AMD Family 1Ah Models 00h–0Fh and Models 10h–1Fh ACPI v6.5 Porting Guide

Publication # **58088** Revision: **0.90**

Issue Date: **November 2024**

Contents

List of Figures

List of Tables

Revision History

Date	Revision	Change Description
November 2024	0.90	Updated PRMT to clarify Output Buffer descriptions.
July 2024	0.85	Updated PRMT information.
March 2024	0.75	Initial public release.

Chapter 1 Introduction

The Advanced Configuration and Power Interface (ACPI) Specification defines a common set of firmware interfaces between the OS and the hardware platform that enables configuration and power management of the computer system and its devices. ACPI was co-developed by Hewlett-Packard, Intel, Microsoft, Phoenix, and Toshiba with the initial specification published in 1996. Since October 2013, the UEFI Forum has taken the responsibility of developing and maintaining ACPI.

ACPI is a prevalent abstraction method used today for PC, Workstation, and Server platforms. Hardware manufacturers have products and processes built around ACPI descriptions and are requesting the same technology between x86 and ARM product offerings. ACPI is the key element in Operating System directed configuration and Power Management (OSPM).

The intent of this document is to provide guidance on the ACPI interfaces required to support compliant operating systems running on a platform using AMD Family 1Ah Models 00h–0Fh and Models 10h–1Fh Processors. It represents definitions AMD has developed based on published specifications and reference documents in collaboration with BIOS and Operating System partners.

1.1 Reference Documents

1. *Advanced Configuration and Power Interface Specification, Version 6.5 – August 2022*
   https://uefi.org/sites/default/files/resources/ACPI_Spec_6_5_Aug29.pdf
2. *Unified Extensible Firmware Interface (UEFI) Specification, Version 2.10 – August 2022*
   https://uefi.org/sites/default/files/resources/UEFI_Spec_2_10_Aug29.pdf
3. *IA-PC HPET (High Precision Event Timers) Specification, Revision 1.0a – October 2004*
   https://www.intel.com/content/dam/www/public/us/en/documents/technical-specifications/software-developers-hpet-spec-1-0a.pdf
4. *Intelligent Platform Management Interface Specification, Second Generation (v2.0), Revision 1.1 – October 1, 2013*
   https://www.intel.com/content/dam/www/public/us/en/documents/product-briefs/ipmi-second-gen-interface-spec-v2-rev1-1.pdf
5. *Platform Runtime Mechanism Specification, Version 1.0 – November 2020*
   https://uefi default/files/resources/Platform%20Runtime%20Mechanism%20-%20with%20legal%20notice.pdf.org/sites/

Chapter 2 Overview of ACPI Tables

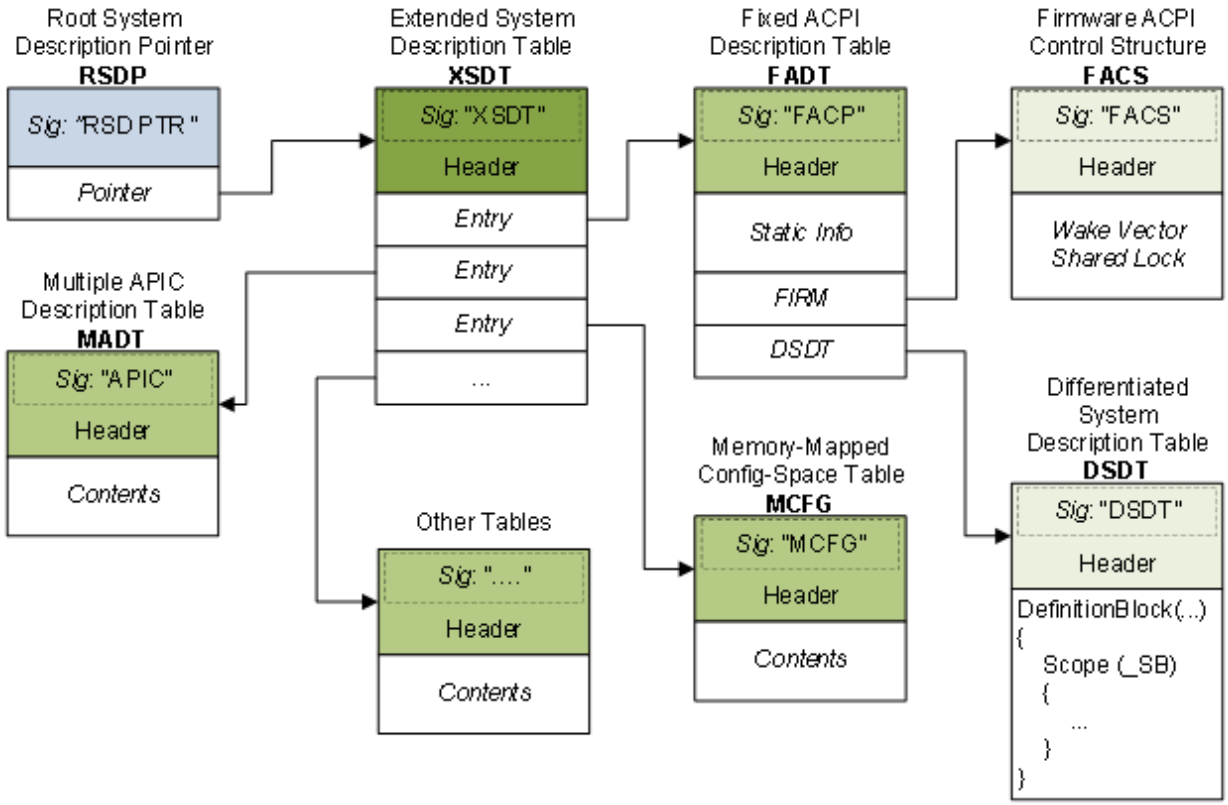

Figure 1. ACPI Table Overview

As shown in Figure 1, the OSPM interface receives a pointer to the Root System Description Pointer (RSDP) from the OS Boot loader and uses information in the RSDP to determine the addresses of all other ACPI tables in system memory.

Chapter 3 Root System Description Pointer (RSDP) Structure

Table 1. RSDP Structure Format (36 bytes)

Field	Byte Length	Byte Offset	Value
Signature	8	0	'RSD PTR'
Checksum	1	8	<checksum>
OEM ID	6	9	<'AMDINC'>
Revision	1	15	2
RsdtAddress	4	16	0
Length	4	20	36
XsdtAddress	8	24	<XSDT Address>
Extended Checksum	1	32	<ext checksum>
Reserved	3	33	0

Chapter 4 Extended System Description Table (XSDT)

Table 2. XSDT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'XSDT'
Length	4	4	<36 + 8×n descriptors>
Revision	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Description Header[0]	8	36	<FADT Address>
Description Header[1]	8	44	<MADT Address>
Description Header[2]	8	56	<MCFG Address>
Description Header[n-1]	8	<...>	<...>

Chapter 5 Fixed ACPI Description Table (FADT)

Table 3. FADT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'FACP'
Length	4	4	276
Revision (Major Version)	1	8	6
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
FIRMWARE_CTRL	4	36	<FACS Address>
DSDT	4	40	0
Reserved	1	44	0 [1 is also allowed]
Preferred_PM_Profile	1	45	0x04 [Enterprise Server]
SCI_INT	2	46	0x09
SMI_CMD	4	48	0xB2
ACPI_ENABLE	1	52	0xA0
ACPI_DISABLE	1	53	0xA1
S4BIOS_REQ	1	54	0
PSTATE_CNT	1	55	0
PM1a_EVT_BLK	4	56	<0 or 0x800>
PM1b_EVT_BLK	4	60	0
PM1a_CNT_BLK	4	64	<0 or 0x804>
PM1b_CNT_BLK	4	68	0
PM2_CNT_BLK	4	72	0
PM_TMR_BLK	4	76	<0 or 0x808>
GPE0_BLK	4	80	<0 or 0x820>
GPE1_BLK	4	84	0
PM1_EVT_LEN	1	88	4
PM1_CNT_LEN	1	89	2
PM2_CNT_LEN	1	90	0
PM_TMR_LEN	1	91	4

Table 3. FADT Format (Continued)

Field	Byte Length	Byte Offset	Value
GPE0_BLK_LEN	1	92	8
GPE1_BLK_LEN	1	93	0
GPE1_BASE	1	94	0
CST_CNT	1	95	0
P_LVL2_LAT	2	96	<<C2 Latency>>
P_LVL3_LAT	2	98	0x3E9
FLUSH_SIZE	2	100	0x400
FLUSH_STRIDE	2	102	0x10
DUTY_OFFSET	1	104	0x01
DUTY_WIDTH	1	105	0x03
DUTY_ALRM	1	106	0x0D
MON_ALRM	1	107	0
CENTURY	1	108	0x32
IAPC_BOOT_ARCH	2	109	0
Reserved	1	111	0
FADT Flags	4	112	<<FADT Flags>>
RESET_REG	12	116	<<RESET GAS>>
RESET_VALUE	1	128	0x06
ARM_BOOT_ARCH	2	129	0
FADT Minor Version	1	131	5
X_FIRMWARE_CTRL	8	132	0
X_DSDT	8	140	<DSDT Address>
X_PM1a_EVT_BLK	12	148	<<PM1a_EVT GAS>>
X_PM1b_EVT_BLK	12	160	<<NULL GAS>>
X_PM1a_CNT_BLK	12	172	<<PM1a_CNT GAS>>
X_PM1b_CNT_BLK	12	184	<<NULL GAS>>
X_PM2_CNT_BLK	12	196	<<NULL GAS>>
X_PM_TMR_BLK	12	208	<<PM_TMR GAS>>
X_GPE0_BLK	12	220	<<GPE0 GAS>>
X_GPE1_BLK	12	232	<0, NULL GAS>
SLEEP_CONTROL_REG	12	244	<0, NULL GAS>
SLEEP_STATUS_REG	12	256	<0, NULL GAS>
Hypervisor Vendor Identity	8	268	<0>

Table 4. FADT C2 Latency (2 bytes)

Value	Description
<0x64>	Value ≤ 100 microseconds if C2-state is supported
<0x65>	Value > 100 microseconds if C2-state is not supported

Table 5. FADT Flags (4 bytes)

Field	Bit Length	Bit Offset	Value
WBINVD	1	0	1
WBINVD_FLUSH	1	1	0
PROC_C1	1	2	1
P_LVL2_UP	1	3	<0/1> [1 = C2-state supported]
PWR_BUTTON	1	4	0
SLP_BUTTON	1	5	1
FIX_RTC	1	6	0
RTC_S4	1	7	0
TMR_VAL_EXT	1	8	1
DCK_CAP	1	9	0
RESET_REG_SUP	1	10	1
SEALED_CASE	1	11	0
HEADLESS	1	12	0
CPU_SW_SLP	1	13	0
PCI_EXP_WAK	1	14	0
USE_PLATFORM_CLOCK	1	15	0
S4_RTC_STS_VALID	1	16	0
REMOTE_POWER_ON_CAPABLE	1	17	1
FORCE_APIC_CLUSTER_MODEL	1	18	0
FORCE_APIC_PHYSICAL_DESTINATION	1	19	<0/1> [0 = x2APIC supported]
HW_REDUCED_ACPI	1	20	0
LOW_POWER_S0_IDLE_CAPABLE	1	21	0
PERSISTENT_CPU_CACHES	2	22	0
Reserved	8	24	0

Table 6. FADT RESET Generic Address Structure (12 bytes)

Field	Byte Length	Byte Offset	Value
Address Space ID	1	0	1 [I/O Space]
Register Bit Width	1	1	8
Register Bit Offset	1	2	0

Table 6. FADT RESET Generic Address Structure (12 bytes) (Continued)

Field	Byte Length	Byte Offset	Value
Access Size	1	3	0 [Legacy]
Address	8	4	0xCF9

Table 7. FADT PM1a_EVT Generic Address Structure (12 bytes)

Field	Byte Length	Byte Offset	Value
Address Space ID	1	0	1 [I/O Space]
Register Bit Width	1	1	32
Register Bit Offset	1	2	0
Access Size	1	3	2 [16-bit]
Address	8	4	0x800

Table 8. FADT PM1a_CNT Generic Address Structure (12 bytes)

Field	Byte Length	Byte Offset	Value
Address Space ID	1	0	1 [I/O Space]
Register Bit Width	1	1	16
Register Bit Offset	1	2	0
Access Size	1	3	2 [16-bit]
Address	8	4	0x804

Table 9. FADT PM_TMR Generic Address Structure (12 bytes)

Field	Byte Length	Byte Offset	Value
Address Space ID	1	0	1 [I/O Space]
Register Bit Width	1	1	32
Register Bit Offset	1	2	0
Access Size	1	3	3 [32-bit]
Address	8	4	0x808

Table 10. FADT GPE0 Generic Address Structure (12 bytes)

Field	Byte Length	Byte Offset	Value
Address Space ID	1	0	1 [I/O Space]
Register Bit Width	1	1	64
Register Bit Offset	1	2	0
Access Size	1	3	1 [8-bit]
Address	8	4	0x820

Chapter 6 Firmware ACPI Control Structure (FACS)

Table 11. FACS Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'FACS'
Length	4	4	64
Hardware Signature	4	8	<boot signature>
Firmware Waking Vector	4	12	0
Global Lock	4	16	<global lock>
FACS Flags	4	20	<<FACS Flags>>
X_Firmware Waking Vector	8	24	<wake vector>
Version	1	32	2
Reserved	3	33	0
OSPM Flags	4	36	<<OSPM Flags>>
Reserved	24	40	0

Table 12. FACS Flags (4 bytes)

Field	Bit Length	Bit Offset	Value	Comment
S4BIOS_F	1	0	<0/1>	1=Supported, 0=Not Supported
63BIT_WAKE_SUPPORTED_F	1	1	<0/1>	1=Supported, 0=Not Supported
Reserved	30	2	0	

Table 13. FACS OSPM Flags (4 bytes)

Field	Bit Length	Bit Offset	Value	Comment
63BIT_WAKE_F	1	0	<0/1>	1=Supported, 0=Not Supported
Reserved	31	1	0	

Chapter 7 Serial Port Console Redirection (SPCR) Table

Table 14. SPCR Table Format

Field	Byte Length	Byte Offset	Value	Comments
Signature	4	0	'SPCR'	
Length	4	4	80	
Revision	1	8	2	
Checksum	1	9	<checksum>	
OEM ID	6	10	<'AMDINC'>	
OEM Table ID	8	16	<'AMDCRB'>	
OEM Revision	4	24	0	
Creator ID	4	28	<'AMD'>	
Creator Revision	4	32	<0>	
Interface Type	1	36	0	16550 interface
Reserved	3	37	0	Must be 0
Base Address	12	40	<<SPCR GAS>>	
Interrupt Type	1	52	0	Not supported
IRQ	1	53	0	Valid only if Interrupt Type is set
Global System Interrupt	4	54	0	Valid only if Interrupt Type is set
Baud Rate	1	58	7	7 = 115200
Parity	1	59	0	0 = No Parity
Stop Bits	1	60	1	1 = 1 Stop-bit
Flow Control	1	61	0	0 = None
Terminal Type	1	62	<0..3>	0 = VT100…3 = ANSI
Reserved	1	63	0	Must be 0
PCI Device ID	2	64	0xFFFF	Not a PCI device
PCI Vendor ID	2	66	0xFFFF	Not a PCI device
PCI Bus Number	1	68	0	Not a PCI device
PCI Device Number	1	69	0	Not a PCI device
PCI Function Number	1	70	0	Not a PCI device
PCI Flags	4	71	0	Not a PCI device
PCI Segment	1	75	0	Not a PCI device
Reserved	4	76	0	Must be 0

Table 15. SPCR Generic Address Structure (12 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Address Space ID	1	0	0	System Memory
Register Bit Width	1	1	32	32-bit
Register Bit Offset	1	2	0	
Access Size	1	3	3	Dword (32-bit)
Address	8	4	0xFEDC_9000	

Chapter 8 Memory-Mapped Configuration Space (MCFG) Table

Table 16. MCFG Table Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'MCFG'
Length	4	4	60
Revision	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Reserved	8	36	0
ConfigSpace Structure[]	16	44	<<CfgSpace>>

Table 17. MCFG Configuration Space Structure Format

Field	Byte Length	Byte Offset	Value	Comments
Base Address	8	0	<0xE000_0000>	Platform-specific
PCI Segment Group Number	2	8	0	
Start Bus Number	1	10	0	
End Bus Number	1	11	<255>	Platform-specific
Reserved	4	12	0	

Chapter 9 High Precision Event Timer (HPET) Table

Table 18. HPET Table Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'HPET'
Length	4	4	56
Revision	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Hardware Block ID	4	36	<<HPET ID>> [0x1022_8203]
Base Address	12	40	<<HPET GAS>>
HPET Number	1	52	0
Minimum Clock Ticks	2	53	0x37EE
Flags	1	55	0 [No page protection]

Table 19. HPET Hardware Block ID (4 bytes)

Field	Bit Length	Bit Offset	Value
HW Rev ID	8	0	0x01
Timer Comparators	5	8	2
Counter Size Cap	1	13	0
Reserved	1	14	0
Legacy IRQ Capable	1	15	1
PCI Vendor ID	16	16	0x1022 [AMD]

Table 20. HPET Base Address (12 bytes)

Field	Byte Length	Byte Offset	Value
Address Space ID	1	0	0 [System Memory]
Register Bit Width	1	1	64
Register Bit Offset	1	2	0
Access Size	1	3	0 [Legacy]
Address	8	4	0xFED0_0000

Chapter 10 Multiple APIC Description Table (MADT)

Table 21. MADT Format

Field	Byte Length	Byte Offset	Value	Comments
Signature	4	0	'APIC'	
Length	4	4	<44 + APIC structs>	
Revision	1	8	6	
Checksum	1	9	<checksum>	
OEM ID	6	10	<'AMDINC'>	
OEM Table ID	8	16	<'AMDCRB'>	
OEM Revision	4	24	<0>	
Creator ID	4	28	<ASL Compiler ID>	
Creator Revision	4	32	<ASL Compiler Revision>	
Controller Address	4	36	0xFEE0_0000	
APIC Flags	4	40	1	PC-AT-compatible
CPU Local APIC[0..n-1]	16	44	<<CPU Local x2APIC>>	n CPUs
Local APIC NMI	12	44 + n×16	<<Local x2APIC NMI>>	
I/O APIC[0..m-1]	12	56 + n×16	<<I/O APIC>>	
Int Source Override0	10	56 + n×16 + m×12	<<Int Source0>>	
Int Source Override1	10	66 + n×16 + m×12	<<Int Source1>>	

Table 22. MADT CPU Local x2APIC (16 bytes per CPU)

Field	Byte Length	Byte Offset	Value	Comments
Type	1	0	0x9	0x9 = Processor Local x2APIC
Length	1	1	16	
Reserved	2	2	0	
x2APIC ID	4	4	<APIC_ID>	Unique per CPU
Flags	4	8	1	1 = Processor enabled, 0 = Disabled
Processor UID	4	12	<0..n-1>	_UID for each CPU in DSDT [n CPUs]

Note: *The OS enumerates logical processors (CPU0, CPU1, etc.) by the order of Local APIC (x2APIC) entries in MADT (i.e., entry0 for CPU0, entry1 for CPU1, etc.). Therefore, as a performance optimization, the recommended order for Local APIC entries is: Thread0 for all CCDs in the system then Thread1 for all CCDs in the system.*

Table 23. MADT Local x2APIC NMI (12 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	1	0	0xA	0xA = Local x2APIC NMI
Length	1	1	12	
INT Flags	2	2	<<NMI Flags>>	
Processor UID	4	4	0xFFFF_FFFF	Signifies all processors
Local APIC LINT#	1	8	0x01	LINT# to which NMI is connected
Reserved	3	9	0	

Table 24. MADT NMI Flags (2 bytes)

Field	Bit Length	Bit Offset	Value	Comments
Polarity	2	0	1	00=Bus, 01=Active High, 10=Reserved, 11=Active Low
Trigger Mode	2	2	1	00=Bus, 01=Edge, 10=Reserved, 11=Level
Reserved	12	4	0	

Table 25. MADT I/O APIC (12 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	1	0	0x1	0x1 = I/O APIC
Length	1	1	12	
I/O APIC ID	1	2	<IOAPIC ID>	0x80
Reserved	1	3	0	
I/O APIC Address	4	4	<IOAPIC Base>	0xFEC0_0000
GSI Base	4	8	0	

Table 26. MADT Interrupt Source Override0 (10 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	1	0	0x0	0x2 = Interrupt Source Override
Length	1	1	10	
Bus	1	2	0	Constant, 0 = ISA
Source	1	3	0	Bus-relative IRQ source
Global System Interrupt	4	4	2	Legacy, cascaded with IRQ9
Flags	2	8	<<Int0 Flags>>	

Table 27. MADT Int0 Source Flags (2 bytes)

Field	Bit Length	Bit Offset	Value	Comments
Polarity	2	0	0	00=Bus, 01=Active High, 10=Reserved, 11=Active Low

Table 27. MADT Int0 Source Flags (2 bytes) (Continued)

Field	Bit Length	Bit Offset	Value	Comments
Trigger Mode	2	2	0	00=Bus, 01=Edge, 10=Reserved, 11=Level
Reserved	12	4	0	

Table 28. MADT Interrupt Source Override1 (10 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	1	0	0x0	0x2 = Interrupt Source Override
Length	1	1	10	
Bus	1	2	0	Constant, 0 = ISA
Source	1	3	9	Bus-relative IRQ source
Global System Interrupt	4	4	9	Legacy IRQ9
Flags	2	8	<<Int1 Flags>>	

Table 29. MADT Int1 Source Flags (2 bytes)

Field	Bit Length	Bit Offset	Value	Comments
Polarity	2	0	3	00=Bus, 01=Active High, 10=Reserved, 11=Active Low
Trigger Mode	2	2	3	00=Bus, 01=Edge, 10=Reserved, 11=Level
Reserved	12	4	0	

Chapter 11 System Resource Affinity Table (SRAT)

Table 30. SRAT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'SRAT'
Length	4	4	<48 + size of structs>
Revision	1	8	3
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Reserved1	4	36	1 <for backward compatibility>
Reserved	8	40	0
APIC Affinity Structs[0..n-1]	<. . .>	48	<<APIC Affinity>> [n CPUs]
Memory Affinity Structs[0..m-1]	<. . .>	<. . .>	<<Memory Affinity>> [m Ranges]

Table 31. SRAT Processor Local APIC/SAPIC Affinity Structure

Field	Byte Length	Byte Offset	Value
Type	1	0	0 [APIC Affinity]
Length	1	1	16
Proxmity Domain[7:0]	1	2	<0..d-1> [d Domains]
APIC ID	1	3	<APIC_ID> [unique per CPU]
Flags	4	4	<<APIC Flags>>
SAPIC EID	1	8	0 <not applicable>
Proximity Domain[31:8]	3	9	0
Clock Domain	4	12	0 <common TSC>

Table 32. SRAT APIC Flags (4 bytes)

Field	Bit Length	Bit Offset	Value	Comments
Enabled	1	0	<0/1>	Structure ignored if value is 0
Reserved	31	1	0	

Table 33. SRAT Memory Affinity Structure

Field	Byte Length	Byte Offset	Value
Type	1	0	1 [Memory Affinity]
Length	1	1	40
Proximity Domain	4	2	<0..d-1> [d Domains]
Reserved	2	6	0
Base Address Low	4	8	<Base address low>
Base Address High	4	12	<Base address high>
Length Low	4	16	<Length low>
Length High	4	20	<Length high>
Reserved	4	24	0
Flags	4	28	<<Memory Flags>>
Reserved	8	32	0

Table 34. SRAT Memory Flags (4 bytes)

Field	Bit Length	Bit Offset	Value	Comments
Enabled	1	0	<0/1>	Structure ignored if value is 0
Hot Pluggable	1	1	0	
No Volatile	1	2	0	
Reserved	29	3	0	

Chapter 12 System Locality Distance Information Table (SLIT)

12.1 System Configuration Examples

The following are examples of 1P and 2P NUMA configurations.

1P	N0	N1	N2	N3
N0	**10**	12	12	12
N1	12	**10**	12	12
N2	12	12	**10**	12
N3	12	12	12	**10**

Figure 2. One Socket (1P) System - Quadrants as NUMA Nodes (4 Nodes per Socket)

2P	N0 (S0)	N1 (S0)	N2 (S1)	N3 (S1)
N0 (S0)	**10**	12	20	20
N1 (S0	12	**10**	20	20
N2 (S1)	20	20	**10**	12
N3 (S1)	20	20	12	**10**

Figure 3. Two Socket (2P) System - Halves as NUMA Nodes (2 Nodes per Socket)

Table 35. SLIT Format - 1P Example (NPS4)

Field	Byte Length	Byte Offset	Value
Signature	4	0	'SLIT'
Length	4	4	60
Revision	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Number of Localities	8	36	4
Entry[0][0]	1	44	10
Entry[0][1]	1	45	12

Table 35. SLIT Format - 1P Example (NPS4) (Continued)

Field	Byte Length	Byte Offset	Value
Entry[0][2]	1	46	12
Entry[0][3]	1	47	12
Entry[1][0]	1	48	12
Entry[1][1]	1	49	10
Entry[1][2]	1	50	12
Entry[1][3]	1	51	12
Entry[2][0]	1	52	12
Entry[2][1]	1	53	12
Entry[2][2]	1	54	10
Entry[2][3]	1	55	12
Entry[3][0]	1	56	12
Entry[3][1]	1	57	12
Entry[3][2]	1	58	12
Entry[3][3]	1	59	10

Table 36. SLIT Format - 2P Example (NPS2)

Field	Byte Length	Byte Offset	Value
Signature	4	0	'SLIT'
Length	4	4	60
Revision	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Number of Localities	8	36	4
Entry[0][0]	1	44	10
Entry[0][1]	1	45	12
Entry[0][2]	1	46	20
Entry[0][3]	1	47	20
Entry[1][0]	1	48	12
Entry[1][1]	1	49	10
Entry[1][2]	1	50	20
Entry[1][3]	1	51	20

Table 36. SLIT Format - 2P Example (NPS2) (Continued)

Field	Byte Length	Byte Offset	Value
Entry[2][0]	1	52	20
Entry[2][1]	1	53	20
Entry[2][2]	1	54	10
Entry[2][3]	1	55	12
Entry[3][0]	1	56	20
Entry[3][1]	1	57	20
Entry[3][2]	1	58	12
Entry[3][3]	1	59	10

Chapter 13 Maximum System Characteristics Table (MSCT)

Table 37. MSCT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'MSCT'
Length	4	4	78
Revision (Major Version)	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Proximity Domain Offset	4	36	56
Max Proximity Domains	4	40	<d-1> [d Domains]
Mac Clock Domains	4	44	0
Max Physical Address	8	48	<Max Address in system>
Proximity Domain Info	22	56	<<Max Prox Domain>>

Table 38. MSCT Maximum Proximity Domain Info (22 bytes)

Field	Byte Length	Byte Offset	Value
Revision	1	0	1
Length	1	1	22
Domain Range Start	4	2	0
Domain Range End	4	6	<d-1> [d Domains]
Max Processor Capacity	4	10	<48 or less> [Per domain]
Max Memory Capacity	8	14	<Max bytes per Domain>

Note: *This assumes that all domains have similar max capacities. If that is not the case, then there must be structure for each domain, or for ranges of domains that have similar max capacities.*

Chapter 14 UEFI ACPI Data Table

Table 39. UEFI Table Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'UEFI'
Length	4	4	66
Revision	1	8	2 [SMM Communication Table]
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Identifier	16	36	<GUID {0xC68ED8E2, 0x9DC6, 0x4CBD, 0x9D, 0x94, 0xDB, 0x65, 0xAC, 0xC5, 0xC3, 0x32}>
Data Offset	2	52	54
SW SMI Number	4	54	0x01
Buffer Ptr Address	8	58	<Buffer Ptr> [64-bit Address]

Note: *For details, refer to the "Unified Extensible Firmware Interface (UEFI) Specification [2]: Appendix 0, Table 0-1: UEFI Table Structure" [2].*

Chapter 15 Boot Graphics Resource Table (BGRT)

Table 40. BGRT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'BGRT'
Length	4	4	56
Revision (Major Version)	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Version	2	36	1
Status	1	38	<Status> [1 = Displayed]
Image Type	1	39	<Type> [0 = Bitmap]
Image Address	8	40	<Image Address>
Image Offset X	4	48	<Offset X value>
Image Offset Y	4	52	<Offset Y value>

Chapter 16 Firmware Performance Data Table (FPDT)

Table 41. FPDT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'FPDT'
Length	4	4	36 + <Performance Records>
Revision (Major Version)	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Performance Records	2	36	<<Basic Boot Performance>>

Table 42. FPDT Basic Boot Performance Data Record (16 bytes)

Field	Byte Length	Byte Offset	Value
Record Type	2	0	0 [Pointer Record]
Record Length	1	2	16
Revision	1	3	1
Reserved	4	4	0
Boot Record Address	8	8	<Record Address>

Chapter 17 Windows SMM Security Migration Table (WSMT)

Table 43. WSMT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'WSMT'
Length	4	4	40
Revision (Major Version)	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Protection Flags	4	36	<<WSMT Flags>>

Table 44. WSMT Protection Flags (4 bytes)

Field	Bit Length	Bit Offset	Value
FIXED_COMM_BUFFERS	1	0	1
COMM_BUFFER_NESTED_PTR_PROTECTION	1	1	1
SYSTEM_RESOURCE_PROTECTION	1	2	1
Reserved	29	3	0

Chapter 18 Boot Error Record Table (BERT)

Table 45. BERT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'BERT'
Length	4	4	48
Revision (Major Version)	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Boot Error Region Length	4	36	<Region Length>
Boot Error Region Address	8	40	<Region Address>

Chapter 19 Hardware Error Source Table (HEST)

Table 46. HEST Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'HEST'
Length	4	4	<40 + n structs>
Revision (Major Version)	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Error Source Count	4	36	<4 + GHES structs>
MCE Structure	<. . .>	40	<<MCE Struct>>
CMC Structure	<. . .>	<. . .>	<<CMC Struct>>
DMC Structure	<. . .>	<. . .>	<<DMC Struct>>
Generic Structure (1 or more)	64	<. . .>	<<GHES>>

Table 47. HEST Machine Check Exception Structure (40+ bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	2	0	0	0 = MCE Structure
Source ID	2	2	<Unique Id>	
Reserved	2	4	0	
Flags	1	6	<<HEST Flags>>	
Enabled	1	7	1	
Pre-allocate Count	4	8	1	
Max Sections Per Record	4	12	1	
Global Capability	8	16	<Global Capability>	May vary
Global Control	8	24	<Global Control>	May vary
Hardware Banks	1	32	<m banks>	
Reserved	7	33	0	
MCA Banks	<m×28>	40	<<MCA Banks>>	m banks

Table 48. HEST Corrected Machine Check Structure (48+ bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	2	0	1	1 = CMC Structure
Source ID	2	2	<Unique Id>	
Reserved	2	4	0	
Flags	1	6	<<HEST Flags>>	
Enabled	1	7	1	
Pre-allocate Count	4	8	1	
Max Sections Per Record	4	12	1	
Notification Structure	28	16	<<Notify Struct>>	
Hardware Banks	1	44	<m>	
Reserved	3	45	0	
MCA Banks	<m×28>	48	<<MCA Banks>>	m banks

Table 49. HEST Deferred Machine Check Structure (48+ bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	2	0	11	11 = DMC Structure
Source ID	2	2	<Unique Id>	
Reserved	2	4	0	
Flags	1	6	<<HEST Flags>>	
Enabled	1	7	1	
Pre-allocate Count	4	8	1	
Max Sections Per Record	4	12	1	
Notification Structure	28	16	<<Notify Struct>>	
Hardware Banks	1	44	<m>	
Reserved	3	45	0	
MCA Banks	<m×28>	48	<<MCA Banks>>	m banks

Table 50. HEST Flags (1 byte)

Field	Bit Length	Bit Offset	Value
FIRMWARE_FIRST	1	0	0
Reserved	1	1	0
GHES_ASSIST	1	2	<0/1> [1 = Enabled]
Reserved	5	3	0

Table 51. HEST Machine Check Bank Structure (28 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Bank Number	1	0	<0..m-1>	
Clear Status On Init	1	1	0	0 = Clear, 1 = Do not clear
Status Format	1	2	2	2 = AMD64MCA
Reserved	1	3	0	
Control MSR	4	4	<MCA_CTL>	
Control Data	8	8	<control data>	0 = Bank#4, Else all 0xF's
Status MSR	4	16	<MCA_STATUS>	
Address MSR	4	20	<MCA_ADDR>	
Misc MSR	4	24	<MCA_MISC0>	

Notes:

1. *MCA bank MSRs are located from MSRC000_2000 to MSRC000_2FFF.*
2. *The number of machine check banks can be found at MSR0000_0179[Count].*
3. *All processors maintain the same mapping of MSR number to MCA bank number: MSRC000_2000 for MCA Bank 0, MSRC000_2010 for MCA Bank 1, etc.*
4. *The legacy MCA registers per bank are located at:*

MSR	Offset from MSR Address
MCA_CTL	+0
MCA_STATUS	+1
MCA_ADDR	+2
MCA_MISC0	+3

Table 52. HEST Generic Hardware Error Source Structure (64 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	2	0	9	9 = GHES Structure
Source ID	2	2	<Unique Id>	
Related Source ID	2	4	0xFFFF	0xFFFF = None
Flags	1	6	<<HEST Flags>>	
Enabled	1	7	1	
Pre-allocate Count	4	8	1	
Max Raw Data Length	4	12	<Raw Data Length>	May vary
Max Sections Per Record	4	16	1	
Error Status Address	12	20	<Error Block Address>	
Notification Structure	28	32	<<Notify Struct>>	
Error Status Length	4	60	<Error Block Length>	

Table 53. HEST Notify Structure (28 bytes)

Field	Byte Length	Byte Offset	Value	Comments
Type	1	0	0	<0..11>, 0 = Polled
Length	1	1	28	
Config Write Enable	2	2	0	
Poll Interval	4	4	0x1388	5000 ms
Vector	4	8	0	
Polling Threshold Value	4	12	<Threshold>	May vary
Polling Threshold Window	4	16	0	
Error Threshold Value	4	20	0	
Error Threshold Window	4	24	0	

Chapter 20 Error Injection (EINJ) Table

Table 54. EINJ Table Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'EINJ'
Length	4	4	<length>
Revision (Major Version)	1	8	2
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Injection Header Size	4	36	<header size>
Injection Flags (Reserved)	1	40	0
Reserved	3	41	0
Injection Entry Count	4	44	<entry count>
Injection Entries[n]	<. . .>	48	<entries>

Chapter 21 Platform Health Assessment Table (PHAT)

Table 55. PHAT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'PHAT'
Length	4	4	267 [36 + Firmware Version Record + Reset Reason Record(s)]
Revision	1	8	2
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
Firmware Version Data Record	40	36	<<Firmware Version Record>>
Firmware Health Data Record	191	76	<<Reset Reason Record>>

Table 56. PHAT Firmware Version Record

Field	Byte Length	Byte Offset	Value
Type	2	0	0 = Firmware Version
Length	2	2	40 [12 + 28 × Record Count]
Revision	1	4	1
Reserved	3	5	0
Record Count	4	8	1
Version Element	28	12	<<Version Element>>

Table 57. PHAT Version Element

Field	Byte Length	Byte Offset	Value
Component GUID	16	0	63083674-5786-4d19-860b-e5a67d252c3b
Version Value	8	16	1
Producer ID	4	24	'AMDI'

Table 58. PHAT Reset Reason Record

Field	Byte Length	Byte Offset	Value
Type	2	0	1 = Firmware Health

Table 58. PHAT Reset Reason Record (Continued)

Field	Byte Length	Byte Offset	Value
Length	2	2	191 [116 + Data-Specific Data]
Revision	1	4	1
Reserved	2	5	0
AmHealthy	1	7	0 = Errors found
1 = No errors found			
2 = Unknown			
3 = Advisory (additional device-specific data exposed)			
Device Signature GUID	16	8	7a014ce2-f263-4b77-b88a-e6336b782c14 ***Note:*** *This GUID is normative for this record type and must not be changed.*
Device-Specific Data Offset	4	24	116
UEFI Device Path	88	28	VenHw(7A014CE2-F263-4B77-B88A-E6336B782C14) ***Note:*** *UTF-16 string, per EFI_DEVICE_PATH_PROTOCOL definition.*
Device-Specific Data	75	116	<<Reset Reason Data>>

Table 59. PHAT Reset Reason Data

Field	Byte Length	Byte Offset	Value
Supported Sources Bitmap	1	0	[0]: Unknown source [1]: Hardware source [2]: Firmware source [3]: Software source [4]: Supervisor source [7:5]: Reserved
Reset Source Bitmap	1	1	[0]: Unknown source [1]: Hardware source [2]: Firmware source [3]: Software source [4]: Supervisor source [7:5]: Reserved ***Note:*** *Only one bit should be set.*

Table 59. PHAT Reset Reason Data (Continued)

Field	Byte Length	Byte Offset	Value
Reset Sub-Source Bitmap	1	2	Unknown source: [0]: Unknown sub-source Hardware source: [0]: Unknown Firmware source: [0]: Unknown Software source: [1]: Operating System [2]: Hypervisor Supervisor source: [0]: Unknown ***Note:*** *This field must be zero if a sub-source is not defined.*
Reset Reason	1	3	0 = Unknown 1 = Cold Boot 2 = Cold Reset 3 = Warm Reset 4 = System/Software Update 32 = Unexpected Reset 33 = Fault 34 = Timeout 35 = Thermal 36 = Power Loss 37 = Power Button
Vendor-Specific Count	2	4	3 [number of Vendor-Specific Reset Reason entries]
Vendor-Specific Entry[n]	69	6	<<Vendor-Specific Reason Data>>

Table 60. PHAT Vendor-Specific Reason Data - FCH::PM::S5_RESET_STATUS

Field	Byte Length	Byte Offset	Value
Vendor-Specific GUID	16	0	1f425831-da46-4f65-9296-3c4d44c387ab
[FCH::PM::S5_RESET_STATUS]			
Length	2	16	24 [20 + Payload]
Revision	2	18	0x100
Byte [0] = Minor (0)			
Byte [1] = Major (1)			
Payload	4	20	FCH::PM::S5_RESET_STATUS

Table 61. PHAT Vendor-Specific Reason Data - FCH::PM::BREAKEVENT

Field	Byte Length	Byte Offset	Value
Vendor-Specific GUID	16	0	5cea94aa-1274-491d-89ed-f099b91fc6d6
[FCH::PM::BREAKEVENT]			
Length	2	16	24 [20 + Payload]
Revision	2	18	0x100
Byte [0] = Minor (0)			
Byte [1] = Major (1)			
Payload	4	20	FCH::PM::BREAKEVENT

Table 62. PHAT Vendor-Specific Reason Data - FCH::PM::RTCSHADOW

Field	Byte Length	Byte Offset	Value
Vendor-Specific GUID	16	0	55280bcc-b510-4d0e-b650-95853eba8950
[FCH::PM::RTCSHADOW]			
Length	2	16	21 [20 + Payload]
Revision	2	18	0x100
Byte [0] = Minor (0)			
Byte [1] = Major (1)			
Payload	1	20	FCH::PM::RTCSHADOW

Chapter 22 Platform Runtime Mechanism Table (PRMT)

Table 63. PRMT Format

Field	Byte Length	Byte Offset	Value
Signature	4	0	'PRMT'
Length	4	4	406 [60 + PRM Module Information]
Revision	1	8	1
Checksum	1	9	<checksum>
OEM ID	6	10	<'AMDINC'>
OEM Table ID	8	16	<'AMDCRB'>
OEM Revision	4	24	<0>
Creator ID	4	28	<ASL Compiler ID>
Creator Revision	4	32	<ASL Compiler Revision>
PrmPlatformGuid	16	36	4a3d492c-e023-4ad1-9317-52920e99a6ec
PrmModuleInfoOffset	4	52	60
PrmModuleInfoCount	4	56	1
PrmModuleInfoStructure [PrmModuleInfoCount]	346	60	<<PRM Module Information>>

Table 64. PRM Module Information Structure - Address Translation

Field	Byte Length	Byte Offset	Value
StructRevision	2	0	1
StructLength	2	2	346 [38 + 44 × HandlerCount]
StructIdentifier	16	4	8dceeebb-5741-4092-884d-144ec472682d [Address Translation]
ModuleMajorRev	2	20	1
ModuleMinorRev	2	22	1
HandlerCount	2	24	7
HandlerInfoOffset	4	26	38
RuntimeMmioRanges	8	30	0 [NULL Pointer]
HandlerInfoStructure [HandlerCount]	176	38	<<PRM Handler Information>>

Table 65. PRM Handler Information Structure - Normalized to DRAM Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	1

Table 65. PRM Handler Information Structure - Normalized to DRAM Address (Continued)

Field	Byte Length	Byte Offset	Value
Length	2	2	44
Identifier	16	4	7626c6ae-f973-429c-a91c-107d7be298b0 [Normalized to DRAM Address]
HandlerPhysAddress	8	20	[System Physical Address]
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 66. PRM Handler Information Structure - DRAM to Normalized Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	1
Length	2	2	44
Identifier	16	4	0639bd1c-3e33-4055-bae7-36cceba8376e [DRAM to Normalized Address]
HandlerPhysAddress	8	20	[System Physical Address]
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 67. PRM Handler Information Structure - Normalized to System Physical Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	1
Length	2	2	44
Identifier	16	4	e7180659-a65d-451d-92cd-2b56f12beba6 [Normalized to System Physical Address]
HandlerPhysAddress	8	20	[System Physical Address]
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 68. PRM Handler Information Structure - System Physical to Normalized Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	2
Length	2	2	44
Identifier	16	4	00c77891-7fcb-4d01-94e1-72f8e4ee1af7 [System Physical to Normalized Address]
HandlerPhysAddress	8	20	[System Physical Address]

Table 68. PRM Handler Information Structure - System Physical to Normalized Address (Continued)

Field	Byte Length	Byte Offset	Value
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 69. PRM Handler Information Structure - System Physical to DRAM Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	2
Length	2	2	44
Identifier	16	4	d1c6b8f2-f9ac-4bf0-855e-dbd582ce4b20 [System Physical to DRAM Address]
HandlerPhysAddress	8	20	[System Physical Address]
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 70. PRM Handler Information Structure - DRAM to System Physical Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	1
Length	2	2	44
Identifier	16	4	69aa0a9c-e3fc-4b0d-929e-aa1bde5d9a9b [DRAM to System Physical Address]
HandlerPhysAddress	8	20	[System Physical Address]
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 71. PRM Handler Information Structure - CXL DPA to System Physical Address

Field	Byte Length	Byte Offset	Value
Revision	2	0	1
Length	2	2	44
Identifier	16	4	ee41b397-25d4-452c-ad54-48c6e3480b94 [CXL DPA to System Physical Address]
HandlerPhysAddress	8	20	[System Physical Address]
StaticDataBuffer	8	28	[System Physical Address] <<PRM Static Data Buffer>>
AcpiParameterBuffer	8	36	0 [NULL Pointer]

Table 72. PRM Static Data Buffer Structure

Field	Byte Length	Byte Offset	Value
Signature	4	0	'PRMS'
Length	4	4	[8 + Data]
Data	Varies	8	[Variable length data for Handler]

22.1 OS Invocation of PRM Handlers

22.1.1 Invocation Flow

1. The caller (e.g., OS RAS Driver) passes the following information to the ACPI Bridge Driver:
   - GUID of the PRM handler to be invoked.
   - Pointer to a PRM Parameter Buffer (allocated by the caller).
2. The ACPI Bridge Driver then:
   - Identifies the PRM Handler pointer corresponding to the GUID that was passed.
   - Converts the PRM Handler Pointer from a Physical Address to a Virtual Address.
   - Extracts the Static Data Buffer Pointer and the Runtime MMIO Ranges Pointer to create a PRM Context Buffer, which is passed to the PRM Handler.
   - Invokes the PRM handler with the following calling convention:

```
EFI_STATUS PRM_EXPORT_API (EFIAPI *PRM_HANDLER) (
   IN VOID *ParameterBuffer OPTIONAL,
   IN PRM_MODULE_CONTEXT_BUFFER *ContextBuffer OPTIONAL
   );
```

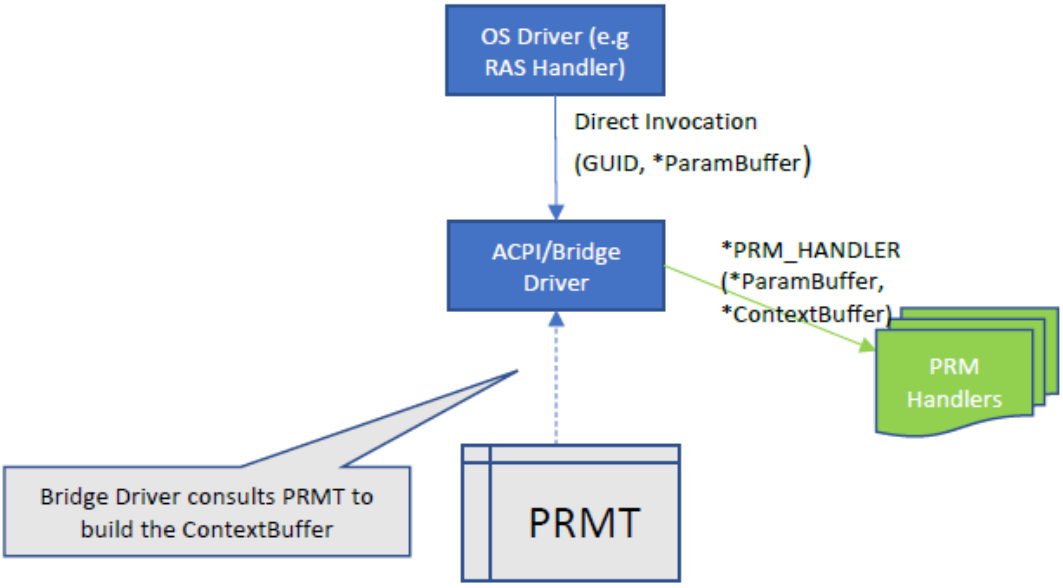

Figure 4. PRM Handler Invocation Flow

22.1.2 PRM Context Buffer

The Context Buffer is a well-defined Structure per PRM handler that describes resources available to the handler during its execution. This buffer is allocated by the OS, and the OS is responsible for converting physical addresses to virtual addresses as applicable.

Table 73. PRM Context Buffer Structure

Field	Byte Length	Byte Offset	Value
Signature	4	0	'PRMC'
Revision	2	4	1
Reserved	2	6	0
Identifier	16	8	[GUID of represented PRM Handler]
StaticDataBuffer (PRM Handler)	8	24	[Virtual Address Pointer]
RuntimeMmioRanges (PRM Module)	8	32	0 [NULL Pointer]

The Context Buffer is allocated by the ACPI Bridge Driver, and it is populated using data discovered in the PRMT ACPI table and then passed as an argument to PRM handlers. For any pointer that is NULL in the ACPI table, a NULL pointer may be passed to PRM handlers.

22.1.3 PRM Parameter Buffer

The Parameter Buffer is allocated by the caller and consumed by the invoked PRM Handler, and its internal data format is a contract between the caller and the PRM Handler.

Table 74. PRM Parameter Buffer - Normalized to DRAM Address (28 bytes)

Field	Byte Length	Byte Offset	Value
Normalized Address	8	0	e.g., MCA_ADDR_UMC for DramEccErr
Socket Number	1	8	e.g., CPUID_Fn8000001E_EAX[ExtApicId] >> CPUID_Fn80000008_ECX[ApicIdSize]
UMC Bank Instance ID	8	9	e.g., MCA_IPID_UMC
Output Buffer	11	17	[Virtual Address Pointer] Refer to Table 75

Table 75. NA→DA Output Buffer (11 bytes)

Field	Byte Length	Byte Offset
Chip Select	1	0
Bank Group	1	1
Bank Address	1	2
Row Address	4	3
Column Address	2	7
Rank Multiplier	1	9
Sub Channel	1	10

Table 76. PRM Parameter Buffer - DRAM To Normalized Address (28 bytes)

Field	Byte Length	Byte Offset	Value
Socket Number	1	0	e.g., CPUID_Fn8000001E_EAX[ExtApicId] >> CPUID_Fn80000008_ECX[ApicIdSize]
UMC Bank Instance ID	8	1	e.g., MCA_IPID_UMC
Chip Select	1	9	e.g., MCA_SYND_UMC[2:0] for EcsRowErr
Bank Group	1	10	e.g., MCA_ADDR_UMC[22:20] for EcsRowErr
Bank Address	1	11	e.g., MCA_ADDR_UMC[19:18] for EcsRowErr
Row Address	4	12	e.g., MCA_ADDR_UMC[17:0] for EcsRowErr
Column Address	2	16	e.g., 0 for EcsRowErr
Rank Multiplier	1	18	e.g., MCA_SYND_UMC[7:4] for EcsRowErr
Sub Channel	1	19	e.g., MCA_SYND_UMC[3] for EcsRowErr
Output Buffer	8	20	[Virtual Address Pointer] Refer to Table 77

Table 77. DA→NA Output Buffer (8 bytes)

Field	Byte Length	Byte Offset
Normalized Address	8	0

Table 78. PRM Parameter Buffer - Normalized to System Physical Address (25 bytes)

Field	Byte Length	Byte Offset	Value
Normalized Address	8	0	e.g., MCA_ADDR_UMC for DramEccErr
Socket Number	1	8	e.g., CPUID_Fn8000001E_EAX[ExtApicId] >> CPUID_Fn80000008_ECX[ApicIdSize]
UMC Bank Instance ID	8	9	e.g., MCA_IPID_UMC
Output Buffer	8	17	[Virtual Address Pointer] Refer to Table 79

Table 79. NA→SPA Output Buffer (8 bytes)

Field	Byte Length	Byte Offset
System Physical Address	8	0

Table 80. PRM Parameter Buffer - System Physical to Normalized Address (25 bytes)

Field	Byte Length	Byte Offset	Value
System Physical Address	8	0	e.g., From previous Translation

Table 80. PRM Parameter Buffer - System Physical to Normalized Address (25 bytes) (Continued)

Field	Byte Length	Byte Offset	Value
Output Buffer	17	8	[Virtual Address Pointer] Refer to Table 81

Table 81. SPA→NA Output Buffer (17 bytes)

Field	Byte Length	Byte Offset
Normalized Address	8	0
Socket Number	1	8
UMC Bank Instance ID	8	9

Table 82. PRM Parameter Buffer - System Physical to DRAM Address (28 bytes)

Field	Byte Length	Byte Offset	Value
System Physical Address	8	0	e.g., From previous Translation
Output Buffer	20	8	[Virtual Address Pointer] Refer to Table 83

Table 83. SPA→DA Output Buffer (20 bytes)

Field	Byte Length	Byte Offset
Chip Select	1	0
Bank Group	1	1
Bank Address	1	2
Row Address	4	3
Column Address	2	7
Rank Multiplier	1	9
Sub Channel	1	10
Socket Number	1	11
UMC Bank Instance ID	8	12

Table 84. PRM Parameter Buffer - DRAM to System Physical Address (28 bytes)

Field	Byte Length	Byte Offset	Value
Socket Number	1	0	e.g., CPUID_Fn8000001E_EAX[ExtApicId] >> CPUID_Fn80000008_ECX[ApicIdSize]
UMC Bank Instance ID	8	1	e.g., MCA_IPID_UMC
Chip Select	1	9	e.g., MCA_SYND_UMC[2:0] for EcsRowErr
Bank Group	1	10	e.g., MCA_ADDR_UMC[22:20] for EcsRowErr
Bank Address	1	11	e.g., MCA_ADDR_UMC[19:18] for EcsRowErr

Table 84. PRM Parameter Buffer - DRAM to System Physical Address (28 bytes) (Continued)

Field	Byte Length	Byte Offset	Value
Row Address	4	12	e.g., MCA_ADDR_UMC[17:0] for EcsRowErr
Column Address	2	16	e.g., 0 for EcsRowErr
Rank Multiplier	1	18	e.g., MCA_SYND_UMC[7:4] for EcsRowErr
Sub Channel	1	19	e.g., MCA_SYND_UMC[3] for EcsRowErr
Output Buffer	8	20	[Virtual Address Pointer] Refer to Table 85

Table 85. DA→SPA Output Buffer (8 bytes)

Field	Byte Length	Byte Offset
System Physical Address	8	0

Table 86. PRM Parameter Buffer - CXL DPA to System Physical Address (20 bytes)

Field	Byte Length	Byte Offset	Value
CXL Device Physical Address (DPA)	8	0	CXL DPA (e.g., from CXL Component Event Log)
CXL Endpoint SBDF	4	8	• Byte 3 - PCIe Segment • Byte 2 - Bus Number • Byte 1 - Device Number [Bits 7:3], Function Number Bits [2:0] • Byte 0 - RESERVED (MBZ)
Output Buffer	8	12	[Virtual Address Pointer] Refer to Table 87

Table 87. DPA→SPA Output Buffer (8 bytes)

Field	Byte Length	Byte Offset
System Physical Address	8	0

22.2 PRM Module Development

Refer to the PRM Software Organization section of the *Platform Runtime Mechanism Specification* [5].

Chapter 23 CXL Memory Expansion

The SoC supports memory expansion via CXL.mem capable devices.

23.1 ACPI Requirements

CXL Early Discovery Table (CEDT)

- To define one or more:
  - CXL Host Bridge Structure (CHBS)
  - CXL Fixed Memory Window Structure (CFMWS)

System Resource Affinity Table (SRAT)

- To define an extra memory-only proximity domain per socket for CXL.mem

System Locality Information Table (SLIT)

- To define Distance to the extra CXL.mem proximity domain per socket

Heterogeneous Memory Attribute Table (HMAT)

- To define Latency and Bandwidth to the extra CXL.mem proximity domain per socket

Differentiated System Description Table (DSDT, ASL Code)

- To allow binding to CXL-aware driver on legacy OSs
  - Device object with _HID ACPI0017
- To define CXL Host Bridge for each CXLmem device
  - Device object with _HID ACPI0016
- For control of CXL features
  - _OSC method under CXL Host Bridge device

For details on ACPI requirements, refer to the documents listed in Table 88.

Table 88. ACPI Requirements Reference Documents

Document Title	Link
CXL 2.0 Specification	https://www.computeexpresslink.org/_files/ugd/0c1418_764cbe0ec41a43d7969d34c81e837c2c.pdf
CXL 2.0 Errata	https://www.computeexpresslink.org/_files/ugd/0c1418_7589f56572b94dd8bc526f21064265a6.pdf
CXL 2.0 ECN: CEDT CFMWS & QTG_DSM	https://www.computeexpresslink.org/_files/ugd/0c1418_a4c32b57abeb41688fa52b9df67938b9.pdf
CXL Type-3 Memory Device Software Guide	https://cdrdv2.intel.com/v1/dl/getContent/643805
Coherent Device Attribute Table (CDAT) Specification	https://uefi.org/sites/default/files/resources/Coherent%20Device%20Attribute%20Table_1.02.pdf