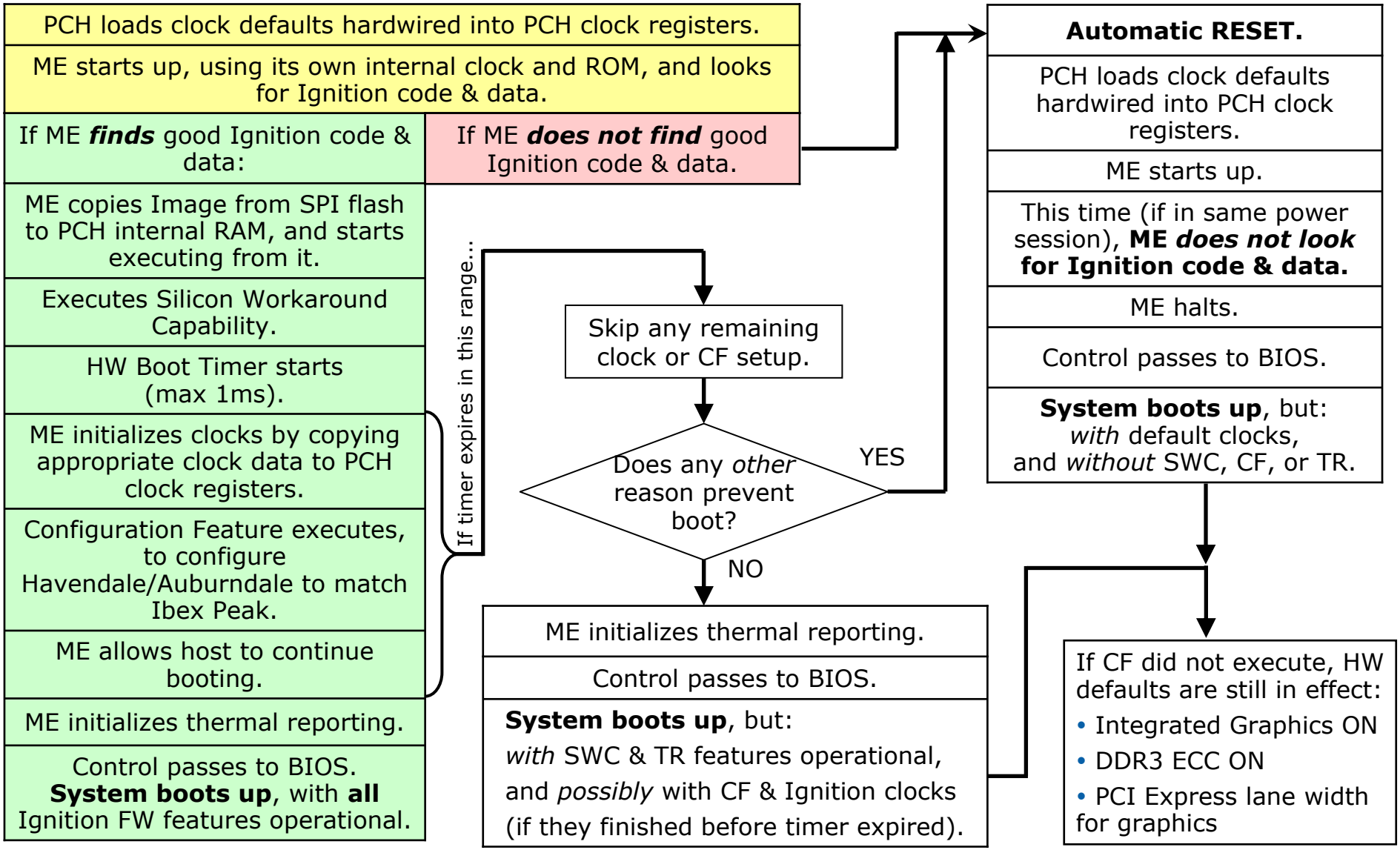
Flash Memory Architecture for Ignition FW - Desktop - With Firmware Hub



Flash Memory Architecture for Ignition FW - Mobile - With Firmware Hub



What Happens at Startup (more detailed view)



Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

References (and how to get documents)

- Intel® Management Engine Ignition Firmware Kit
  - Includes the following items, and many others:
    - Flash Image Tool
    - Flash Programming Tool
    - MEManuf
    - System Tools User Guide
    - FW Bring Up Guide
    - PCH SPI Flash Programming Guide

- To obtain the kit from the Intel Validation Internet Portal (VIP):
  - Logon to https://platformsw.intel.com
    - For access to VIP, contact your Intel representative.
  - Under 'Search for Kits', and 'Select Kits by Platform', select Ibex Peak.
  - Select the latest version of "Intel® Management Engine Ignition Firmware."

Module Table of Contents

- Technology Overview
- Implementing in Manufacturing
- Post Sales (Field Repair)
- Backup
- References (and how to get documents)
- Glossary

Glossary

AMT	Intel® Advanced Management Technology
BIOS	Basic Input/Output System
CF	Configuration Feature (Ignition FW feature)
EC	Embedded Controller
ECC	Error Correction Code
EMI	Electromagnetic Interference
FITc	Flash Image Tool
FOV	Fixed Offset Variable
FPT	Flash Programming Tool
FW	Firmware
FWH	Firmware Hub
GbE	Gigabit Ethernet
HECI	Host Embedded Controller Interface

HW	Hardware
ICT	In-Circuit Test
IT	Information Technology
LAN	Local Area Network
ME	Intel® Management Engine
Mfg	Manufacturing
OQM	Outgoing Quality Monitor
PCH	Processor Control Hub
SKU	Stock Keeping Unit
SPI	Serial Peripheral Interface
SWC	Silicon Workaround Capability (Ignition FW feature)
TR	Thermal Reporting (Ignition FW feature)

How NDA Customers Access Intel® MAS Courses from the Learn.Intel.Com site

- First, turn off your pop-up blocker – this is very important, if you do not, you will not be able to launch any course.
- Visit http://learn.intel.com
- Click the link below the "Sign in" field **to create an Intel® Learning Network user account.**
- Log in your account, and in the top right-hand corner, click on "Profile"
- Click the "Preferences" tab on the left
- In the "Access Codes" field, type in "**ME709**" and click the "Add" button

Note: If you encounter any problems performing these instructions, use the support link under the support and downloads section at the top of the web page.

Customer Manufacturing Advantage Service

developer.intel.com/design/quality/index.htm
learn.intel.com

Prevent,
Manage &
Train