



# **Intel® Xeon® D-1700 Processor Family Intel® Firmware Support Package (Intel® FSP)**

## **Integration Guide**

---

***February 2024***

***Revision 003US***



Intel technologies may require enabled hardware, software or service activation.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described 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.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Performance varies by use, configuration, and other factors. Learn more on the Performance Index site.

Your costs and results may vary.

All product plans and roadmaps are subject to change without notice.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Results have been estimated or simulated.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

Intel contributes to the development of benchmarks by participating in, sponsoring, and/or contributing technical support to various benchmarking groups, including the BenchmarkXPRT Development Community administered by Principled Technologies.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visiting the [Intel Resource and Document Center](#).

© 2024 Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

# Contents

---

<b>1</b>	<b>Introduction</b>	10
1.1	Purpose	10
1.2	Intended Audience	10
1.3	Related Documents	10
1.4	Acronyms and Terminology	10
<b>2</b>	<b>Intel FSP Overview</b>	13
2.1	Technical Overview	13
2.2	Intel FSP Distribution Package	14
2.2.1	Package Layout	14
<b>3</b>	<b>Intel FSP Integration</b>	15
3.1	Assumptions Used in this Document	15
3.2	Boot Flow	15
3.3	Intel FSP Component Extraction	15
3.3.1	Intel FSP Information Header	16
3.4	Intel FSP Image ID and Revision	16
3.5	Intel FSP Global Data	17
3.6	Intel FSP APIs	17
3.6.1	TempRamInit API	17
3.6.2	FspMemoryInit API	18
3.6.3	TempRamExit API	19
3.6.4	FspSiliconInit API	19
3.6.5	NotifyPhase API	20
3.6.5.1	PostPciBusEnumeration Notification	20
3.6.5.2	ReadyToBoot Notification	20
3.6.5.3	EndOfFirmware Notification	20
<b>4</b>	<b>Intel FSP Porting Recommendations</b>	21
4.1	Locking SMI Register	21
4.2	Recommended Settings	21
4.3	FSP_STATUS_RESET_REQUIRED	21
<b>5</b>	<b>Intel FSP Output</b>	23
5.1	SMBIOS Information HOB	23
5.2	IIO Universal Data Structure (IIO_UDS) HOB	24
5.3	Performance Data HOB	27
<b>6</b>	<b>Intel FSP Configuration Firmware File</b>	28
6.1	UPD Data Structure	28
6.1.1	Intel FSP Configuration Data	28
6.1.1.1	Intel FSP-T UPD Structure	28
6.1.1.2	Detailed Description	29
6.1.2	FSP-T UPD Data Region	29
6.1.2.1	PcdRegionTerminator	29
6.1.3	FSP-M UPD Data Region	29
6.1.3.1	StackBase	29
6.1.3.2	StackSize	29
6.1.3.3	BootLoaderTolumSize	30
6.1.3.4	Bootmode	30
6.1.3.5	PcdEnableBiosSsaRMT	30
6.1.3.6	PcdEnableBiosSsaRMTonFCB	30
6.1.3.7	PcdBiosSsaPerBitMargining	30

6.1.3.8	PcdBiosSsaDisplayTables .....	31
6.1.3.9	PcdBiosSsaPerDisplayPlots .....	31
6.1.3.10	PcdBiosSsaLoopCount .....	31
6.1.3.11	PcdBiosSsaBacksideMargining .....	31
6.1.3.12	PcdBiosSsaEarlyReadIdMargining .....	32
6.1.3.13	PcdBiosSsaStepSizeOverride .....	32
6.1.3.14	PcdBiosSsaRxDqs .....	32
6.1.3.15	PcdBiosSsaRxVref .....	32
6.1.3.16	PcdBiosSsaTxDq .....	33
6.1.3.17	PcdBiosSsaTxVref .....	33
6.1.3.18	PcdBiosSsaCmdAll .....	33
6.1.3.19	PcdBiosSsaCmdVref .....	34
6.1.3.20	PcdBiosSsaCtlAll .....	34
6.1.3.21	PcdBiosSsaEridDelay .....	34
6.1.3.22	PcdBiosSsaEridVref .....	35
6.1.3.23	PcdBiosSsaDebugMessages .....	35
6.1.3.24	PcdTccEnable .....	35
6.1.3.25	PcdEccSupport .....	35
6.1.3.26	PcdFastBoot .....	36
6.1.3.27	PcdMemTest .....	36
6.1.3.28	PcdMemTurnaroundOpt .....	36
6.1.3.29	PcdDdrFreq .....	36
6.1.3.30	PcdCommandTiming .....	37
6.1.3.31	PcdCustomRefreshRate .....	37
6.1.3.32	PcdHsuartDevice .....	38
6.1.3.33	PcdHeciCommunication .....	38
6.1.3.34	PcdVtdSupport .....	38
6.1.3.35	PcdPchUsb3Port .....	38
6.1.3.36	PcdPchUsb2Port .....	39
6.1.3.37	PcdPchUsb3PortOc .....	39
6.1.3.38	PcdPchUsb2PortOc .....	40
6.1.3.39	PcdUsb2PeTxiSet .....	40
6.1.3.40	PcdUsb2TxiSet .....	41
6.1.3.41	PcdUsb2PreDeEmp .....	41
6.1.3.42	PcdUsb2PreEmpHalfBit .....	42
6.1.3.43	PcdIIOPciePortBifurcation .....	42
6.1.3.44	PcdIIOPcieRLinkDeEmphasis .....	43
6.1.3.45	PcdIIOPciePort1ADeEmphasis .....	43
6.1.3.46	PcdIIOPciePort1BDeEmphasis .....	43
6.1.3.47	PcdIIOPciePort1CDeEmphasis .....	43
6.1.3.48	PcdIIOPciePort1DDeEmphasis .....	44
6.1.3.49	PcdIIOPcieLinkSpeedRLink .....	44
6.1.3.50	PcdIIOPciePort1ALinkSpeed .....	44
6.1.3.51	PcdIIOPciePort1BLinkSpeed .....	44
6.1.3.52	PcdIIOPciePort1CLinkSpeed .....	45
6.1.3.53	PcdIIOPciePort1DLinkSpeed .....	45
6.1.3.54	PcdIIOPcieRLinkAspm .....	45
6.1.3.55	PcdIIOPciePort1AAspm .....	46
6.1.3.56	PcdIIOPciePort1BAspm .....	46
6.1.3.57	PcdIIOPciePort1CAspm .....	46
6.1.3.58	PcdIIOPciePort1DAspm .....	46
6.1.3.59	PcdBifurcationPcie0 .....	47
6.1.3.60	PcdBifurcationPcie2 .....	47
6.1.3.61	PcdBifurcationPcie1 .....	47
6.1.3.62	PcdDfxWarmResetEliminationEn .....	48
6.1.3.63	PcdSkuClockGeneratorAddress .....	48
6.1.3.64	PcdSkuSSCSecondarySmbusUsed .....	48
6.1.3.65	PcdSkipClockGenerator .....	48
6.1.3.66	PcdEnableClockSpreadSpec .....	49
6.1.3.67	PcdPwrPerfTuning .....	49

6.1.3.68	PcdAltEngPerfBIAS .....	50
6.1.3.69	PcdCustomerRevision .....	50
6.1.3.70	PcdMemoryThermalThrottling .....	50
6.1.3.71	PcdFspDebugPrintErrorLevel .....	50
6.1.3.72	PcdDciEn .....	51
6.1.3.73	PcdDciDbcMode .....	51
6.1.3.74	PcdDciUsb3TypecUfpDbg.....	51
6.1.3.75	PcdPchTraceHubMode .....	51
6.1.3.76	PcdPchTraceHubMemReg0Size.....	52
6.1.3.77	PcdPchTraceHubMemReg1Size.....	52
6.1.3.78	PcdEnableIMR3 .....	52
6.1.3.79	PcdProcessorX2Apic .....	53
6.1.3.80	PcdHyperThreading .....	53
6.1.3.81	PcdPcieHotPlugEnable.....	53
6.1.3.82	PcdIioPcieLinkHPCapable .....	53
6.1.3.83	PcdIioPciePort1AHPCapable.....	54
6.1.3.84	PcdIioPciePort1BHPCapable.....	54
6.1.3.85	PcdIioPciePort1CHPCapable.....	54
6.1.3.86	PcdIioPciePort1DHPCapable .....	54
6.1.3.87	PcdIioPcieLinkHPSurprise.....	55
6.1.3.88	PcdIioPciePort1AHPSurprise .....	55
6.1.3.89	PcdIioPciePort1BHPSurprise .....	55
6.1.3.90	PcdIioPciePort1CHPSurprise .....	55
6.1.3.91	PcdIioPciePort1DHPSurprise .....	56
6.1.3.92	PcdProcessorEistEnable.....	56
6.1.3.93	PcdBootPState .....	56
6.1.3.94	PcdProcessorHWPMEnable .....	56
6.1.3.95	PcdProcessorHWPMInterrupt.....	57
6.1.3.96	PcdProcessorEPPEnable.....	57
6.1.3.97	PcdProcessorEppProfile .....	57
6.1.3.98	PcdPackageCState .....	57
6.1.3.99	PcdProcessorC1eEnable .....	58
6.1.3.100	PcdC2C3TT .....	58
6.1.3.101	PcdC3Enable .....	58
6.1.3.102	PcdC6Enable .....	58
6.1.3.103	PcdMonitorMWait .....	59
6.1.3.104	PcdCStateLatencyCtrlValid0.....	59
6.1.3.105	PcdCStateLatencyCtrlMultiplier0.....	59
6.1.3.106	PcdCStateLatencyCtrlValue0 .....	59
6.1.3.107	PcdCStateLatencyCtrlValid1.....	59
6.1.3.108	PcdCStateLatencyCtrlMultiplier1 .....	60
6.1.3.109	PcdCStateLatencyCtrlValue1 .....	60
6.1.3.110	PcdCStateLatencyCtrlValid2.....	60
6.1.3.111	PcdCStateLatencyCtrlMultiplier2 .....	60
6.1.3.112	PcdCStateLatencyCtrlValue2 .....	60
6.1.3.113	PcdConfigTdpLock .....	61
6.1.3.114	PcdConfigTdpLevel .....	61
6.1.3.115	PcdAvxSupport .....	61
6.1.3.116	PcdAvxLicensePreGrant.....	61
6.1.3.117	PcdAvxIccpLevel .....	62
6.1.3.118	PcdGpssTimer .....	62
6.1.3.119	PcdTStateEnable .....	62
6.1.3.120	PcdEnableProcHot .....	62
6.1.3.121	PcdEnableThermalMonitor .....	63
6.1.3.122	PcdAcExceptionOnSplitLockEnable .....	63
6.1.3.123	PcdPcieAllocatingFlow .....	63
6.1.3.124	PcdIioLlcWaysMask .....	63
6.1.3.125	PcdVMDEnabled .....	64
6.1.3.126	PcdVMDPchPortEnable .....	64
6.1.3.127	PcdVMDPortEnableA .....	64

6.1.3.128	PcdVMDPortEnableB	64
6.1.3.129	PcdVMDPortEnableC	64
6.1.3.130	PcdVMDPortEnableD	65
6.1.3.131	PcdVMDHotPlugEnable	65
6.1.3.132	PcdVMDCfgBarSz	65
6.1.3.133	PcdVMDCfgBarAttr	65
6.1.3.134	PcdVMDMemBarSz1	66
6.1.3.135	PcdVMDMemBar1Attr	66
6.1.3.136	PcdVMDMemBarSz2	66
6.1.3.137	PcdVMDMemBar2Attr	67
6.1.3.138	PcdVMDDirectAssign	67
6.1.3.139	PcdPowerLimit1Enable	67
6.1.3.140	PcdPowerLimit2Enable	67
6.1.3.141	PcdTurboMode	68
6.1.3.142	PcdPcieGlobalAspm	68
6.1.3.143	PcdPchLegacyIoLowLatency	68
6.1.3.144	PcdPchDmiAspm	68
6.1.3.145	PcdDramRaplEnable	69
6.1.3.146	PcdCkeProgramming	69
6.1.3.147	PcdApdEnable	69
6.1.3.148	PcdPpdEnable	69
6.1.3.149	PcdTccDsoTuningEn	70
6.1.3.150	PcdTccSoftwareSramEn	70
6.1.3.151	PcdTccErrorLogEn	70
6.1.3.152	PcdTccStreamCfgBasePreMem	70
6.1.3.153	PcdTccStreamCfgSizePreMem	70
6.1.3.154	PcdTmePtr	71
6.1.3.155	PcdTmeEnable	71
6.1.3.156	PcdMkTmeEnable	71
6.1.3.157	PcdIioPcieMultiVcEnable	71
6.1.3.158	PcdPcieRootPortEn	72
6.1.3.159	PcdSgxEnable	72
6.1.3.160	PcdSgxAutoRegistrationAgent	72
6.1.3.161	PcdSgxQoS	72
6.1.3.162	PcdSgxDebugMode	72
6.1.3.163	PcdSgxLeWr	73
6.1.3.164	PcdSgxLePubKeyHash0	73
6.1.3.165	PcdSgxLePubKeyHash1	73
6.1.3.166	PcdSgxLePubKeyHash2	73
6.1.3.167	PcdSgxLePubKeyHash3	73
6.1.3.168	PcdSpsIccClkSscSetting	73
6.1.3.169	PchPciePIISsc	74
6.1.4	FSP-S UPD Data Region	74
6.1.4.1	PcdEnableSATA	74
6.1.4.2	PcdSATAmode	74
6.1.4.3	PcdSATAInterruptMode	75
6.1.4.4	PcdSATA0PortEnable	75
6.1.4.5	PcdSATA0PortHotplug	75
6.1.4.6	PcdSATA1PortEnable	76
6.1.4.7	PcdSATA1PortHotplug	76
6.1.4.8	PcdSATA2PortEnable	76
6.1.4.9	PcdSATA2PortHotplug	76
6.1.4.10	PcdEmmc	77
6.1.4.11	PcdEmmcHS400Support	77
6.1.4.12	PcdPcieRootPort0LinkSpeed	77
6.1.4.13	PcdPcieRootPort1LinkSpeed	77
6.1.4.14	PcdPcieRootPort2LinkSpeed	78
6.1.4.15	PcdPcieRootPort3LinkSpeed	78
6.1.4.16	PcdPcieRootPort4LinkSpeed	78
6.1.4.17	PcdPcieRootPort5LinkSpeed	78

6.1.4.18	PcdPcieRootPort6LinkSpeed .....	79
6.1.4.19	PcdPcieRootPort7LinkSpeed .....	79
6.1.4.20	PcdPcieRootPort8LinkSpeed .....	79
6.1.4.21	PcdPcieRootPort9LinkSpeed .....	79
6.1.4.22	PcdPcieRootPort10LinkSpeed .....	80
6.1.4.23	PcdPcieRootPort11LinkSpeed .....	80
6.1.4.24	PcdPcieRootPort0Aspm .....	80
6.1.4.25	PcdPcieRootPort1Aspm .....	80
6.1.4.26	PcdPcieRootPort2Aspm .....	81
6.1.4.27	PcdPcieRootPort3Aspm .....	81
6.1.4.28	PcdPcieRootPort4Aspm .....	81
6.1.4.29	PcdPcieRootPort5Aspm .....	82
6.1.4.30	PcdPcieRootPort6Aspm .....	82
6.1.4.31	PcdPcieRootPort7Aspm .....	82
6.1.4.32	PcdPcieRootPort8Aspm .....	83
6.1.4.33	PcdPcieRootPort9Aspm .....	83
6.1.4.34	PcdPcieRootPort10Aspm .....	83
6.1.4.35	PcdPcieRootPort11Aspm .....	84
6.1.4.36	PcdPcieRootPort0ConnectionType .....	84
6.1.4.37	PcdPcieRootPort1ConnectionType .....	84
6.1.4.38	PcdPcieRootPort2ConnectionType .....	84
6.1.4.39	PcdPcieRootPort3ConnectionType .....	85
6.1.4.40	PcdPcieRootPort8ConnectionType .....	85
6.1.4.41	PcdPcieRootPort9ConnectionType .....	85
6.1.4.42	PcdPcieRootPort10ConnectionType .....	85
6.1.4.43	PcdPcieRootPort11ConnectionType .....	86
6.1.4.44	PcdPcieRootPort0HotPlug .....	86
6.1.4.45	PcdPcieRootPort1HotPlug .....	86
6.1.4.46	PcdPcieRootPort2HotPlug .....	86
6.1.4.47	PcdPcieRootPort3HotPlug .....	86
6.1.4.48	PcdPcieRootPort8HotPlug .....	87
6.1.4.49	PcdPcieRootPort9HotPlug .....	87
6.1.4.50	PcdPcieRootPort10HotPlug .....	87
6.1.4.51	PcdPcieRootPort11HotPlug .....	87
6.1.4.52	PcdPcieRootPort4ConnectionType .....	88
6.1.4.53	PcdPcieRootPort5ConnectionType .....	88
6.1.4.54	PcdPcieRootPort6ConnectionType .....	88
6.1.4.55	PcdPcieRootPort7ConnectionType .....	88
6.1.4.56	PcdPcieRootPort4HotPlug .....	89
6.1.4.57	PcdPcieRootPort5HotPlug .....	89
6.1.4.58	PcdPcieRootPort6HotPlug .....	89
6.1.4.59	PcdPcieRootPort7HotPlug .....	89
6.1.4.60	PcdLockDownBiosWpd .....	89
6.1.4.61	PcdLockDownBiosInterface .....	90
6.1.4.62	PcdLockDownGlobalSmi .....	90
6.1.4.63	PcdLockDownBiosLock .....	90
6.1.4.64	PcdSbAccessUnlock .....	90
6.1.4.65	PcdPcieRootPortVppOverride .....	91
6.1.4.66	PcdPcieRootPortVppPort .....	91
6.1.4.67	PcdPcieRootPortVppAddress .....	92
6.1.4.68	PcdPcieRootPortPtmEnable .....	92
6.1.4.69	PcdWriteProtectionEnable .....	93
6.1.4.70	PcdReadProtectionEnable .....	93
6.1.4.71	PcdProtectedRangeLimit .....	93
6.1.4.72	PcdProtectedRangeBase .....	93
6.1.4.73	PcdDevIntConfigPtr .....	93
6.1.4.74	PcdNumOfDevIntConfig .....	94
6.1.4.75	PcdIntConfigPxRcConfig .....	94
6.1.4.76	PcdIntConfigGpioIrqRoute .....	94
6.1.4.77	PcdIntConfigSciIrqSelect .....	94

6.1.4.78	PcdPcieRootPort0L1SubStates .....	94
6.1.4.79	PcdPcieRootPort1L1SubStates .....	95
6.1.4.80	PcdPcieRootPort2L1SubStates .....	95
6.1.4.81	PcdPcieRootPort3L1SubStates .....	95
6.1.4.82	PcdPcieRootPort4L1SubStates .....	95
6.1.4.83	PcdPcieRootPort5L1SubStates .....	96
6.1.4.84	PcdPcieRootPort6L1SubStates .....	96
6.1.4.85	PcdPcieRootPort7L1SubStates .....	96
6.1.4.86	PcdPcieRootPort8L1SubStates .....	97
6.1.4.87	PcdPcieRootPort9L1SubStates .....	97
6.1.4.88	PcdPcieRootPort10L1SubStates .....	97
6.1.4.89	PcdPcieRootPort11L1SubStates .....	98
6.1.4.90	PcdTccCacheCfgBase .....	98
6.1.4.91	PcdTccCacheCfgSize .....	98
6.1.4.92	PcdTccStreamCfgBase .....	98
6.1.4.93	PcdTccStreamCfgSize .....	98
6.1.4.94	PcdTccCrlBinBase .....	98
6.1.4.95	PcdTccCrlBinSize .....	99
6.1.4.96	PcdPchRlinkClockGating .....	99
6.1.4.97	PcdPcieClockGatingEnabled .....	99
6.1.4.98	PcdPchIoApic24119Entries .....	99

## Figures

3-1	Intel FSP Component Layout View .....	16
-----	---------------------------------------	----

## Tables

1-1	Platform and Intel FSP Documentation .....	10
1-2	Acronyms and Terminology .....	10
3-1	Memory Range and Cache Attributes .....	19
4-1	BAR Definitions .....	21
4-2	Intel FSP API Return Status and the Requested Reset Type .....	22



# Revision History

---

Date	Revision	Description
September 2022	001US	Initial Release (PV 001)
December 2022	002US	Introduction of new FSP-M UPDs related to Intel® SGX: PcdSgxEnable PcdSgxAutoRegistrationAgent PcdSgxQoS PcdSgxDebugMode PcdSgxLeWr PcdSgxLePubKeyHash0 PcdSgxLePubKeyHash1 PcdSgxLePubKeyHash2 PcdSgxLePubKeyHash3
February 2024	003US	Introduction of new FSP-M UPDs for enablement and configuration of spread spectrum: PciePIISsc and SpsIccClkSscSetting

§

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to describe the steps required to integrate the Intel® Firmware Support Package (Intel® FSP) for the Intel® Xeon® D-1700 Processor Family into a bootloader solution. This document is a supplement to the *Intel® Firmware Support Package External Architecture Specification v2.1 (Intel® FSP EAS v2.1)*.

## 1.2 Intended Audience

This document is targeted at all platform and system developers who need to consume Intel FSP binaries in their bootloader solutions. This includes, but is not limited to, system BIOS developers, bootloader developers, system integrators, and end users.

## 1.3 Related Documents

**Table 1-1. Platform and Intel FSP Documentation**

Document Name	Reference Number
<i>Intel® Firmware Support Package External Architecture Specification v2.1 (Intel® FSP EAS v2.1)</i>	<a href="https://www.intel.com/content/www/us/en/intelligent-systems/intel-firmware-support-package/intel-fsp-overview.html">https://www.intel.com/content/www/us/en/intelligent-systems/intel-firmware-support-package/intel-fsp-overview.html</a>
Platform Initialization Specification	<a href="http://www.uefi.org/specifications">www.uefi.org/specifications</a>
<i>Binary Configuration Tool (BCT) for Intel® FSP</i>	<a href="https://github.com/IntelFsp/BCT">https://github.com/IntelFsp/BCT</a>

## 1.4 Acronyms and Terminology

**Table 1-2. Acronyms and Terminology**

Acronym	Definition
ACPI	Advanced Configuration and Power Interface
API	Application Programming Interface
ASPM	Active State Power Management
BAR	Base Address Register
BCT	Binary Configuration Tool
BDSM	Base Data of Stolen Memory
BIOS	Basic Input Output System
BSSA	BIOS Shared Software Architecture
BSF	Boot Setting File
BSP	Boot Strap Processor
BWG	BIOS Writer's Guide

**Table 1-2. Acronyms and Terminology**

Acronym	Definition
CAR	Cache-as-RAM
CPU	Central Processing Unit
CRB	Customer Reference Board
DIMM	Dual In-Line Memory Module
DPR	DMA Protected Range
EAS	External Architecture Specification
eMMC	embedded Multi-Media-Card
FIT	Firmware Interface Table
FW	Firmware
GB	Gigabyte
GPIO	General Purpose Input Output
GTT	Graphics Translation Table
HPET	High Precision Event Timer
HWPM	Hardware Power Management
IED	Intel Enhanced Debug
Intel® AVX	Intel® Advanced Vector Extensions
Intel® DCI	Intel® Direct Connect Interface
Intel® FSP	Intel® FSP Firmware Support Package
Intel® FSP API	Intel® FSP Firmware Support Package Interface
IOT	Internal Observation Trace
LLC	Last Level Cache
MOT	Memory Observation Trace
MTRR	Memory Type and Range Register
NS	Network Solution
OpROM	Optional ROM
OS	Operating System
PCH	Platform Controller Hub
PCI	Peripheral Component Interconnect
PCIe	Peripheral Component Interconnect Express
PMC	Power Management Controller
PMRR	Protected Memory Range Reporting
RAM	Random Access Memory
REMAP	Remapped Memory Area
ROM	Read Only Memory
RMT	Rank Margining Test
SBSP	System BSP
SPD	Serial Presence Detect
SMI	System Management Interrupt
SMM	System Management Mode
SoC	System on a Chip

**Table 1-2. Acronyms and Terminology**

Acronym	Definition
SPI	Serial Peripheral Interface
TOLUD	Top of Low Usable Memory
TOUUD	Top of Upper Usable Memory
TSEG	Memory Reserved at the Top of Memory to be used as SMRAM
UPD	Updatable Product Data
USB	Universal Serial Bus
VPD	Virtual Product Data

§

## 2 Intel FSP Overview

---

### 2.1 Technical Overview

The Intel® Firmware Support Package (Intel® FSP) provides chipset and processor initialization in a format that can easily be incorporated into many existing bootloaders.

The Intel FSP performs the necessary initialization steps as documented in the BWG including initialization of the CPU, memory controller, chipset, and certain bus interfaces, if necessary.

The Intel FSP is NOT a stand-alone bootloader; therefore, it needs to be integrated into a host bootloader to carry out other bootloader functions, such as initializing non-Intel components, conducting bus enumeration, and discovering devices in the system and all industry standard initialization.

The Intel FSP binary can be integrated easily into many different bootloaders, such as Coreboot\*, EDKII, and so on, and into the embedded OS directly.

This Intel FSP is compliant to the *Intel® Firmware Support Package External Architecture Specification v2.1 (Intel® FSP EAS v2.1)*.

Below are some required steps for the integration:

- Customizing

The static Intel FSP configuration parameters are part of the Intel FSP binary and can be customized by the BCT tool (refer to [Table 1-1, "Platform and Intel FSP Documentation"](#)).

- Rebasing

The Intel FSP is not Position Independent Code (PIC) and the whole Intel FSP has to be rebased if it is placed at a location that is different from the preferred address during the build process.

- Placing

Once the Intel FSP binary is ready for integration, the bootloader build process needs to be modified to place this Intel FSP binary at the specific rebasing location identified above.

- Interfacing

The boot loader needs to add code to set up the operating environment for the Intel FSP, call the Intel FSP with the correct parameters, and parse the Intel FSP output to retrieve the necessary information returned by the Intel FSP.

## 2.2 Intel FSP Distribution Package

The Intel FSP distribution package contains the following:

- FSP Binary - FspRel.bin
- UPD Data structure definitions - FsptUpd.h, FspmUpd.h, FspUpd.h, FspUpd.h
- BSF File - FspRel.bsf

Refer to [Table 1-1](#), the *Binary Configuration Tool (BCT) for the Intel® FSP* for the download link.

### 2.2.1 Package Layout

- ICELAKE\_D\_FSP\_KIT:
  - IcelakeDEFspBinPkg
    - Docs
      - IcelakeD\_FSP\_Rel\_Notes\_v0\_2.pdf
      - IcelakeD\_FSP\_Integ\_Guide\_v0\_2.pdf
    - FspBin
      - FspRel.bsf (BSF file for configuring the data using BCT tool)
      - FspRel.bin (FSP Release Binary)
    - Include
      - FsptUpd.h, FspmUpd.h, FspUpd.h, and FspUpd.h (FSP UPD structure and related definitions)

## §

## 3 Intel FSP Integration

---

### 3.1 Assumptions Used in this Document

The Intel FSP is built with a preferred base address of 0xffff85000 for FSP-T, 0xffda8000 for FSP-M and 0xffd3a000 for FSP-S; the Intel FSP binary is assumed to be placed at the same address as part of the bootloader build. Users may rebase the Intel FSP binary at a different location with the Binary Configuration Tool (BCT) before integrating to the bootloader.

For other assumptions and conventions, refer to the FSP Interface (FSP API) section of the *Intel® FSP EAS v2.1*.

### 3.2 Boot Flow

Refer to [Table 1-1](#), the *Intel® Firmware Support Package External Architecture Specification v2.1 (Intel® FSP EAS v2.1)* for the boot flow chart.

**Note:** The Intel FSP does not return the reset-required status.

### 3.3 Intel FSP Component Extraction

The Intel FSP image can be split into three different components (FSP-T, FSP-M and FSP-S), and each component can be located at different base addresses according to its execution location.

In the boot flow, there are three different execution stages:

- Execution in ROM
- Execution in temporary memory (Cache-as-RAM)
- Execution in system memory

The three extracted Intel FSP components can be exactly mapped into different execution stages on the Intel FSP boot flow.

- FSP-T executes in ROM.
- FSP-M executes in temporary memory. After the memory is initialized, the generic code like PEI dispatcher and other Intel FSP data is migrated into permanent memory.
- FSP-S executes in memory.

The Intel FSP layout is shown in [Figure 3-1](#). The following base addresses are used by default:

- FSP-T component is set to 0xFFFF85000.
- FSP-M component is set to 0xFFDA8000.
- FSP-S component is set to 0xFFD3A000.

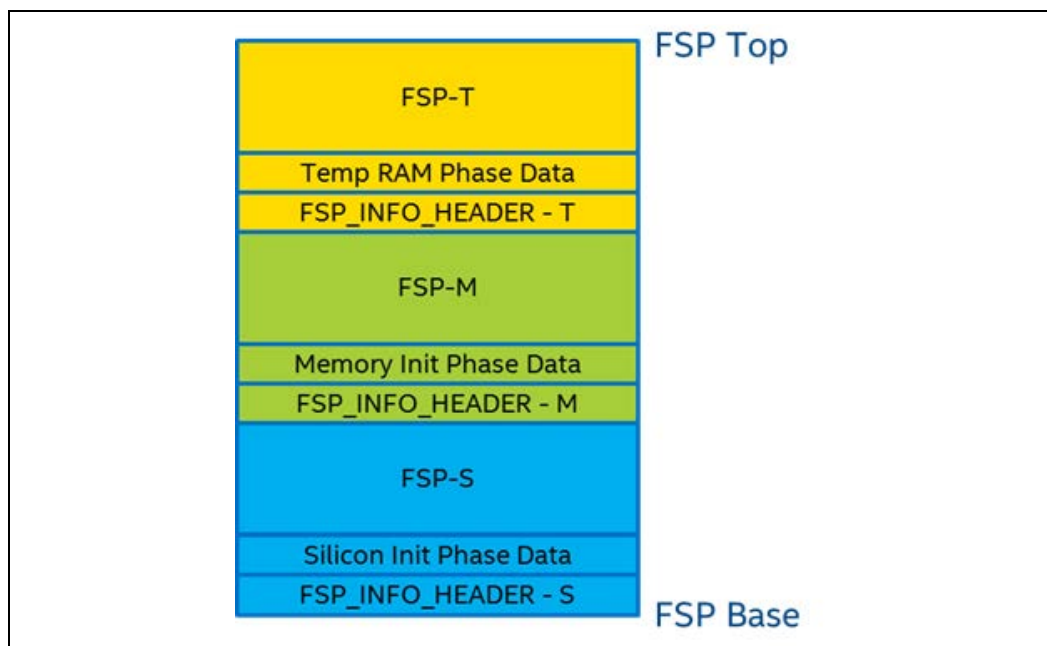
If the Intel FSP component needs to be loaded at a different address, use the BCT tool to rebase it before the integration.

The Intel FSP Binary is released as a single image (.FD extension). Use the Python\* script, `SplitFspBin.py`, to split the FD in to the different Intel FSP components. `SplitFspBin.py` is available at <https://github.com/tianocore/edk2/tree/master/IntelFsp2Pkg/Tools>.

The sample command shown below creates three binaries named after the input Intel FSP binary and appended with “\_M”, “\_S”, and “\_T” respectively.

```
python IntelFsp2Pkg\Tools\SplitFspBin.py split -f <FSP Binary>
```

**Figure 3-1. Intel FSP Component Layout View**



### 3.3.1 Intel FSP Information Header

The Intel FSP has an `FSP_INFO_HEADER` structure embedded in each Intel FSP component. It provides critical information required by the bootloader to successfully interface with the Intel FSP. Refer to [Table 1-1](#), the *Intel® Firmware Support Package External Architecture Specification v2.1 (Intel® FSP EAS v2.1)* for information on the structure of the Intel FSP Information Header listed in [Figure 3-1](#).

## 3.4 Intel FSP Image ID and Revision

The `FSP_INFO_HEADER` structure inside each Intel FSP component also contains an Image Identifier field and an Image Revision field that provide the identification and revision information for the Intel FSP binary. It is important to verify these fields while integrating the Intel FSP as the Intel FSP API parameters could change over different Intel FSP Image identifiers and revisions.



All the Intel FSP FV segments (FSP-T, FSP-M and FSP-S) must have the same Intel FSP Image ID and revision number, using FV segments with different revision numbers in a single Intel FSP image is not valid. The Intel FSP API parameters documented in this integration guide are applicable for the Image ID and revision specified as below.

The current Intel FSP ImageId string in the Intel FSP information header is ICXD-FSP. The ImageRevision field is shown in the Intel FSP release notes.

## 3.5 Intel FSP Global Data

Intel FSP uses some amount of TempRam area to store Intel FSP global data which contains some critical data like pointers to Intel FSP information headers and UPD configuration regions, Intel FSP/Bootloader stack pointers required for stack switching and so on. HPET Timer register PcdGlobalDataPointerAddress is reserved to store address of this global data, and hence bootloader should not use this register for any other purpose. If TempRAM initialization is done by the boot loader, then HPET must be initialized to the base so that access to the register will work fine.

## 3.6 Intel FSP APIs

This release of the Intel FSP supports all APIs required by the *Intel® FSP EAS v2.1*. The Intel FSP information header contains the address offset for these APIs. Refer to the *Intel® FSP EAS v2.1* for register usage and calling conventions. Any usage not described by the specification is described in the individual sections below.

The below sections highlight any changes that are specific to this release.

### 3.6.1 TempRamInit API

Refer to the `TempRamInit` section in the *Intel® FSP EAS v2.1* for complete details including the prototype, parameters, and return value details for this API.

If the bootloader initializes the Temporary RAM (CAR) and chipset BARs, calling this API should be skipped.

`TempRamInit` does basic early initialization; primarily, setting up temporary RAM using cache. The API returns a temporary memory data region that can be used by the bootloader with ECX pointing to the beginning of the temporary memory and EDX pointing to the end of temporary memory + 1. The total temporary RAM currently available is given by `PcdTemporaryRamSize` starting from the base address given by `PcdTemporaryRamBase`. Out of total temporary memory available, the last bytes of space reserved by the Intel FSP for `TempRamInit` (`PcdFspReservedBufferSize`) if the temporary RAM initialization is done by Intel FSP, and the remaining space from `TemporaryRamBase` (ECX) to `TemporaryRamBase+TemporaryRamSize-FspReservedBufferSize` (EDX) is available for both the bootloader and Intel FSP binary.

The temporary memory data region returned by this Intel FSP release is from `0xFE80_0000` (ECX) to `0xFE87_FF00` (EDX).

`TempRamInit` also sets up the code caching of the region passed `CodeCacheBase` and `CodeCacheLength`, which are input parameters to `TempRamInit` API. If 0 is passed in for `CodeCacheBase`, the base used will be 4 GB - 1 - length to be code cached instead of starting from `CodeCacheBase`.

**Note:** The Entire FSPM binary must be cached, this is a requirement for this Intel FSP. Ensure that `CodeCacheBase` and `CodeCacheLength` encapsulate FSPM.

**Note:** Programming MTRR `CodeCacheLength` will be reduced if the SKU LLC size is smaller than the requested.

It is a requirement for the firmware to have a Firmware Interface Table (FIT), which contains pointers to each microcode update. The microcode update is loaded for all logical processors before the reset vector. If more than one microcode update for the CPU is present, the microcode update with the latest revision is loaded.

`FSPT_UPD.MicrocodeRegionBase` and `FSPT_UPD.MicrocodeRegionLength` are input parameters to the `TempRamInit` API. If these values are 0, the Intel FSP will not attempt to update microcode. However, if a valid region is passed and a newer microcode update revision is in this region, it will be loaded by the Intel FSP.

**Note:** As it is a requirement for the firmware (that is, the bootloader) to have a FIT entry for microcode, the microcode will be loaded before the reset vector. Therefore, it is recommended to set `FSPT_UPD.MicrocodeRegionBase` and `FSPT_UPD.MicrocodeRegionLength` values to 0, so Intel FSP will skip microcode loading as it has already been done.

MTRRs are programmed to the default values to have the following memory map.

Memory Range	Cache Attribute
0xFE800000 – 0xFE880000	Write back
<code>CodeCacheBase</code> – <code>CodeCacheLength</code>	Write Protect

## 3.6.2 FspMemoryInit API

Refer to the `FspMemoryInit` section in the *Intel® FSP EAS v2.1* for the prototype, parameters, and return value details for this API.

The `FspmUpdPtr` is a pointer to the `FSPM_UPD` structure which is described in header file `FspmUpd.h`.

The minimum Intel FSP stack size required for this revision of the Intel FSP is 320 KB, stack base is 0x0 by default.

The bootloader must pass the valid CAR region for Intel FSP stack use through these UPDs:

- `FSPM_UPD.FspmArchUpd.StackBase`
- `FSPM_UPD.FspmArchUpd.StackSize`

**Note:** Certain platforms might need some GPIOs to be initialized prior to the memory initialization. In this case, the bootloader needs to configure the required GPIO pins properly before calling into `FspMemoryInit`. For example, to read SPD data, the SMBUS pins must be configured properly.

### 3.6.3 TempRamExit API

Refer to the *TempRamExit* section in the *Intel® FSP EAS v2.1* for the prototype, parameters, and return value details for this API.

If the bootloader initialized the temporary RAM (CAR) as well as the chipset BARs and skipped calling the *TempRamInit* API, then the bootloader **MUST** skip calling this API, and the bootloader should tear down the temporary memory area setup in the cache and to bring the cache back to a normal mode of operation.

This revision of FSP does not have any fields/structures to pass as a parameter for this API, Pass Null for *TempRamExitParamPtr*.

At the end of the *TempRamExit* execution, the original code and data caching are disabled. The Intel FSP reconfigures the following MTRRs for performance optimization.

**Table 3-1. Memory Range and Cache Attributes**

Memory Range	Cache Attribute
0x00000000 – 0x0009FFFF	Write back
0x000C0000 – Top of Low Memory	Write back
0xFF000000 – 0xFFFFFFFF (Flash region)	Write protect
0x1000000000 – Top of High Memory	Write back

The bootloader can reconfigure the MTRRs immediately after this API call.

### 3.6.4 FspSiliconInit API

Refer to the *FspSiliconInit* section in the *Intel® FSP EAS v2.1* for the prototype, parameters, and return value details for this API.

The *FspUpdPtr* is the pointer to the *FSPS\_UPD* structure which is described in header file *FspUpd.h*.

The boot loader is responsible for programming the MTRRs for SBSP as needed after *TempRamExit* but before entering *FspSiliconInit*. If the MTRRs are not programmed properly, boot performance might be impacted.

The region of 0x9F000 - 0x9FFFF is used by *FspSiliconInit* for starting APs. If this data is important to the boot loader, then the boot loader needs to preserve it before calling *FspSiliconInit*.

It is a requirement for the boot loader to have Firmware Interface Table (FIT), which contains pointers to each microcode. The microcode is loaded for all cores before reset vector. If more than one microcode update for the CPU is present, the latest revision is loaded.

*MicrocodeRegionBase* and *MicrocodeRegionLength* are both input parameters to *TempRamInit* and *UPD* for *SiliconInit* API. *UPD* has priority and will be searched for a later revision than *TempRamInit*. If *MicrocodeRegionBase* and *MicrocodeRegionLength* values are 0, Intel FSP will not attempt to update the microcode. If a microcode region is passed, and if a later revision of microcode is present in this region, Intel FSP will load it.



The PCH required initialization is done for the following HECI, USB, HSIO, PCI Express\*, and Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d).

### 3.6.5 NotifyPhase API

Refer to the `NotifyPhase` section in the *Intel® FSP EAS v2.1* for the prototype, parameters, and return value details for this API.

**Note:** It is **REQUIRED** that all the supported `NotifyPhase` APIs are called and in the recommended order.

#### 3.6.5.1 PostPciBusEnumeration Notification

The bootloader calls this phase, `EnumInitPhaseAfterPciEnumeration`, after the PCI bus enumeration but before execution of third-party code such as option ROMs. Currently, no special operation is done in this phase, but in the future updates, programming may be added in this phase.

#### 3.6.5.2 ReadyToBoot Notification

The bootloader calls this phase, `EnumInitPhaseReadyToBoot`, before giving control to the OS loader. It includes some final initialization steps recommended by the BWG, including power management settings, sends the Intel® Management Engine (Intel® ME) message End of Post (EOP), and switching devices into the ACPI mode if required, and so on.

#### 3.6.5.3 EndOfFirmware Notification

This phase `EnumInitEndOfFirmware` is to be called before the firmware/preboot environment transfers management of all system resources to the OS or next level execution environment. It includes final locking of chipset registers.

## §

# 4 Intel FSP Porting Recommendations

This chapter discusses the recommendations when porting with Intel FSP.

## 4.1 Locking SMI Register

Since the global SMI bit is recommended to be locked before any third-party OpROM execution and highly dependent on platform code implementation after SMM configuration, Intel FSP will not lock it by default.

The boot loader is responsible for locking the following registers after the SMM configuration.

- Set AcpiBase + 0x30[0] to 1b to enable global SMI.
- Set PMC PCI offset A0h [4] = 1b to lock SMI.

## 4.2 Recommended Settings

PMC PciCfgSpace is not PCI compliant. Intel FSP will hide the PMC controller to avoid external software or OS from corrupting the BAR addresses. Intel FSP will program the PMC controller I/O and MMIO BARs with the following addresses. Use these addresses in the bootloader code instead of reading from the PMC controller.

**Table 4-1. BAR Definitions**

Register	Values
ABASE	0x00000500
PWRMBASE	0xFE000000
PCIEXBAR_BASE_ADDRESS	0x80000000

## 4.3 FSP\_STATUS\_RESET\_REQUIRED

As per the *Intel® FSP EAS v2.1*, any reset required in the Intel FSP flow will be reported as return status `FSP_STATUS_RESET_REQUIREDx` by the API. It is the boot loader's responsibility to reset the system according to the reset type requested.

The table below specifies the return status returned by the Intel FSP API and the requested reset type.



**Table 4-2. Intel FSP API Return Status and the Requested Reset Type**

FSP_STATUS_RESET_REQUIRED Code	Reset Type Requested
0x40000001	Cold Reset
0x40000002	Warm Reset
0x40000003	Global Reset - Puts the system to Global reset through HECI or Full Reset through PCH
0x40000004	Reserved
0x40000005	Reserved
0x40000006	Reserved
0x40000007	Reserved
0x40000008	Reserved

§

# 5 Intel FSP Output

The Intel FSP builds a series of data structures called the Hand-Off-Blocks (HOBs) as it progresses through initializing the silicon.

Refer to [Table 1-1, Platform Initialization Specification](#). Refer to the Intel® FSP Output chapter of the *Intel® FSP EAS v2.1* for details about Intel FSP Architectural HOBs.

The section below describes the HOBs implemented in the Intel FSP that are not covered in the previous two specifications.

## 5.1 SMBIOS Information HOB

The Intel FSP will report the SMBIOS through a HOB with the GUID below. This information can be consumed by the bootloader to produce the SMBIOS tables.

```
#define FSP_SMBIOS_MEMORY_INFO_HOB_GUID \
{ 0x1a1108c, 0x9dee, 0x4984, { 0x88, 0xc3, 0xee, 0xe8, 0xc4,
0x9e, 0xfb, 0x89 } };

#define MAX_CHANNELS_NUM 2
#define MAX_DIMMS_NUM 2
typedef struct {
    UINT8 DimmId;
    UINT32 SizeInMb;
    UINT16 MfgId;
    /* Module part number for DRR3 is 18 bytes but DRR4 is 20
    bytes as per JEDEC Spec, so reserving 20 bytes */
    UINT8 ModulePartNum[20];
} DIMM_INFO;
typedef struct {
    UINT8 ChannelId;
    UINT8 DimmCount;
    DIMM_INFO DimmInfo[MAX_DIMMS_NUM];
} CHANNEL_INFO;
typedef struct {
    UINT8 Revision;
    UINT8 DataWidth;
    /** As defined in SMBIOS 3.0 spec
    Section 7.18.2 and Table 75
    **/
    UINT16 MemoryType;
    UINT16 MemoryFrequencyInMHz;
    /** As defined in SMBIOS 3.0 spec
    Section 7.17.3 and Table 72
    **/
    UINT8 ErrorCorrectionType;
    UINT8 ChannelCount;
    CHANNEL_INFO ChannelInfo[MAX_CHANNELS_NUM];
} FSP_SMBIOS_MEMORY_INFO;
```

## 5.2 IIO Universal Data Structure (IIO\_UDS) HOB

The Intel FSP will report the IIO\_UDS through a HOB with the GUID below. This information can be consumed by the bootloader to initialize IIO devices. IIO\_UDS is compound by a series of nested structures shown below. Refer to `FspmUpd.h`.

```
#define IIO_UNIVERSAL_DATA_GUID { 0x7FF396A1, 0xEE7D, 0x431E, { 0xBA,
0x53, 0x8F, 0xCA, 0x12, 0x7C, 0x44, 0xC0 } }

#define BL_MAX_CHA_MAP 4
#define BL_MAX_FW_KTI_PORTS 3
#define BL_MAX_SOCKET 1
#define BL_NUMBER_PORTS_PER_SOCKET 5
#define BL_TYPE_MAX_MMIO_BAR 11
#define BL_MAX_IMC 2
#define BL_MAX_CH 2
#define BL_MAX_LOGIC_IIO_STACK 8
#define BL_MAX_IIO_STACK 6
#define BL_MaxIIO BL_MAX_SOCKET
#define BL_MC_MAX_NODE (BL_MAX_SOCKET * BL_MAX_IMC)

typedef struct {
    UINT8 Valid; // TRUE, if the link is valid
    (i.e reached normal operation)
    UINT8 PeerSocId; // Socket ID
    UINT8 PeerSocType; // Socket Type (0 - CPU; 1 - IIO)
    UINT8 PeerPort; // Port of the peer socket
} BL_QPI_PEER_DATA;

typedef struct {
    UINT8 Valid;
    UINT32 MmioBar[BL_TYPE_MAX_MMIO_BAR];
    UINT8 PcieSegment;
    BL_UINT64_STRUCT SegMmcfgBase;
    UINT16 stackPresentBitmap;
    UINT16 M2PciePresentBitmap;
    UINT8 TotM3Kti;
    UINT8 TotCha;
    UINT32 ChaList[BL_MAX_CHA_MAP];
    UINT32 SocId;
    BL_QPI_PEER_DATA PeerInfo[BL_MAX_FW_KTI_PORTS]; // QPI LEP
info
} BL_QPI_CPU_DATA;

typedef struct {
    UINT8 Valid;
    UINT8 SocId;
    BL_QPI_PEER_DATA PeerInfo[BL_MAX_SOCKET]; // QPI LEP info
} BL_QPI_IIO_DATA;

typedef struct {
    UINT8 Device;
```



```

        UINT8          Function;
    } BL_IIO_PORT_INFO;

typedef struct {
    BL_IIO_PORT_INFO      PortInfo[BL_NUMBER_PORTS_PER_SOCKET];
} BL_IIO_DMI_PCIE_INFO;

typedef struct _BL_STACK_RES {
    UINT8          Personality;
    UINT8          BusBase;
    UINT8          BusLimit;
    UINT16         PciResourceIoBase;
    UINT16         PciResourceIoLimit;
    UINT32         IoApicBase;
    UINT32         IoApicLimit;
    UINT32         Mmio32Base;           // Base of low MMIO
configured for this stack in memory map
    UINT32         Mmio32Limit;          // Limit of low MMIO
configured for this stack in memory map
    UINT64         Mmio64Base;           // Base of high MMIO
configured for this stack in memory map
    UINT64         Mmio64Limit;          // Limit of high MMIO
configured for this stack in memory map
    UINT32         PciResourceMem32Base; // Base of low MMIO
resource available for PCI devices
    UINT32         PciResourceMem32Limit; // Limit of low MMIO
resource available for PCI devices
    UINT64         PciResourceMem64Base; // Base of high MMIO
resource available for PCI devices
    UINT64         PciResourceMem64Limit; // Limit of high MMIO
resource available for PCI devices
    UINT32         VtdBarAddress;
    UINT32         Mmio32MinSize;        // Minimum required size
of MMIO32 resource needed for this stack
} BL_STACK_RES;

typedef struct {
    UINT8          Valid;
    UINT8          SocketID;             // Socket ID of the IIO (0..3)
    UINT8          BusBase;
    UINT8          BusLimit;
    UINT16         PciResourceIoBase;
    UINT16         PciResourceIoLimit;
    UINT32         IoApicBase;
    UINT32         IoApicLimit;
    UINT32         Mmio32Base;           // Base of low MMIO
configured for this socket in memory map
    UINT32         Mmio32Limit;          // Limit of low MMIO
configured for this socket in memory map
    UINT64         Mmio64Base;           // Base of high MMIO
configured for this socket in memory map
    UINT64         Mmio64Limit;          // Limit of high MMIO
configured for this socket in memory map
    BL_STACK_RES   StackRes[BL_MAX_LOGIC_IIO_STACK];
    UINT32         RcBaseAddress;

```

```

    BL_IIO_DMI_PCIE_INFO      PcieInfo;
    UINT8                     DmaDeviceCount;
} BL_IIO_RESOURCE_INSTANCE;

typedef struct {
    UINT16                    PlatGlobalIoBase;           // Global IO Base
    UINT16                    PlatGlobalIoLimit;          // Global IO Limit
    UINT32                    PlatGlobalMmio32Base;       // Global Mmio32 base
    UINT32                    PlatGlobalMmio32Limit;      // Global Mmio32 limit
    UINT64                    PlatGlobalMmio64Base;       // Global Mmio64 Base
[43:0]
    UINT64                    PlatGlobalMmio64Limit;      // Global Mmio64 Limit
[43:0]
    BL_QPI_CPU_DATA           CpuQpiInfo[BL_MAX_SOCKET]; // QPI related info
per CPU
    BL_QPI_IIO_DATA           IioQpiInfo[BL_MAX_SOCKET]; // QPI related info
per IIO
    UINT32                    MemTsegSize;
    UINT32                    MemIedSize;
    UINT64                    PciExpressBase;
    UINT32                    PciExpressSize;
    UINT32                    MemTolm;
    BL_IIO_RESOURCE_INSTANCE  IIO_resource[BL_MAX_SOCKET];
    UINT8                     numofIIO;
    UINT8                     MaxBusNumber;
    UINT32                    packageBspApicID[BL_MAX_SOCKET]; // This data
array is valid only for SBSP, not for non-SBSP CPUs. <AS> for CpuSv
    UINT8                     EVMode;
    UINT8                     Pci64BitResourceAllocation;
    UINT8                     SkuPersonality[BL_MAX_SOCKET];
    UINT8                     VMDStackEnable[BL_MaxIIO][BL_MAX_IIO_STACK];
    UINT16                    IoGranularity;
    UINT32                    Mmio32Granularity;
    BL_UINT64_STRUCT          Mmio64Granularity;
    UINT8                     RemoteRequestThreshold;    //5370389
    UINT32                    UboxMmioSize;
    UINT32                    MaxAddressBits;
} BL_PLATFORM_DATA;

typedef struct {
    UINT8                     CurrentUpiLinkSpeed; // Current programmed UPI
Link speed (Slow/Full speed mode)
    UINT8                     CurrentUpiLinkFrequency; // Current requested
UPI Link frequency (in GT)
    UINT8                     OutKtiCpuSktHotPlugEn;           // 0 -
Disabled, 1 - Enabled for PM X2APIC
    UINT32                    OutKtiPerLinkL1En[BL_MAX_SOCKET]; // output
kti link enabled status for PM
    UINT8                     IsocEnable;
    UINT32                    meRequestedSize; // Size of the memory range
requested by ME FW, in MB
    UINT32                    ieRequestedSize; // Size of the memory range
requested by IE FW, in MB
    UINT8                     DmiVcl;
    UINT8                     DmiVcm;

```

```

        UINT32                CpuPCPSInfo;
        UINT8                 cpuSubType;
        UINT8                 SystemRasType;
        UINT8                 numCpus;
/ 1,..4. Total number of CPU packages installed and detected (1..4)by QPI
RC
        UINT16                tolmLimit;
        UINT32                tohmLimit;
        BL_RC_VERSION         RcVersion;
        BOOLEAN               MsrTraceEnable;
        UINT8                 DdrXoverMode;           // DDR 2.2 Mode
        // For RAS
        UINT8                 bootMode;
        UINT8                 OutClusterOnDieEn; // Whether RC enabled COD
support
        UINT8                 OutSncEn;
        UINT8                 OutNumOfCluster;
        UINT8                 imcEnabled[BL_MAX_SOCKET][BL_MAX_IMC];
        UINT16                LlcSizeReg;
        UINT8                 chEnabled[BL_MAX_SOCKET][BL_MAX_CH];
        UINT8                 memNode[BL_MC_MAX_NODE];
        UINT8                 IoDcMode;
        UINT8                 DfxRstCplBitsEn;
    } BL_SYSTEM_STATUS;

typedef struct {
        BL_PLATFORM_DATA      PlatformData;
        BL_SYSTEM_STATUS      SystemStatus;
    } BL_IIO_UDS;

```

## 5.3 Performance Data HOB

The Intel FSP will report the Performance Data through a HOB with the GUID below. The Performance Data HOB is defined as below. Refer to `FspmUpd.h`. This HOB API entry and exit point timestamps, so the bootloader can consume it and report the Intel FSP performance.

```

#define gPerformanceDataHobGuid { 0xC1FF44B6, 0xC94E, 0x478B, { 0x9C,
0xFD, 0x12, 0x14, 0x85, 0x19, 0x42, 0x78 } }

typedef struct {
        UINT32                PerfSig;
        UINT16                PerfLen;
        UINT16                Reserved4;
        UINT32                PerfIdx;
        UINT64                PerfData[32];
    } FSP_PERF_INFO;

```

### §

## 6 Intel FSP Configuration Firmware File

---

The Intel FSP binary contains a configurable data region that is used by the Intel FSP during the initialization. Refer to the Intel FSP Configuration Firmware File chapter of the *Intel® FSP EAS v2.1* for details.

**Note:**

A stack in temporary memory is used to store the UPD data structure. This UPD data structure is copied, updated, and then passed to the Intel FSP API. When permanent memory is initialized, the Intel FSP sets up a new stack in the permanent memory and tears down the temporary memory. However, the Intel FSP saves the whole bootloader temporary memory region in a GUID HOB. The bootloader can access the data in the temporary memory by parsing this HOB. The migrated temporary memory contains an identical copy of the original data. Pointers to data stored in the temporary memory needs to be updated to the location of the same data in the permanent memory.

### 6.1 UPD Data Structure

The UPD data structure and related structure definitions are provided in the `FspUpd.h`, `FsptUpd.h`, `FspmUpd.h` and `FspsUpd.h` file in the release package. The basic information for each option is provided in the BCT configuration file. Use the BCT tool to load this BSF file and get the detailed configuration option information.

#### 6.1.1 Intel FSP Configuration Data

##### 6.1.1.1 Intel FSP-T UPD Structure

FSP-T Core UPD Configuration is described in the FSP-T UPD Structure section of the *Intel® FSP EAS v2.1*.

```
#include <FsptUpd.h>
typedef struct {
    UINT32                MicrocodeRegionBase;
    UINT32                MicrocodeRegionLength;
    UINT32                CodeRegionBase;
    UINT32                CodeRegionLength;
    UINT8                 Reserved1[16];
} FSPT_CORE_UPD;
```

### 6.1.1.2 Detailed Description

FSP-T Core UPD Configuration

The documentation for this structure was generated from the following file:

`FsptUpd.h`

**MicrocodeRegionBase:** Base address of the microcode region. This address must be 16 byte aligned.

**MicrocodeRegionLength:** Length of the microcode region. The length must be total size of all patches or 0xFFFFFFFF if auto size detection is desired.

**CodeRegionBase:** Base address of the cacheable flash region.

**CodeRegionLength:** Length of the cacheable flash region. A size of 0 indicates that no code caching is desired.

## 6.1.2 FSP-T UPD Data Region

This UPD data region can be configured statically by the BCT tool in the same way as the VPD data region, but can also be overridden by the bootloader at runtime. This provides more flexibility for the bootloader to customize these options dynamically as needed.

### 6.1.2.1 PcdRegionTerminator

This field has a value of 0x55AA indicating the end of UPD data. **DO NOT MODIFY.**

## 6.1.3 FSP-M UPD Data Region

The UPD parameters that are part of the MemoryInitUpd and consumed by the FspMemoryInit API are described below.

### 6.1.3.1 StackBase

Stack base for Intel FSP use. This Intel FSP is using the bootloader's Stack, so this StackBase will be provided by the bootloader for the FSP Heap.

See the *Intel® Firmware Support Package External Architecture Specification v2.1*, refer to [Table 1-1](#) for more information.

Default Value = 0x0

### 6.1.3.2 StackSize

To pass the stack size for Intel FSP use, the Bootloader can programmatically get the Intel FSP requested StackSize by using the defaults in the FSP-M component. This Intel FSP is using the bootloader's stack, this StackSize is the minimum Heap size expected by this revision of the Intel FSP. See the *Intel® Firmware Support Package External Architecture Specification v2.1*, refer to [Table 1-1](#) for more information.

Default Value = 0x00020000

### 6.1.3.3 BootLoaderTolumSize

To pass Bootloader Tolum size. See the *Intel® Firmware Support Package External Architecture Specification v2.1*, refer to [Table 1-1](#) for more information.

Default Value = 0x00000000

### 6.1.3.4 Bootmode

To maintain Bootmode details. See the *Intel® Firmware Support Package External Architecture Specification v2.1*, refer to [Table 1-1](#) for more information.

Default Value = 0x00000000

### 6.1.3.5 PcdEnableBiosSsaRMT

Enable BIOS Shared Software Architecture (BSSA) Rank Margin Tool (RMT): Enables and disables SSA RMT. See the *Idaville DDR EV Tools and Methodology* for recommendations, refer to [Table 1-1](#) for information.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.6 PcdEnableBiosSsaRMTonFCB

Enable BIOS SSA RMT on Fast Cold Boot. Enables/Disables SSA RMT on a Fast Cold Boot.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.7 PcdBiosSsaPerBitMargining

Enable RMT per Bit Margining: Enables and disables margining on a per bitgranularity.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.8 PcdBiosSsaDisplayTables

Enable SSA Tables Display. Enables/Disables displaying results as tables.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.9 PcdBiosSsaPerDisplayPlots

Enable SSA Plot Display. Enables/Disables the display of per bit results as plots.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.10 PcdBiosSsaLoopCount

Loop count for rank test. Exponential loop count for a single rank test

Valid range: 0x01 ~ 0x1F

Examples:

Value = 0x02, 0x03, 0x14

Example for invalid values:

Invalid value: 0x00, 0xF1

Default Value = 0x10

### 6.1.3.11 PcdBiosSsaBacksideMargining

Enable Backside Margining. Enables/Disables margin test on the register or buffer backside. See the *Idaville DDR EV Tools and Methodology* for recommendations, refer to [Table 1-1](#) for information.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### 6.1.3.12 PcdBiosSsaEarlyReadIdMargining

Enable Early Read ID Margining (ERID). Enables/Disables PMem Early Read Id Test. See the *Idaville DDR EV Tools and Methodology* for recommendations on ERID margining, refer to [Table 1-1](#) for information.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### 6.1.3.13 PcdBiosSsaStepSizeOverride

Enable Step Size Override. Enables/Disables overriding the default step sizes.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### 6.1.3.14 PcdBiosSsaRxDqs

Step size of RxDqs. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

#### 6.1.3.15 PcdBiosSsaRxVref

Step size of RxVrefs. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1



#### **6.1.3.16 PcdBiosSsaTxDq**

Step size of TxDqs. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

#### **6.1.3.17 PcdBiosSsaTxVref**

Step size of TxVrefs. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

#### **6.1.3.18 PcdBiosSsaCmdAll**

Step size of CmdAll. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

### **6.1.3.19 PcdBiosSsaCmdVref**

Step size of CmdVref. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

### **6.1.3.20 PcdBiosSsaCtlAll**

Step size of CtlAll. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

### **6.1.3.21 PcdBiosSsaEridDelay**

Step size of EridDelay. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

#### **6.1.3.22 PcdBiosSsaEridVref**

Step size of EridVref. This option is valid only if 'Enable Step size override' is enabled.

Valid inputs are:

1 - "1" - Auto

2 - "2"

4 - "4"

8 - "8"

Default Value = 1

#### **6.1.3.23 PcdBiosSsaDebugMessages**

Enable SSA RMT Debug Message. Enables the BSSA RMT debug messages.

Valid inputs are:

2 - "Disable"

5 - "Enable"

Default Value = 2

#### **6.1.3.24 PcdTccEnable**

TCC Enable. Enable or Disable Intel® Time Coordinated Computing (Intel® TCC) features.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.3.25 PcdEccSupport**

ECC Support. Enable/disable ECC Support.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.26 PcdFastBoot

Fast Boot. Enable/Disable Fast Boot.

Valid inputs are:

0 - "Disabled"

1 - "Enabled"

Default Value = 1

### 6.1.3.27 PcdMemTest

Memory Test. Enable/Disable Memory Test.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.28 PcdMemTurnaroundOpt

Memory Turnaround Time Optimization. Enable/Disable Memory turnaround time optimization. When enabled, calculate and program optimal TA (DR, DS, DD, SG, SR) setting based on formulas and training results.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.29 PcdDdrFreq

Memory Frequency. Set DDR Memory Frequency Limit

Valid inputs are:

0 - "AUTO". With this option, Intel FSP will look for the maximum supported frequency which is common for all DIMMs and the cores.

4 - "1200"

5 - "1333"

6 - "1400"

7 - "1600"

8 - "1800"

9 - "1866"

11 - "2133"

12 - "2200"

13 - "2400"

14 - "2600"

15 - "2666"

16 - "2800"

17 - " 2933"

Default Value = 0x9

### 6.1.3.30 PcdCommandTiming

Memory Command Timing. Select the desired memory controller command timing. 3N timing mode is only supported and intended for use by memory training only. It is not intended for normal operation.

Valid inputs are:

0 - "Auto" With this option, Intel FSP will determine the Command Timing based on the DIMM type.

1 - "1N"

2 - "2N"

3 - "3N"

Default Value = 0x0

### 6.1.3.31 PcdCustomRefreshRate

Memory Custom Refresh Rate. Set Desired rate in 0.1x units of the standard 7.8 usec interval. The valid range is 20 – 80 (for instance, 2x to 8x).

Valid range: 20 ~ 80

Examples:

Valid Values: 22, 24, 40,36

Example for invalid values:

Invalid value: 21,23,45,31

Default Value = 20

### 6.1.3.32 PcdHsuartDevice

HSUART Device: Select the PCI High Speed UART (HSUART) device for serial port. It is expected that the bootloader will configure the PCI HSUART device for serial port prior to calling the FspInitEntry API. The bootloader should update this UPD with the function number of the HSUART device so that the Intel FSP can output to the serial port. The supported device functions are 0, 1, and 2. The Intel FSP defaults to function 0 if this UPD is not configured with one of the supported function numbers.

Valid inputs are:

0 - "HSUART0"

1 - "HSUART1"

2 - "HSUART2"

Default Value = 0

### 6.1.3.33 PcdHeciCommunication

Intel ME HECI Communication. Enable/Disable Intel ME HECI Communication.

Valid inputs are:

0 - "Disabled"

1 - "Enabled"

Default Value = 1

### 6.1.3.34 PcdVtdSupport

Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d)

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.35 PcdPchUsb3Port

Enable USB3 Port: Enable/Disable per USB3 Ports. One byte for each port, byte0 for port0, byte1 for port1 and so on.

Valid values per port are 0x00 for disable, 0x01 for enable.

**Warning:** Invalid values are not handled and lead to indeterminate configurations.

Examples:

Enable port 0, disable other ports: Value = 0x00000001.

Enable port 1, disable other ports: Value = 0x00000100.

Enable port 2, disable other ports: Value = 0x00010000.

Enable port 3, disable other ports: Value = 0x01000000.

Invalid values:

Values other than 0 or 1 per byte are invalid.

Example for invalid values:

Invalid value: 0x00000002

Invalid value: 0x10101010

Invalid value: 0x0f0a0c09

Default Value = 0x01010101 (All ports enabled)

### 6.1.3.36 PcdPchUsb2Port

Enable USB2 Ports. Enable/Disable per USB2 Ports. One byte for each port, byte0 for port0, byte1 for port1 and so on.

Valid values per port are 0x00 for disable, 0x01 for enable.

**Warning:** Invalid values are not handled and lead to indeterminate configurations.

Examples:

Enable port 0, disable other ports: Value = 0x00000001.

Enable port 1, disable other ports: Value = 0x00000100.

Enable port 2, disable other ports: Value = 0x00010000.

Enable port 3, disable other ports: Value = 0x01000000.

Invalid values:

Values other than 0 or 1 per byte are invalid.

Example for invalid values:

Invalid value: 0x00000002, 0x10101010, 0x0f0a0c09

Default Value = 0x01010101 (All ports enabled)

### 6.1.3.37 PcdPchUsb3PortOc

Enable USB3 Port Over Current Configuration. Enable over current pin assignment per USB3 port. 0xFF mean skip over current pin. One byte for each port, byte0 for port0, byte1 for port1 and so on.

Valid range: 0 ~ 0xFFFFFFFF

Default Value = 0xFFFFFFFF

### 6.1.3.38 PcdPchUsb2PortOc

Enable USB2 Port Over Current Configuration. Enable over current pin assignment per USB2 port. 0xFF mean skip over current pin. One byte for each port, byte0 for port0, byte1 for port1 and so on.

Valid values per port are 0x00 for pin 0, 0x01 for pin1, 0x02 for pin2, 0x03 for pin3, 0x04 for pin4, 0x05 for pin5, 0x06 for pin6, 0x07 for pin7, 0xff for skip.

Default Value = 0xFFFF0000

**Warning:** Invalid values are not handled and lead to indeterminated configurations.

Examples:

- For port 0, set pin0, skip others: Value = 0xFFFFF00.
- For port 1, set pin1, skip others: Value = 0xFFFF01FF.
- For port 2 and 3, set pin 2 and 3, skip others: Value = 0x0304FFFF.

Invalid examples:

- Invalid value: 0x10101010
- Invalid value: 0x08090a0b
- Invalid value: 0x0f0a0c09

### 6.1.3.39 PcdUsb2PeTxiSet

USB2 Per Port HS Pre-emphasis Bias. One byte for each port, byte0 for port0, byte1 for port1 and so on. One byte for each port.

Valid values are:

000b - 0 mV

001b - 11.25 mV

010b - 16.9 mV

011b - 28.15 mV

100b - 28.15 mV

101b - 39.35 mV

110b - 45 mV

111b - 56.3 mV

Valid range: 0 ~ 0x07070707

Examples:

Values: 0x00000706, 0x06060707

**Warning:** Invalid values are not handled and lead to indeterminate configurations.

Example for invalid values:



Invalid Values: 0x07270757, 0x17067007

Default Value = 0x04040404

#### 6.1.3.40 PcdUsb2TxiSet

USB2 Per Port HS Transmitter Bias. One byte for each port, byte0 for port0, byte1 for port1 and so on. One byte for each port.

Valid values are:

000b - 0 mV

001b - 11.25 mV

010b - 16.9 mV

011b - 28.15 mV

100b - 28.15 mV

101b - 39.35 mV

110b - 45 mV

111b - 56.3 mV

Valid range: 0 ~ 0x07070707

Examples:

Valid Values: 0x00000706, 0x06060707

**Warning:** Invalid values are not handled and lead to indeterminate configurations.

Example for invalid values:

Invalid Values: 0x07270757, 0x17067007

Default Value = 0x0 (0mV in all USB2 ports)

#### 6.1.3.41 PcdUsb2PreDeEmp

USB2 Per Port HS Transmitter Emphasis. One byte for each port.

Valid values are:

00b - Emphasis OFF

01b - De-emphasis ON

10b - Pre-emphasis ON

11b - Pre-emphasis & De-emphasis ON

Valid range: 0 ~ 0x03030303



Examples:

Valid Values: 0x00020101, 0x03020301

**Warning:** Invalid values are not handled and lead to indeterminate configurations.

Example for invalid values:

Invalid Values: 0x30102003, 0x20300301

Default Value = 0x03030303 (Pre-emphasis & De-emphasis ON in all USB2 ports)

#### 6.1.3.42 PcdUsb2PreEmpHalfBit

USB2 Per Port Half Bit Pre-emphasis. One byte for each port.

Valid values are:

1b - half-bit pre-emphasis

0b - full-bit pre-emphasis

Valid range: 0 ~ 0x01010101

Examples:

Valid Values: 0x00000101, 0x01010000

**Warning:** Invalid values are not handled and lead to indeterminate configurations.

Invalid values:

Values other than 0 or 1 per byte are invalid.

Example for invalid values:

Invalid Values: 0x00100111, 0x00001010

Default Value = 0x0 (full-bit pre-emphasis for all USB2 ports)

#### 6.1.3.43 PcdIIOPciePortBifurcation

IIO PCIe Port 1 Bifurcation. IIO PCI Express port bifurcation for selected slot(s). See the EDS for PCIe\* Port Bifurcation register data.

Valid inputs are:

0xFF - "Auto". With this option, the Intel FSP will try to setup the bifurcation based on the platform.

0 - "X4X4X4X4"

1 - "X4X4X8"

2 - "X8X4X4"

3 - "X8X8"

4 - "X16"

Default Value = 0xFF

#### **6.1.3.44 PcdIIoPcieRLinkDeEmphasis**

IIO PCIe R-Link DeEmphasis. Desired DeEmphasis level for IIO PCIe R-Link

Valid inputs are:

0 - "6 dB"

1 - "3.5 dB"

Default Value = 1

#### **6.1.3.45 PcdIIoPciePort1ADeEmphasis**

IIO PCIe Port 1A DeEmphasis. Desired DeEmphasis level for IIO PCIe Port 1A

Valid inputs are:

0 - "6 dB"

1 - "3.5 dB"

Default Value = 1

#### **6.1.3.46 PcdIIoPciePort1BDeEmphasis**

IIO PCIe Port 1B DeEmphasis. Desired DeEmphasis level for IIO PCIe Port 1B

Valid inputs are:

0 - "6 dB"

1 - "3.5 dB"

Default Value = 1

#### **6.1.3.47 PcdIIoPciePort1CDeEmphasis**

IIO PCIe Port 1C DeEmphasis. Desired DeEmphasis level for IIO PCIe Port 1C

Valid inputs are:

0 - "6 dB"

1 - "3.5 dB"

Default Value = 1



#### **6.1.3.48 PcdIIoPciePort1DDeEmphasis**

IIO PCIe Port 1D DeEmphasis. Desired DeEmphasis level for IIO PCIe Port 1D

Valid inputs are:

0 - "6 dB"

1 - "3.5 dB"

Default Value = 1

#### **6.1.3.49 PcdIIoPcieLinkSpeedRLink**

IIO PCIe R-Link Link Speed. Desired Link Speed for IIO PCIe R-Link

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup based on the device.

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 0

#### **6.1.3.50 PcdIIoPciePort1ALinkSpeed**

IIO PCIe Port 1A Link Speed. Desired Link Speed for IIO PCIe Port 1A

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup based on the device.

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

4 - "GEN4"

Default Value = 0

#### **6.1.3.51 PcdIIoPciePort1BLinkSpeed**

IIO PCIe Port 1B Link Speed. Desired Link Speed for IIO PCIe Port 1B

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup based on the device.

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

4 - "GEN4"

Default Value = 0

### **6.1.3.52 PcdIIOpciePort1CLinkSpeed**

IIO PCIe Port 1C Link Speed. Desired Link Speed for IIO PCIe Port 1C

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup based on the device.

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

4 - "GEN4"

Default Value = 0

### **6.1.3.53 PcdIIOpciePort1DLinkSpeed**

IIO PCIe Port 1D Link Speed. Desired Link Speed for IIO PCIe Port 1D

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup based on the device.

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

4 - "GEN4"

Default Value = 0

### **6.1.3.54 PcdIIOpcieRLinkAspm**

IIO PCIe R-Link Aspm. Desired Active state power management settings for IIO PCIe R-Link

Valid inputs are:

0 - "Disabled"

4 - "Auto". With this option, the Intel FSP will try to setup based on the ASPM support.

Default Value = 4

#### **6.1.3.55 PcdIIoPciePort1AAspm**

IIO PCIe Port 1A Aspm. Desired Active state power management settings for IIO PCIe Port 1A

Valid inputs are:

0 - "Disabled"

4 - "Auto". With this option, the Intel FSP will try to setup based on the ASPM support.

Default Value = 4

#### **6.1.3.56 PcdIIoPciePort1BAspm**

IIO PCIe Port 1B Aspm. Desired Active state power management settings for IIO PCIe Port 1B

Valid inputs are:

0 - "Disabled"

4 - "Auto". With this option, the Intel FSP will try to setup based on the ASPM support.

Default Value = 4

#### **6.1.3.57 PcdIIoPciePort1CAspm**

IIO PCIe Port 1C Aspm. Desired Active state power management settings for IIO PCIe Port 1C

Valid inputs are:

0 - "Disabled"

4 - "Auto". With this option, the Intel FSP will try to setup based on the ASPM support.

Default Value = 4

#### **6.1.3.58 PcdIIoPciePort1DAspm**

IIO PCIe Port 1D Aspm. Desired Active state power management settings for IIO PCIe Port 1D

Valid inputs are:

0 - "Disabled"

4 - "Auto". With this option, the Intel FSP will try to setup based on the ASPM support.

Default Value = 4

### 6.1.3.59 PcdBifurcationPcie0

PCH PCIe Controller 0 Bifurcation. Configure PCI Express controller 0 bifurcation.

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup the bifurcation based on the platform.

5 - "4x2"

6 - "1x4 2x2"

7 - "2x2 1x4"

8 - "2x4"

9 - "1x8"

Default Value = 0

### 6.1.3.60 PcdBifurcationPcie2

PCH PCIe Controller 2 Bifurcation. Configure PCI Express controller 2 bifurcation.

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup the bifurcation based on the platform.

5 - "4x2"

6 - "1x4 2x2"

7 - "2x2 1x4"

8 - "2x4"

9 - "1x8"

Default Value = 0

### 6.1.3.61 PcdBifurcationPcie1

PCH PCIe Controller 1 Bifurcation. Configure PCI Express controller 1 bifurcation.

Valid inputs are:

0 - "Auto". With this option, the Intel FSP will try to setup the bifurcation based on the platform.

5 - "4x2"

6 - "1x4 2x2"

7 - "2x2 1x4"

8 - "2x4"

9 - "1x8"

Default Value = 0

#### **6.1.3.62 PcdDfxWarmResetEliminationEn**

Warm Reset Elimination Enable. Warm Reset Elimination Enable or Disable.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.3.63 PcdSkuClockGeneratorAddress**

Clock Generator Address. Set SKU Clock generator address. Valid field should be set. Skip Clock Generator should be set to Disable.

Valid range: 0 ~ 0xff

Default Value = 0xD4

#### **6.1.3.64 PcdSkuSSCSecondarySmbusUsed**

SSC Secondary SMBus use. Enable or Disable SSC Secondary SMBus use. Skip Clock Generator should be set to Disable.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.3.65 PcdSkipClockGenerator**

Skip Clock Generator. Enable or disable to skip clock generator configuration.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0



### 6.1.3.66 PcdEnableClockSpreadSpec

SSC Enable for Host. Enable Spread Spectrum Control for the Host (CPU) PCIe High-Speed Root Ports. Skip Clock Generator should be set to Disable.

Valid inputs are:

- 0 - "Disable"
- 1 - "Enable SSC with 0.25 spread"
- 2 - "Enable SSC with 0.5 spread"

Default Value = 0

PcdPciePortClkGateEnable

PCIe port Clock Gating. Enable/Disable PCI-E Clock Gating for each port.

First byte represents Clock gating for port 1A, second byte for port 2A, respectively for each PCI-E Port. Each byte can have one of the following values:  
0x00(Disable)~0x01(Enable)

Valid range: 0 ~ 0x01

Default Value =

[illegible]

### 6.1.3.67 PcdPwrPerfTuning

Power Performance Tuning. Specifies the EPB controller.

Valid inputs are:

- 0 - "OS Controls EPB"
- 1 - "BIOS Controls EPB"
- 2 - "PECI Controls EPB"

In OS mode, IA32\_ENERGY\_PERF\_BIAS is used. Option 0 is valid only if PcdProcessorHWPMEnable is 0/1.

In BIOS mode, ENERGY\_PERF\_BIAS\_CONFIG is used.

In PECI mode, PCS53 is used.

Default Value = 0

#### 6.1.3.68 PcdAltEngPerfBIAS

ENERGY\_PERF\_BIAS\_CFG mode. This PCD is valid if Power Performance Tuning is set to 1 (BIOS Controls EPB). Use input from ENERGY\_PERF\_BIAS\_CONFIG mode selection. PERF/Balanced Perf/Balanced Power/Power

Valid inputs are:

0 - "Performance"

7 - "Balanced Performance"

8 - "Balanced Power"

15 - "Power"

Default Value = 7

#### 6.1.3.69 PcdCustomerRevision

Customer Revision. The Customer can set this revision string for their own purpose.

Default Value = 0x76,0x65,0x72,0x73,0x69,0x6F,0x6E,0x20,0x78,0x78,0x78,0x00

#### 6.1.3.70 PcdMemoryThermalThrottling

Memory Thermal Throttling. Enable/disable Memory Thermal Throttling.

Valid inputs are:

0 - "Disabled"

2 - "Enabled"

Default Value = 2

#### 6.1.3.71 PcdFspDebugPrintErrorLevel

FSP Debug Print Level. Select the FSP debug print level.

Valid inputs are:

0 - "NO DEBUG"

1 - "MIN DEBUG"

2 - "MED DEBUG"

3 - "VERBOSE DEBUG"

Default Value = 2

#### **6.1.3.72 PcdDciEn**

Intel® DCI Enable. Enable/Disable DCI.

Valid inputs are:

0 - "Disabled"

1 - "Enabled"

Default Value = 0

#### **6.1.3.73 PcdDciDbcMode**

USB DbC Enable Mode. USB Debug mode can be selected.

Valid inputs are:

0 - "Disabled" (Disable USB DbC)

1 - "USB2" (Enable USB2 DbC only)

2 - "USB3" (Enable USB3 DbC only)

3 - "Both" (Enable both USB2/USB3 Dbc)

4 - "NoChange (default)" (Comply with HW value, keep it unchanged)

Default Value = 4

#### **6.1.3.74 PcdDciUsb3TypecUfpDbg**

USB3 Type-C UFP2DFP Debug Support. USB3 UFP2DFP Debug support.

Valid inputs are:

0 - "Disabled"

1 - "Enabled"

2 - "No Change"

Default Value = 2

#### **6.1.3.75 PcdPchTraceHubMode**

PCH Trace Hub Enable Mode. Select Host or Target for Trace Hub debugger tool.

Valid inputs are:

0 - "Disable"

1 - "Target debugger"

2 - "Host debugger"

Default Value = 0

#### **6.1.3.76 PcdPchTraceHubMemReg0Size**

PCH TH Mem Buffer Size 0. Select size of memory region 0 buffer.

Valid inputs are:

0 - "None/OS"

1 - "1 MB"

2 - "8 MB"

3 - "64 MB"

4 - "128 MB"

5 - "256 MB"

6 - "512 MB"

Default Value = 0

#### **6.1.3.77 PcdPchTraceHubMemReg1Size**

PCH TH Mem Buffer Size 1. Select size of memory region 1 buffer.

Valid inputs are:

0 - "None/OS"

1 - "1 MB"

2 - "8 MB"

3 - "64 MB"

4 - "128 MB"

5 - "256 MB"

6 - "512 MB"

Default Value = 0

#### **6.1.3.78 PcdEnableIMR3**

Isolated Memory Range 3 (IMR3) Enable: Enable/Disable IMR3.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.79 PcdProcessorX2Apic**

Processor X2APIC Enable. Enable/Disable Processor X2APIC.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.80 PcdHyperThreading**

Intel® Hyper-Threading Technology (Intel® HT Technology) Enable/Disable: Enable or Disable Intel Hyper-Threading Technology.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.81 PcdPcieHotPlugEnable**

PCIe Hot Plug Enable. Enable/Disable PCIe Hot Plug.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.82 PcdIIOpcieLinkHPCapable**

IIO PCIe R-Link Hot Plug Capable. Hot Plug Capable for IIO PCIe R-Link.

Valid inputs are:

0 - "Disable"

1 - "Enable"

2 - "Auto". With this option, the Intel FSP will try to setup the Hot Plug Capable on IIO PCIe R-Link.

Default Value = 2



#### **6.1.3.83 PcdIIO PCIe Port 1A HPCapable**

IIO PCIe Port 1A Hot Plug Capable. Hot Plug Capable for IIO PCIe Port 1A

Valid inputs are:

0 - "Disable"

1 - "Enable"

2 - "Auto". With this option, the Intel FSP will try to setup the Hot Plug Capable on IIO PCIe Port 1A.

Default Value = 2

#### **6.1.3.84 PcdIIO PCIe Port 1B HPCapable**

IIO PCIe Port 1B Hot Plug Capable. Hot Plug Capable for IIO PCIe Port 1B

Valid inputs are:

0 - "Disable"

1 - "Enable"

2 - "Auto". With this option, the Intel FSP will try to setup the Hot Plug Capable on IIO PCIe Port 1B.

Default Value = 2

#### **6.1.3.85 PcdIIO PCIe Port 1C HPCapable**

IIO PCIe Port 1C Hot Plug Capable. Hot Plug Capable for IIO PCIe Port 1C

Valid inputs are:

0 - "Disable"

1 - "Enable"

2 - "Auto". With this option, the Intel FSP will try to setup the Hot Plug Capable on IIO PCIe Port 1C.

Default Value = 2

#### **6.1.3.86 PcdIIO PCIe Port 1D HPCapable**

IIO PCIe Port 1D Hot Plug Capable. Hot Plug Capable for IIO PCIe Port 1D

Valid inputs are:

0 - "Disable"

1 - "Enable"

2 - "Auto". With this option, the Intel FSP will try to setup the Hot Plug Capable on IIO PCIe Port 1D.

Default Value = 2

### **6.1.3.87 PcdIIO PCIe Link HP Surprise**

IIO PCIe R-Link Hot Plug Surprise. Enable/Disable Hot Plug Capable Surprise for IIO PCIe R-Link.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.88 PcdIIO PCIe Port 1A HP Surprise**

IIO PCIe Port 1A Hot Plug Surprise. Enable/Disable Hot Plug Capable Surprise for IIO PCIe Port 1A.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.89 PcdIIO PCIe Port 1B HP Surprise**

IIO PCIe Port 1B Hot Plug Surprise. Enable/Disable Hot Plug Capable Surprise for IIO PCIe Port 1B.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.90 PcdIIO PCIe Port 1C HP Surprise**

IIO PCIe Port 1C Hot Plug Surprise. Enable/Disable Hot Plug Capable Surprise for IIO PCIe Port 1C.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.3.91 PcdIIoPciePort1DHPSurprise**

IIO PCIe Port 1D Hot Plug Surprise. Enable/Disable Hot Plug Capable Surprise for IIO PCIe Port 1D.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.3.92 PcdProcessorEistEnable**

Enhanced Intel SpeedStep<sup>®</sup> Technology (P-states). Enable/Disable EIST (P-States).

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.3.93 PcdBootPState**

Boot performance mode. Select Boot Performance State. EIST should be enabled.

Valid inputs are:

0 – “Max Performance”

1 – “Max Efficient”

2- “Set by Intel Node Manager”

Default Value = 0

#### **6.1.3.94 PcdProcessorHWPMEnable**

Hardware P-States. Select Hardware P-State control.

Valid inputs are:

0 – “Disable”

1 – “Native Mode”

2- “Out of Band Mode”

3- “Native Mode with No Legacy Support”

Default Value = 1



### 6.1.3.95 PcdProcessorHWPMInterrupt

Hardware PM Interrupt. Enable/Disable Hardware PM Interrupt. Hardware P-States should be native mode.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.96 PcdProcessorEPPEnable

Energy Performance Preference (EPP) Enable. Enable/Disable EPP. Hardware P-States should not be disabled.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.97 PcdProcessorEppProfile

EPP profile. Select HWPM Profile (EPP). Hardware P-States should be OOB and EPP should be enabled.

Valid inputs are:

0 - "Performance"

128 - "Balanced Performance"

192 - "Balanced Power"

255 - "Power"

Default Value = 128

### 6.1.3.98 PcdPackageCState

Package C State. Package C State limit.

Valid inputs are:

0 - "C0/C1 state"

1 - "C2 state"

2 - "C6(non-Retention) state"

255 - "Auto". With this option, the Intel FSP will try to setup the Value based on C state limit.



Default Value = 255

#### **6.1.3.99 PcdProcessorC1eEnable**

Enhanced Halt State (C1E). Enable/Disable Core C1E auto promotion Control.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.3.100 PcdC2C3TT**

C2C3TT. C2 to C3 Transition Timer. Default = 0, means [AUTO].

Valid range: 0 ~ 255

Default Value = 0

#### **6.1.3.101 PcdC3Enable**

CPU C3 report. Enable/Disable CPU C3 (ACPI C2) report to OS.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.3.102 PcdC6Enable**

CPU C6 report. Enable/Disable CPU C6 (ACPI C3) report to OS.

Valid inputs are:

0 – “Disable”

1 – “Enable”

255 – “Auto”. With this option, the Intel FSP will try to setup CPU based on ACPI C3.

Default Value = 255

### 6.1.3.103 PcdMonitorMWait

Enable Monitor MWAIT. Enable/Disable Monitor and MWAIT instructions

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.104 PcdCStateLatencyCtrlValid0

C State Latency Control VALID [0]. Enable/Disable validity of the Value field in this register.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.105 PcdCStateLatencyCtrlMultiplier0

C State Latency Control MULTIPLIER [0]. Indicates the unit of measurement that is defined for the Value field in this register. Valid field should be set.

Valid range: 0x0 ~ 0x7

Default Value = 0x00

### 6.1.3.106 PcdCStateLatencyCtrlValue0

C State Latency Control VALUE[0]. The Interrupt Response Time Limit is given in units defined in the Multiplier field of this register. Valid field should be set.

Valid range: 0x0 ~ 0x3ff

Default Value = 0x0000

### 6.1.3.107 PcdCStateLatencyCtrlValid1

C State Latency Control VALID[1]. Enable/Disable validity of the Value field in this register.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.108 PcdCStateLatencyCtrlMultiplier1

C State Latency Control MULTIPLIER[1]. Indicates the unit of measurement that is defined for the Value field in this register. Valid field should be set.

Valid range: 0x0 ~ 0x7

Default Value = 0x00

### 6.1.3.109 PcdCStateLatencyCtrlValue1

C State Latency Control VALUE[1]. The Interrupt Response Time Limit is given in units defined in the Multiplier field of this register. Valid field should be set.

Valid range: 0x0 ~ 0x3ff

Default Value = 0x0000

### 6.1.3.110 PcdCStateLatencyCtrlValid2

C State Latency Control VALID[2]. Enable/Disable validity of the Value field in this register.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.111 PcdCStateLatencyCtrlMultiplier2

C State Latency Control MULTIPLIER[2]. Indicates the unit of measurement that is defined for the Value field in this register. Valid field should be set.

Valid range: 0x0 ~ 0x7

Default Value = 0x00

### 6.1.3.112 PcdCStateLatencyCtrlValue2

C State Latency Control VALUE [2]. The Interrupt Response Time Limit is given in units defined in the Multiplier field of this register. Valid field should be set.

Valid range: 0x0 ~ 0x3ff

Default Value = 0x0000

### **6.1.3.113 PcdConfigTdpLock**

Config Thermal Design Power (TDP) Lock. Config TDP CONTROL Lock Bit. EIST should be enabled.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.114 PcdConfigTdpLevel**

Intel® Advanced Vector Extensions (Intel® AVX) P1. Intel AVX P1 level selection. EIST should be enabled.

Valid inputs are:

0 - "Normal"

1 - "Level 1"

2 - "Level 2"

Default Value = 0

### **6.1.3.115 PcdAvxSupport**

Intel AVX Support. Enable/Disable Intel AVX/2/3 instructions. Applicable to only certain SKUs - OC and HEDT.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.116 PcdAvxLicensePreGrant**

Intel AVX License Pre-Grant Override. Enables Intel AVX ICCP pre-grant level override.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.117 PcdAvxIccpLevel

Intel AVX ICCP pre-grant level. Pre-grants an Intel AVX level to the core. Base frequency is not updated. Intel AVX License Pre-Grant Override should be enabled.

Valid inputs are:

1 – “128 Heavy”

2 – “256 Light”

3 – “256 Heavy”

4 – “512 Light”

5 – “512 Heavy”

Default Value = 1

### 6.1.3.118 PcdGpssTimer

GPSS timer. P-state change hysteresis time window.

Valid inputs are:

0 – “0 us”

5 – “50 us”

50 – “500 us”

Default Value = 50

### 6.1.3.119 PcdTStateEnable

Software Controlled T-States. Enable/Disable Software Controlled T-States.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.120 PcdEnableProcHot

PROCHOT Modes. When a processor thermal sensor trips, the PROCHOT# will be driven.

Valid inputs are:

0 – “Output-only”

1 – “Disable”

2 – “Both Input and Output”

3 – “Input-only”

Default Value = 3

### 6.1.3.121 PcdEnableThermalMonitor

Thermal Monitor. Enable/Disable Thermal Monitor.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.122 PcdAcExceptionOnSplitLockEnable

AC Exception On Split Lock. Enable or Disable AC (Alignment Check) Exception On Split Lock.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.123 PcdPcieAllocatingFlow

PCIe Allocating Write Flows. Select Vc0/VCp write selection for all CPU PCIe ports.

Valid inputs are:

0x00 – “Non-Allocating”

0x01 – “Allocating”

Default Value = 0x01

### 6.1.3.124 PcdIioLlcWaysMask

IIO LLC Ways [19:0] (Hex). MSR CBO\_SLICE0\_CR\_IIO\_LLC\_WAYS bitmask

Valid range: 0x0 ~ 0xfffff

Default Value = 0x00000

### 6.1.3.125 PcdVMDEnabled

Enable/Disable Intel® Volume Management Device (Intel® VMD) in this Stack for socket0. First byte: Represents VMD config for PCH port (Stack 0); Second byte: Represents VMD config for IOU 0(Stack1). Each byte takes value 0x00(Disable)~0x01(Enable).

Valid range: 0x00 ~ 0x01

Default Value = {0x0, 0x0}

### 6.1.3.126 PcdVMDPchPortEnable

Configuration PCH root port: Enable VMD ownership root port (valid if PchRootPortIsAllowed for respective port). The first byte represents PCH Root Port 0, second byte is PCH Root Port 1... PCH Root Port 11 respectively for each PCH Root Port.

Valid range: 0x00~0x01

Default Value = {0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0}

### 6.1.3.127 PcdVMDPortEnableA

Enable/Disable Intel® Volume Management Device (Intel® VMD) on specific root port (VMD port A).

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.128 PcdVMDPortEnableB

Enable/Disable Intel VMD Technology on specific root port (VMD port B).

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.129 PcdVMDPortEnableC

Enable/Disable Intel VMD Technology on specific root port (VMD port C).

Valid inputs are:

0x1 - Enabled

0x0 - Disabled



Default Value = 0

### 6.1.3.130 PcdVMDPortEnabled

Enable/Disable Intel VMD Technology on specific root port (VMD port D).

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### 6.1.3.131 PcdVMDHotPlugEnable

Hot Plug Capable Enable/Disable Hot Plug for PCIe Root Ports.

The first byte represents VMD Hot plug config for PCH port. The second byte represents VMD Hot plug config for IOU 0.

Valid range: 0x00~0x01

Default Value = {0x00, 0x00}

### 6.1.3.132 PcdVMDCfgBarSz

Setup VMD Config BAR (CfgBar) size (in bits Min=0x14, Max=0x1B).

The size calculation: 0x14(20 bits)  $\rightarrow (2^{20}) / (1024 * 1024) = 1 \text{ MB}$

0x19(25 bits)  $\rightarrow (2^{25}) / (1024 * 1024) = 32 \text{ MB}$

0x1B(27bits)  $\rightarrow (2^{27}) / (1024 * 1024) = 128 \text{ MB}$

The first byte represents the VMD Config BAR size for PCH port. The second byte represents the VMD Config BAR size for IOU 0.

Valid range: 0x14 ~ 0x1B

Default Value = {0x19, 0x19}

### 6.1.3.133 PcdVMDCfgBarAttr

Set up VMD Config BAR attribute, like 64-bit or prefetchable.

The first byte represents VMD Config BAR Attribute for PCH port. The second byte represents VMD Config BAR Attribute for IOU 0.

Valid inputs are:

0x0 - 32-bit non-prefetchable

0x1 - 64-bit non-prefetchable

0x2 - 64-bit prefetchable

Default Value = {0x02, 0x02}

### 6.1.3.134 PcdVMDMemBarSz1

MemBar1 Size Setup VMD Memory BAR1 size (in bits Min=0x14, Max=0x27)

The size calculation: 0x14(20 bits)  $\rightarrow (2^{20}) / (1024 * 1024) = 1 \text{ MB}$

0x19(25 bits)  $\rightarrow (2^{25}) / (1024 * 1024) = 32 \text{ MB}$

0x1B(27bits)  $\rightarrow (2^{27}) / (1024 * 1024) = 128 \text{ MB}$

0x27(39bits)  $\rightarrow (2^{39}) / (1024 * 1024 * 1024) = 512 \text{ GB}$

The first byte represents VMD Memory BAR1 size for PCH port. The second byte represents VMD Memory BAR1 size for IOU 0.

Valid range: 0x14 ~ 0x27

Default Value = {0x19, 0x19}

### 6.1.3.135 PcdVMDMemBar1Attr

MemBar1 attribute Set up VMD Memory BAR1 attribute, like 64-bit or prefetchable First Byte: represents VMD Memory BAR1 attribute for PCH port. Second Byte: Represents VMD Memory BAR1 attribute for IOU 0.

Valid inputs are:

0x0 - 32-bit non-prefetchable

0x1 - 64-bit non-prefetchable

0x2 - 64-bit prefetchable

Default Value = {0x0, 0x0}

### 6.1.3.136 PcdVMDMemBarSz2

MemBar2 Size Setup VMD Memory BAR2 size (in bits Min=0x14, Max = 0x27)

The size calculation: 0x14(20 bits)  $\rightarrow (2^{20}) / (1024 * 1024) = 1 \text{ MB}$

0x16(22 bits)  $\rightarrow (2^{22}) / (1024 * 1024) = 4 \text{ MB}$

0x27(39bits)  $\rightarrow (2^{39}) / (1024 * 1024 * 1024) = 512 \text{ GB}$

The first byte represents VMD Memory BAR2 size for PCH port. The second byte represents VMD Memory BAR2 size for IOU 0.

Valid range: 0x14 ~ 0x27

Default Value = {0x14, 0x14}

### 6.1.3.137 PcdVMDMemBar2Attr

MemBar2 attribute Set up VMD Memory BAR2 attribute, like 64-bit or prefetchable.

First byte represents VMD Memory BAR2 attribute for PCH port. Second byte represents VMD Memory BAR2 attribute for IOU 0.

Valid inputs are:

0x0 - 32-bit non-prefetchable

0x1 - 64-bit non-prefetchable

0x2 - 64-bit prefetchable

Default Value = {0x01, 0x01}

### 6.1.3.138 PcdVMDDirectAssign

Enable/Disable VMD for Direct Assign if VMD is enabled.

The first byte represents VMD for Direct Assign for PCH port. The second byte represents VMD for Direct Assign for IOU 0.

Valid range: 0x00~0x01

Default Value = {0x0, 0x0}

### 6.1.3.139 PcdPowerLimit1Enable

Enable/Disable Power Limit 1.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### 6.1.3.140 PcdPowerLimit2Enable

Enable/Disable Power Limit 2.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.141 PcdTurboMode**

Enable/Disable Turbo Mode.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.142 PcdPcieGlobalAspm**

Enable/Disable PCIe ASPM on all IIO PCIe root ports.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.143 PcdPchLegacyIoLowLatency**

Enable/Disable low latency of legacy I/O. Increase power consumption for lower latency.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.144 PcdPchDmiAspm**

PCH DMI ASPM Enable/Disable L1 ASPM for Rlink

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.145 PcdDramRaplEnable**

Enable/Disable DRAM Running Average Power Limit (RAPL)

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.146 PcdCkeProgramming**

PCH DMI ASPM Enable/Disable CKE Throttling

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.147 PcdApdEnable**

PCH DMI ASPM Enable/Disable Auxiliary Power Detected (APD)

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

### **6.1.3.148 PcdPpdEnable**

PCH DMI ASPM Enable/Disable PPD

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

### **6.1.3.149PcdTccDsoTuningEn**

Time Coordinated Computing (Tcc) Tuning enable/disable.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x00

### **6.1.3.150PcdTccSoftwareSramEn**

Time Coordinated Computing (Tcc) Tuning Enabled

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x00

### **6.1.3.151PcdTccErrorLogEn**

Tcc Tuning enable/disable

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x00

### **6.1.3.152PcdTccStreamCfgBasePreMem**

Tcc BIOS Config File Base Address.

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

### **6.1.3.153PcdTccStreamCfgSizePreMem**

TCC BIOS Config File Size.

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

### 6.1.3.154 PcdTmePtr

The address of the table of BL\_TME\_INIT\_DATA.

Valid range: 0x00 ~ 0xFFFFFFFF

Default Value = 0x00000000

### 6.1.3.155 PcdTmeEnable

Enable or Disable Intel® Total Memory Encryption (Intel® TME).

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x00

### 6.1.3.156 PcdMkTmeEnable

Enable or Disable Intel® Total Memory Encryption – Multi-Key (Intel® TME-MK), Intel TME should be enabled before enabling Intel TME-MK.

CPU addressing is restricted to 46th bit by default. With Intel TME-MK normal addressing will be followed.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x00

### 6.1.3.157 PcdIioPcieMultiVcEnable

Enable or Disable IIO PCIe Multi Virtual Channels.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x00

### 6.1.3.158 PcdPcieRootPortEn

PCI Express Root Port. Enable/Disable PcieRootPort from 1 to 12.

Each bit represents a port(bit0-bit11) and last nibble is unused. For example, bit0 controls PcieRootPortPort 1, bit1 controls PcieRootPortPort 2...

Valid range: 0 ~ 0x0FFF

Default Value = 0x0FFF

### 6.1.3.159 PcdSgxEnable

Enable or Disable Intel® Software Guard Extensions (Intel® SGX)

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x1

### 6.1.3.160 PcdSgxAutoRegistrationAgent

Enable or Disable Intel® SGX auto registration

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x0

### 6.1.3.161 PcdSgxQoS

Enable or Disable Intel® SGX Quality of Service (QoS) to use LLC cache for EPC

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x0

### 6.1.3.162 PcdSgxDebugMode

Enable or Disable Intel® SGX Debug mode

Valid inputs are:

0x1 - Enabled

0x0 - Disabled



Default Value = 0x0

### **6.1.3.163 PcdSgxLeWr**

Enable or Disable Intel® SGX Flexible Launch Control

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0x0

### **6.1.3.164 PcdSgxLePubKeyHash0**

Intel® SGX flex launch control public key hash 0

Valid range: 0 ~ 0xFFFFFFFFFFFFFFFF

Default Value: 0

### **6.1.3.165 PcdSgxLePubKeyHash1**

Intel® SGX flex launch control public key hash 1

Valid range: 0 ~ 0xFFFFFFFFFFFFFFFF

Default Value: 0

### **6.1.3.166 PcdSgxLePubKeyHash2**

Intel® SGX flex launch control public key hash 2

Valid range: 0 ~ 0xFFFFFFFFFFFFFFFF

Default Value: 0

### **6.1.3.167 PcdSgxLePubKeyHash3**

Intel® SGX flex launch control public key hash 3

Valid range: 0 ~ 0xFFFFFFFFFFFFFFFF

Default Value: 0

### **6.1.3.168 PcdSpsIccClkSscSetting**

Enable/Disable spread spectrum clock

Valid range: 0:Disabled, 1:Enabled

Default Value: 0

### 6.1.3.169 PchPciePllSsc

Configure PCH PCIE PLL spread spectrum clock

Possible settings:

- Valid spread range: 0x00-0x14 (A value of 0 is SSC of 0.0%. A value of 20 is SSC of 2.0%) (Default)
- Auto: 0xFE (Set to hardware default)
- Disable: 0xFF

Valid range: 0x00 ~ 0xFF

Default Value: 0

## 6.1.4 FSP-S UPD Data Region

The UPD parameters that are part of the `SiliconInitUpd` and are consumed by the `FspSiliconInit` API and are described next.

### 6.1.4.1 PcdEnableSATA

SATA Controllers. Enable/disable SATA Controller. Byte 0,1,2 is for SATA controller 0,1,2 respectively. Byte 4 is unused.

Valid range: 0x0 ~ 0x00ffffff

Examples:

Valid Values: 0x00ff0001, 0x00010f01, 0x00ffff00

Example for invalid values:

Invalid value: 0xff000000, 0x21000101, 0x11000101

Default Value = 0x00010101

### 6.1.4.2 PcdSATAmode

SATA Mode

Byte 0, 1, and 2 is for SATA controller 0, 1, and 2, respectively. Byte 4 is unused as only 3 SATA controllers present. Each byte can have one of the following values: 0: AHCI, 1: RAID.

Valid range: 0x0 ~ 0x00ffffff

Examples:

Valid Values: 0x00ff0001, 0x00010f01, 0x00ffff00

Example for invalid values:

Invalid value: 0xff000000, 0x21000101, 0x11000101

Default Value = 0x00000000

### 6.1.4.3 PcdSATAInterruptMode

SATA Interrupt Mode

Byte 0, 1, and 2 is for SATA controller 0, 1, and 2 respectively. Byte 4 is unused as only 3 SATA controllers present. Each byte can have one of the following values: 0: Msix, 1: Msi, 2: Legacy.

Valid range: 0x0 ~ 0x00ffffff

Examples:

Valid Values: 0x00ff0001, 0x00010f01, 0x00ffff00

Example for invalid values:

Invalid value 0xff000000, 0x21000101, 0x11000101

Default Value = 0x00000000

### 6.1.4.4 PcdSATA0PortEnable

SATA port Enable for Controller 0.

Each one of 8 ports are represented by a nibble. For example: nibble 0 controls port 0, nibble 1 controls port 1 and so on. Each nibble can have one of the following values: 0: Disabled, 1: Enabled.

Valid range: 0x0 ~ 0x11111111

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Default Value = 0x11111111

### 6.1.4.5 PcdSATA0PortHotplug

SATA port Hot Plug capability for Controller 0.

Each one of 8 ports are represented by a nibble. For example: nibble 0 controls port 0, nibble 1 controls port 1 and so on. Each nibble can have one of the following values: 0: Disabled, 1: Enabled.

Valid range: 0x0 ~ 0x11111111

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Default Value = 0x00000000

#### 6.1.4.6 **PcdSATA1PortEnable**

SATA port Enable for Controller 1.

Each one of 8 ports are represented by a nibble. For example: nibble 0 controls port 0, nibble 1 controls port 1 and so on. Each nibble can have one of the following values: 0: Disabled, 1: Enabled.

Valid range: 0x0 ~ 0x11111111

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Default Value = 0x11111111

#### 6.1.4.7 **PcdSATA1PortHotplug**

SATA port Hot Plug capability for Controller 1

Each one of 8 ports are represented by a nibble. For example: nibble 0 controls port 0, nibble 1 controls port 1 and so on. Each nibble can have one of the following values: 0: Disabled, 1: Enabled.

Valid range: 0x0 ~ 0x11111111

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Default Value = 0x00000000

#### 6.1.4.8 **PcdSATA2PortEnable**

SATA port Enable for Controller 2

Each one of 8 ports are represented by a nibble. For example: nibble 0 controls port 0, nibble 1 controls port 1 and so on. Each nibble can have one of the following values: 0: Disabled, 1: Enabled.

Valid range: 0x0 ~ 0x11111111

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Default Value = 0x11111111

#### 6.1.4.9 **PcdSATA2PortHotplug**

SATA port Hot Plug capability for Controller 2

Each one of 8 ports are represented by a nibble. For example: nibble 0 controls port 0, nibble 1 controls port 1 and so on. Each nibble can have one of the following values: 0: Disabled, 1: Enabled.

Valid range: 0x0 ~ 0x11111111

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Default Value = 0x00000000

#### **6.1.4.10 PcdEmmc**

EMMC controller. Enable/Disable EMMC controller.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.11 PcdEmmcHS400Support**

EMMC HS400 Support. Enable/Disable EMMC HS400 Support.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.12 PcdPcieRootPort0LinkSpeed**

PCH PCIe Root Port 0 Link Speed. Desired Link Speed level for PCIe Root Port 0

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.13 PcdPcieRootPort1LinkSpeed**

PCH PCIe Root Port 1 Link Speed. Desired Link Speed level for PCIe Root Port 1

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.14 PcdPcieRootPort2LinkSpeed**

PCH PCIe Root Port 2 Link Speed. Desired Link Speed level for PCIe Root Port 2

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.15 PcdPcieRootPort3LinkSpeed**

PCH PCIe Root Port 3 Link Speed. Desired Link Speed level for PCIe Root Port 3

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.16 PcdPcieRootPort4LinkSpeed**

PCH PCIe Root Port 4 Link Speed. Desired Link Speed level for PCIe Root Port 4

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.17 PcdPcieRootPort5LinkSpeed**

PCH PCIe Root Port 5 Link Speed. Desired Link Speed level for PCIe Root Port 5

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.18 PcdPcieRootPort6LinkSpeed**

PCH PCIe Root Port 6 Link Speed. Desired Link Speed level for PCIe Root Port 6

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.19 PcdPcieRootPort7LinkSpeed**

PCH PCIe Root Port 7 Link Speed. Desired Link Speed level for PCIe Root Port 7

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.20 PcdPcieRootPort8LinkSpeed**

PCH PCIe Root Port 8 Link Speed. Desired Link Speed level for PCIe Root Port 8

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.21 PcdPcieRootPort9LinkSpeed**

PCH PCIe Root Port 9 Link Speed. Desired Link Speed level for PCIe Root Port 9

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.22 PcdPcieRootPort10LinkSpeed**

PCH PCIe Root Port 10 Link Speed. Desired Link Speed level for PCIe Root Port 10

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.23 PcdPcieRootPort11LinkSpeed**

PCH PCIe Root Port 11 Link Speed. Desired Link Speed level for PCIe Root Port 11

Valid inputs are:

1 - "GEN1"

2 - "GEN2"

3 - "GEN3"

Default Value = 2

#### **6.1.4.24 PcdPcieRootPort0Aspm**

PCH PCIe Root Port 0 Aspm. Desired Active state power management settings for PCIe Root Port 0

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.25 PcdPcieRootPort1Aspm**

PCH PCIe Root Port 1 Aspm. Desired Active state power management settings for PCIe Root Port 1

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"



3 - "L0SL1"

Default Value = 2

#### **6.1.4.26 PcdPcieRootPort2Aspm**

PCH PCIe Root Port 2 Aspm. Desired Active state power management settings for PCIe Root Port 2

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.27 PcdPcieRootPort3Aspm**

PCH PCIe Root Port 3 Aspm. Desired Active state power management settings for PCIe Root Port 3

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.28 PcdPcieRootPort4Aspm**

PCH PCIe Root Port 4 Aspm. Desired Active state power management settings for PCIe Root Port 4

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.29 PcdPcieRootPort5Aspm**

PCH PCIe Root Port 5 Aspm. Desired Active state power management settings for PCIe Root Port 5

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.30 PcdPcieRootPort6Aspm**

PCH PCIe Root Port 6 Aspm. Desired Active state power management settings for PCIe Root Port 6

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.31 PcdPcieRootPort7Aspm**

PCH PCIe Root Port 7 Aspm. Desired Active state power management settings for PCIe Root Port 7

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.32 PcdPcieRootPort8Aspm**

PCH PCIe Root Port 8 Aspm. Desired Active state power management settings for PCIe Root Port 8

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.33 PcdPcieRootPort9Aspm**

PCH PCIe Root Port 9 Aspm. Desired Active state power management settings for PCIe Root Port 9

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### **6.1.4.34 PcdPcieRootPort10Aspm**

PCH PCIe Root Port 10 Aspm. Desired Active state power management settings for PCIe Root Port 10

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### 6.1.4.35 PcdPcieRootPort11Aspm

PCH PCIe Root Port 11 Aspm. Desired Active state power management settings for PCIe Root Port 11

Valid inputs are:

0 - "Disabled"

1 - "L0"

2 - "L1"

3 - "L0SL1"

Default Value = 2

#### 6.1.4.36 PcdPcieRootPort0ConnectionType

PCH PCIe Root Port 0 Connection Type. Set Connection Type for PCIe Root Port 0. PCIe Root Port 0 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### 6.1.4.37 PcdPcieRootPort1ConnectionType

PCH PCIe Root Port 1 Connection Type. Set Connection Type for PCIe Root Port 1. PCIe Root Port 1 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### 6.1.4.38 PcdPcieRootPort2ConnectionType

PCH PCIe Root Port 2 Connection Type. Set Connection Type for PCIe Root Port 2. PCIe Root Port 2 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### 6.1.4.39 PcdPcieRootPort3ConnectionType

PCH PCIe Root Port 3 Connection Type. Set Connection Type for PCIe Root Port 3. PCIe Root Port 3 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### 6.1.4.40 PcdPcieRootPort8ConnectionType

PCH PCIe Root Port 8 Connection Type. Set Connection Type for PCIe Root Port 8. PCIe Root Port 8 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### 6.1.4.41 PcdPcieRootPort9ConnectionType

PCH PCIe Root Port 9 Connection Type. Set Connection Type for PCIe Root Port 9. PCIe Root Port 9 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### 6.1.4.42 PcdPcieRootPort10ConnectionType

PCH PCIe Root Port 10 Connection Type. Set Connection Type for PCIe Root Port 10. PCIe Root Port 10 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### **6.1.4.43 PcdPcieRootPort11ConnectionType**

PCH PCIe Root Port 11 Connection Type. Set Connection Type for PCIe Root Port 11. PCIe Root Port 11 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### **6.1.4.44 PcdPcieRootPort0HotPlug**

PCH PCIe Root Port 0 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 0.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.45 PcdPcieRootPort1HotPlug**

PCH PCIe Root Port 1 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 1.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.46 PcdPcieRootPort2HotPlug**

PCH PCIe Root Port 2 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 2.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.47 PcdPcieRootPort3HotPlug**

PCH PCIe Root Port 3 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 3.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.48 PcdPcieRootPort8HotPlug**

PCH PCIe Root Port 8 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 8.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.49 PcdPcieRootPort9HotPlug**

PCH PCIe Root Port 9 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 9.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.50 PcdPcieRootPort10HotPlug**

PCH PCIe Root Port 10 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 10.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.51 PcdPcieRootPort11HotPlug**

PCH PCIe Root Port 11 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 11.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.52 PcdPcieRootPort4ConnectionType**

PCH PCIe Root Port 4 Connection Type. Set Connection Type for PCIe Root Port 4. PCIe Root Port 4 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### **6.1.4.53 PcdPcieRootPort5ConnectionType**

PCH PCIe Root Port 5 Connection Type. Set Connection Type for PCIe Root Port 5. PCIe Root Port 5 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### **6.1.4.54 PcdPcieRootPort6ConnectionType**

PCH PCIe Root Port 6 Connection Type. Set Connection Type for PCIe Root Port 6. PCIe Root Port 6 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1

#### **6.1.4.55 PcdPcieRootPort7ConnectionType**

PCH PCIe Root Port 7 Connection Type. Set Connection Type for PCIe Root Port 7. PCIe Root Port 7 Hotplug enable forces connection type to Slot.

Valid inputs are:

0 - "Built-In"

1 - "Slot"

Default Value = 1



#### **6.1.4.56 PcdPcieRootPort4HotPlug**

PCH PCIe Root Port 4 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 4.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.57 PcdPcieRootPort5HotPlug**

PCH PCIe Root Port 5 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 5.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.58 PcdPcieRootPort6HotPlug**

PCH PCIe Root Port 6 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 6.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.59 PcdPcieRootPort7HotPlug**

PCH PCIe Root Port 7 Hot Plug. Enable/Disable Hot Plug for PCIe Root Port 7.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### **6.1.4.60 PcdLockDownBiosWpd**

Bios WPD. Enable/Disable LockDown BIOS WPD.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled



Default Value = 0

#### **6.1.4.61 PcdLockDownBiosInterface**

Bios Interface. Enable/Disable LockDown BIOS Interface.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.62 PcdLockDownGlobalSmi**

Global Smi. Enable/Disable LockDown Global Smi.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.63 PcdLockDownBiosLock**

Bios Lock. Enable/Disable LockDown BIOS Lock.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.64 PcdSbAccessUnlock**

SbAccessUnlock. Enable/Disable P2sbConfig SbAccessUnlock.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

#### 6.1.4.65 PcdPcieRootPortVppOverride

PCH PCIe Root Port VppOverride. Each one of 12 PCH Port VppOverrides are represented by a nibble. For example, nibble 0 controls PciePort 0, nibble 1 controls PciePort 1. A nibble takes value 0(Disable)~1(Enable). The last 4 nibbles are unused. Nibble default value: 0x0.

Valid range: 0x0 ~ 0x0000111111111111

Examples:

Enable PciePort 0, disable other ports: Value = 0x0000000000000001.

Enable PciePort 1, disable other ports: Value = 0x0000000000000010.

Enable PciePort 2 and 3, disable other ports: Value = 0x0000000000001100.

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Example for invalid values:

Invalid value: 0x0011000000000000

Invalid value: 0x1111000000000000

Default Value = 0x0000000000000000

#### 6.1.4.66 PcdPcieRootPortVppPort

PCH PCIe Root Port VppPort. Each one of 12 PCH VppPorts are represented by a nibble. For example, nibble 0 controls VppPort 0, nibble 1 controls VppPort 1. A nibble takes value 0~1. The last 4 nibbles are unused. Nibble default value: 0x0.

Valid range: 0x0 ~ 0x0000111111111111

Examples:

Enable VppPort 0, disable other ports: Value = 0x0000000000000001.

Enable VppPort 1, disable other ports: Value = 0x0000000000000010.

Enable VppPort 2 and 3, disable other ports: Value = 0x0000000000001100.

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Example for invalid values:

Invalid value: 0x0011000000000000

Invalid value: 0x1111000000000000

Default Value = 0x0000000000000000

#### 6.1.4.67 PcdPcieRootPortVppAddress

PCH PCIe Root Port VppAddress. Each one of 12 PCH VppAddresses are represented by a nibble. For example, nibble 0 controls VppAddress 0, nibble 1 controls VppAddress 1. A nibble takes value 0~7. The last 4 nibbles are unused. Nibble default value: 0x7.

Valid range: 0x0 ~ 0x0000777777777777

Examples:

PCI port 0 VppAddress is 1, Value = 0x0000000000000001.

PCI port 1 VppAddress is 2 and PCI port 11 VppAddress is 6, Value = 0x0000600000000020.

Invalid values:

Values other than 0 to 7 per nibble are invalid.

Example for invalid values:

Invalid value: 0x0071000000000000

Invalid value: 0x1100000000000077

Default Value = 0x0000777777777777

#### 6.1.4.68 PcdPcieRootPortPtmEnable

PCH PCIe Root Port PTM Enable. Each one of 12 PCIe Port PTM Enable are represented by a nibble. For example, nibble 0 controls PciePort 0, nibble 1 controls PciePort 1. A nibble takes value 0~1. The last 4 nibbles are unused. Nibble default value: 0x1.

Valid range: 0x00 ~ 0x0000111111111111

Examples:

Enable PciePort 0, disable other ports: Value = 0x0000000000000001.

Enable PciePort 1, disable other ports: Value = 0x0000000000000010.

Enable PciePort 2 and 3, disable other ports: Value = 0x0000000000001100.

Invalid values:

Values other than 0 or 1 per nibble are invalid.

Example for invalid values:

Invalid value: 0x1011000000000000

Invalid value: 0x1100000000000000

Default Value = 0x0000111111111111

#### 6.1.4.69 PcdWriteProtectionEnable

PCH Flash Protection Ranges Write Enable. Write or erase is blocked by hardware. Each byte represents a WriteProtectionEnable for respective Ranges. Total Protected ranges = 5.

Valid range: 0x00 ~ 0xFFFFFFFF

Default Value = 0x0101010101

#### 6.1.4.70 PcdReadProtectionEnable

PCH Flash Protection Ranges Read Enable. Read is blocked by hardware. Each byte represents a ReadProtectionEnable for respective Ranges. Total Protected ranges = 5.

Valid range: 0x00 ~ 0xFFFFFFFF

Default Value = 0x00, 0x00, 0x00, 0x00, 0x00

#### 6.1.4.71 PcdProtectedRangeLimit

PCH Protect Range Limit. Left shifted address by 12 bits with address bits [11:0] is assumed to be FFFh for limit comparison. Each two bytes represents a ProtectedRangeLimit for respective Ranges. Total Protected ranges = 5.

Valid range: 0x00 ~ 0xFFFFFFFFFFFFFFFF

Default Value = 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00

#### 6.1.4.72 PcdProtectedRangeBase

PCH Protect Range Base. Left shifted address by 12 bits with address bits [11:0] is assumed to be 0. Left shifted address by 12 bits with address bits [11:0] is assumed to be 0.

Valid range: 0x00 ~ 0xFFFFFFFFFFFFFFFF

Default Value = 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00

#### 6.1.4.73 PcdDevIntConfigPtr

Address of PCH\_DEVICE\_INTERRUPT\_CONFIG table. The address of the table of PCH\_DEVICE\_INTERRUPT\_CONFIG.

Valid range: 0x00 ~ 0xFFFFFFFFFFFFFFFF

Default Value = 0x00000000000000000000

#### 6.1.4.74 PcdNumOfDevIntConfig

Number of DevIntConfig Entry. Number of Device Interrupt Configuration Entry. If this is not zero, the DevIntConfigPtr must not be NULL.

Valid range: 0x00 ~ 0x40

Default Value = 0x00

#### 6.1.4.75 PcdIntConfigPxRcConfig

Interrupt config PxRcConfig. PxRcConfig can be configured here. First byte is for PIRQA, second byte is for PIRQB, and so on.

Valid range: 0x00 ~ 0xFFFFFFFFFFFFFFFF

Default Value = 0x0B0B0B0B0B0B0A0B

#### 6.1.4.76 PcdIntConfigGpioIrqRoute

Interrupt config GpioIrqRoute. GpioIrqRoute can be configured here. Valid value should be set.

Valid range: 0x00 ~ 0xFF

Default Value = 0x15

#### 6.1.4.77 PcdIntConfigSciIrqSelect

Interrupt config SciIrqSelect. SciIrqSelect can be configured here. Valid value should be set.

Valid range: 0x00 ~ 0xFF

Default Value = 0x09

#### 6.1.4.78 PcdPcieRootPort0L1SubStates

L1 Substates settings for PCIe Root Port 0.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.79 PcdPcieRootPort1L1SubStates**

L1 Substates settings for PCIe Root Port 1.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.80 PcdPcieRootPort2L1SubStates**

L1 Substates settings for PCIe Root Port 2.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.81 PcdPcieRootPort3L1SubStates**

L1 Substates settings for PCIe Root Port 3.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.82 PcdPcieRootPort4L1SubStates**

L1 Substates settings for PCIe Root Port 4.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.83 PcdPcieRootPort5L1SubStates**

L1 Substates settings for PCIe Root Port 5.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.84 PcdPcieRootPort6L1SubStates**

L1 Substates settings for PCIe Root Port 6.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.85 PcdPcieRootPort7L1SubStates**

L1 Substates settings for PCIe Root Port 7.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3



#### **6.1.4.86 PcdPcieRootPort8L1SubStates**

L1 Substates settings for PCIe Root Port 8.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.87 PcdPcieRootPort9L1SubStates**

L1 Substates settings for PCIe Root Port 9.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.88 PcdPcieRootPort10L1SubStates**

L1 Substates settings for PCIe Root Port 10.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.89 PcdPcieRootPort11L1SubStates**

L1 Substates settings for PCIe Root Port 11.

Valid inputs are:

0: Disabled

1: L1.1

2: L1.2

3: L1.1 and L1.2

Default Value = 3

#### **6.1.4.90 PcdTccCacheCfgBase**

Tcc Cache Config File Base Address

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

#### **6.1.4.91 PcdTccCacheCfgSize**

Tcc Cache Config File Size

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

#### **6.1.4.92 PcdTccStreamCfgBase**

Tcc Stream Buffer Config File Base Address

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

#### **6.1.4.93 PcdTccStreamCfgSize**

Tcc Stream Buffer Config File Size

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

#### **6.1.4.94 PcdTccCrlBinBase**

Tcc CRL Binary File Base Address

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

#### **6.1.4.95 PcdTccCrlBinSize**

Tcc CRL Binary Config File Size

Valid range: 0x0 ~ 0xFFFFFFFF

Default Value = 0x00000000

#### **6.1.4.96 PcdPchRlinkClockGating**

Enable/Disable Rlink Clock Gating.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.97 PcdPcieClockGatingEnabled**

Enable/Disable PCI Express Clock Gating.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 1

#### **6.1.4.98 PcdPchIoApic24119Entries**

Enable/Disable IO APIC entries 24-119.

Valid inputs are:

0x1 - Enabled

0x0 - Disabled

Default Value = 0

