AMD Secure Processor Table (ASPT) Specification, Revision 1

Publication # **58028**	Version: **1.00**
Issue Date: **February 2023**	

Contents

List of Tables

Revision History

Date	Version	Change Description
February 2023	1.00	Initial Public release.

Chapter 1 Introduction

This document describes the AMD Secure Processor Table (ASPT), which defines an alternate interface for the *Secure Encrypted Virtualization API Specification*, publication #55766.

1.1 Definitions

Table 1. Definitions

Term	Defintion
ASP	AMD Secure Processor
ASP Firmware	Firmware running on the AMD Secure Processor
SEV	Secure Encrypted Virtualization Technology

1.2 Reference Documents

Table 2. Reference Documents

PID	Title
55766	*Secure Encrypted Virtualization API Specification*
55901	*Processor Programming Reference (PPR) for AMD Family 19h Models 10h–1Fh Processors*

Chapter 2 AMD Secure Processor Table (ASPT)

This chapter describes the ASPT register fields.

Table 3. AMD Secure Processor Table (ASPT)

Field	Byte Length	Byte Offset	Description
Signature	4	0	'ASPT' = Signature for the AMD Secure Processor Table
Length	4	4	Length, in bytes, of the entire ASPT
Revision	1	8	1
Checksum	1	9	Entire table must sum to zero
OEM ID	6	10	OEM ID (e.g., 'AMDINC')
OEM Table ID	8	16	OEM Table ID (e.g., 'AMDICRB')
OEM Revision	4	24	OEM Revision
Creator ID	4	28	Vendor ID of utility that created the table
Creator Revision	4	32	Revision of utility that created the table
ASP Register Structure Count	4	36	Number of ASP Register Structure (see Table 4)
ASP Register Structure[n]	–	40	Array of ASP Register Structure (see Table 4)

Table 4 shows the ASPT register structures.

Table 4. ASPT Register Structures

Type	Description
0	See Section 2.1 ASP Global Registers.
1	See Section 2.2 SEV Mailbox Registers.
2	See Section 2.3 ACPI Mailbox Registers.

2.1 ASP Global Registers

The ASP Global Registers are common to all ASP software interfaces. For more information, refer to the *Secure Encrypted Virtualization API Specification*, publication #55766.

Table 5. ASP Global Registers

Field	Byte Length	Byte Offset	Description
Type	2	0	0 = ASP Global Registers. This structure is required in the ASPT, and only one will be produced.
Length	2	2	32
Reserved	4	4	Must be 0
Feature Register Address	8	8	System Physical Address to the ASP Feature Register
Interrupt Enable Register Address	8	16	System Physical Address to the ASP Interrupt Enable Register
Interrupt Status Register Address	8	24	System Physical Address to the ASP Interrupt Status Register

2.1.1 ASP Mailbox Interrupt IDs

Table 6. ASP Mailbox Interrupt IDs (6 Bits)

ID Value	Description
0x00	Reserved
0x01	SEV Mailbox Interrupt ID
0x02..0x3F	Reserved

2.2 SEV Mailbox Registers

The SEV Mailbox Registers are defined in the *Secure Encrypted Virtualization API Specification*, publication #55766.

Table 7. SEV Mailbox Registers

Field	Byte Length	Byte Offset	Description
Type	2	0	1 = SEV Mailbox Registers. This structure is required in the ASPT, and only one will be produced.
Length	2	2	32
Mailbox Interrupt ID	1	4	Bits[7:6] = Reserved. Must be 0. Bits[5:0] = SEV Mailbox Interrupt ID. See Section 2.3 ACPI Mailbox Registers.

Table 7. SEV Mailbox Registers

Field	Byte Length	Byte Offset	Description
Reserved	3	5	Must be 0
CmdResp Register Address	8	8	System Physical Address to the SEV CmdResp Register. See Secure Encrypted Virtualization API Specification.
CmdBufAddr_Lo Register Address	8	16	System Physical Address to the SEV CmdBufAddr_Lo Register. See Secure Encrypted Virtualization API Specification.
CmdBufAddr_Hi Register Address	8	24	System Physical Address to the SEV CmdBufAddr_Hi Register. See Secure Encrypted Virtualization API Specification.

2.3 ACPI Mailbox Registers

The ACPI Mailbox Registers provide an interface by which the OSPM can issue commands to ASP Firmware.

Table 8. ACPI Mailbox Registers

Field	Byte Length	Byte Offset	Description
Type	2	0	2 = ACPI Mailbox Registers. This structure is required in the ASPT, and only one will be produced.
Length	2	2	32
Reserved	4	4	Must be 0
CmdResp Register Address	8	8	System Physical Address to the ACPI CmdResp Register. See Section 2.3.1 ACPI CmdResp Register.
Reserved	16	16	Must be 0

2.3.1 ACPI CmdResp Register

The ACPI CmdResp Register provides a Poll-For-Response Mailbox interface by which the OSPM can issue commands to ASP Firmware.

Table 9. ACPI CmdResp Register

Bits	Description
31	0 = Command (written by OSPM) 1 = Response (written by ASP Firmware upon completion)
30:26	Status Code from ASP Firmware. Valid after Bit[31] transitions from 0 to 1. See Section 2.3.3 ACPI Status Codes.
25:16	Command Code. See Section 2.3.2 ACPI Command Codes.
15:0	Input/Output Data. See Section 2.4 ACPI Mailbox Commands.

2.3.2 ACPI Command Codes

Table 10. ACPI Command Codes (10 Bits)

Code	Command Macro	Description
0x80	DISABLE_ASP_PCIE_INTERFACE	Disable the PCIe interface
0x81	QUERY_ASP_PCIE_INTERFACE_STATUS	Query status of the PCIe interface
0x82	SET_ASP_INTERRUPT_DESCRIPTOR_PART1	Set Part1 of the interrupt descriptor
0x83	SET_ASP_INTERRUPT_DESCRIPTOR_PART2	Set Part2 of the interrupt descriptor
0x84	SET_ASP_INTERRUPT_DESCRIPTOR_PART3	Set Part3 of the interrupt descriptor
0x85	ENABLE_ASP_INTERRUPT_DELIVERY	Enable/Disable interrupt delivery

Refer to Section 2.4 ACPI Mailbox Commands for a detailed description of each command.

2.3.3 ACPI Status Codes

Table 11. ACPI Status Codes (5 Bits)

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x01	ASP_ACPI_INVALID_COMMAND	Invalid command code
0x02	ASP_ACPI_INVALID_PARAMETER	Invalid input data
0x03	ASP_ACPI_INVALID_ASP_FW_STATE	The ASP Firmware state is invalid for this command.
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any other reason

2.4 ACPI Mailbox Commands

The following commands can be issued by OSPM via the Mailbox register interface described in Section 2.3.1 ACPI CmdResp Register.

2.4.1 DISABLE_ASP_PCIE_INTERFACE

The DISABLE_ASP_PCIE_INTERFACE command disables the PCIe interface that supports the SEV API defined in the *Secure Encrypted Virtualization API Specification*, publication #55766.

After executed by ASP Firmware:

- Writes to MMIO registers of the PCIe Base Address Register (BAR) are ignored.
- Reads from MMIO registers of the PCIe Base Address Register return all 0xF's.
- Interrupts from the PCIe MSI Capability are disabled.

Input Data

None. Input Data is ignored by ASP Firmware.

Output Data

None. Output Data is not written by ASP Firmware.

Returned Status

Table 12. Returned Status from DISABLE_ASP_PCIE_INTERFACE

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any reason

Notes:

1. *Once the ASP PCIe interface is disabled, it cannot be re-enabled by OSPM.*
2. *If the PCIe interface is already disabled, the command returns ASP_ACPI_COMMAND_SUCCESS.*

2.4.2 QUERY_ASP_PCIE_INTERFACE_STATUS

The QUERY_ASP_PCIE_INTERFACE_STATUS command returns the status of the PCIe interface that supports the SEV API defined in the *Secure Encrypted Virtualization API Specification*, publication #55766.

Input Data

None. Input Data is ignored by ASP Firmware.

Output Data

Table 13. Output Data from QUERY_ASP_PCIE_INTERFACE_STATUS

Returned Data	Description
0x0000	ASP PCIe interface is disabled.
0x0001	ASP PCIe interface is enabled.

Returned Status

Table 14. Returned Status from QUERY_ASP_PCIE_INTERFACE_STATUS

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any reason

2.4.3 SET_ASP_INTERRUPT_DESCRIPTOR_PART1

The SET_ ASP_INTERRUPT_DESCRIPTOR_PART1 command configures Part1 of the Interrupt Descriptor.

Note: *The parts of the 3-part interrupt descriptor can be configured in any order. See Section 2.4.4 SET_ASP_INTERRUPT_DESCRIPTOR_PART2 and Section 2.4.5 SET_ASP_INTERRUPT_DESCRIPTOR_PART3.*

Input Data

Table 15. Input Data for SET_ASP_INTERRUPT_DESCRIPTOR_PART1

Bits	Description
15:0	Bits[15:0] of the Destination ID See Local APIC: Interrupt Command Register (ICR) in x2APIC Mode.

Output Data

None. Output Data is not written by ASP Firmware.

Returned Status

Table 16. Returned Status from SET_ASP_INTERRUPT_DESCRIPTOR_PART1

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any reason

2.4.4 SET_ASP_INTERRUPT_DESCRIPTOR_PART2

The SET_ ASP_INTERRUPT_DESCRIPTOR_PART2 command configures Part2 of the Interrupt Descriptor.

Note: *The parts of the 3-part interrupt descriptor can be configured in any order. See Section 2.4.3 SET_ASP_INTERRUPT_DESCRIPTOR_PART1 and Section 2.4.5 SET_ASP_INTERRUPT_DESCRIPTOR_PART3.*

Input Data

Table 17. Input Data for SET_ASP_INTERRUPT_DESCRIPTOR_PART2

Bits	Description
15:0	Bits[31:16] of the Destination ID See Local APIC: Interrupt Command Register (ICR) in x2APIC Mode.

Output Data

None. Output Data is not written by ASP Firmware.

Returned Status

Table 18. Returned Status from SET_ASP_INTERRUPT_DESCRIPTOR_PART2

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any reason

2.4.5 SET_ASP_INTERRUPT_DESCRIPTOR_PART3

The SET_ ASP_INTERRUPT_DESCRIPTOR_PART3 command configures Part3 of the Interrupt Descriptor.

Note: *The parts of the 3-part interrupt descriptor can be configured in any order. See Section 2.4.3 SET_ASP_INTERRUPT_DESCRIPTOR_PART1 and Section 2.4.4 SET_ASP_INTERRUPT_DESCRIPTOR_PART2.*

Input Data

Table 19. Input Data for SET_ASP_INTERRUPT_DESCRIPTOR_PART3

Bits	Description
15:10	SEV Mailbox Interrupt ID read from Section 2.2 SEV Mailbox Registers
9	Destination Mode 0 = Physical, 1 = Logical
8	Message Type 0 = Fixed, 1 = Lowest Priority
7:0	Interrupt Vector

Output Data

None. Output Data is not written by ASP Firmware.

Returned Status

Table 20. Returned Status from SET_ASP_INTERRUPT_DESCRIPTOR_PART3

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any reason

2.4.6 ENABLE_ASP_INTERRUPT_DELIVERY

The ENABLE_SEV_INTERRUPT_DELIVERY command enables or disables delivery of the Interrupt Descriptor for a specified Mailbox Interrupt ID.

Notes:

1. *This command is valid only if the ASP PCIe interface is disabled. See Section 2.4.2 QUERY_ASP_PCIE_INTERFACE_STATUS and Section 2.4.1 DISABLE_ASP_PCIE_INTERFACE.*
2. *Enabling interrupt delivery is valid only if ASP Firmware has collected at least one complete 3-part interrupt descriptor, which was saved to local store under a specified Mailbox Interrupt ID. See Section 2.4.5 SET_ASP_INTERRUPT_DESCRIPTOR_PART3.*

Input Data

Table 21. Input Data for ENABLE_ASP_INTERRUPT_DELIVERY

Bits	Description
15:10	SEV Mailbox Interrupt ID read from Section 2.2 SEV Mailbox Registers
9:1	Reserved. Must be 0.
0	0 = Disable, 1 = Enable

Output Data

None. Output Data is not written by ASP Firmware.

Returned Status

Table 22. Returned Status from ENABLE_ASP_INTERRUPT_DELIVERY

Code	Status Macro	Description
0x00	ASP_ACPI_COMMAND_SUCCESS	Successful command completion
0x02	ASP_ACPI_INVALID_PARAMETER	The specified Mailbox Interrupt ID does not match a previously saved Interrupt Descriptor, or ASP Firmware has not received a complete 3-part interrupt descriptor.
0x03	ASP_ACPI_INVALID_ASP_FW_STATE	The ASP PCIe interface is not disabled.
0x1F	ASP_ACPI_COMMAND_FAILURE	Command failed for any other reason