Core System Resources Table (CSRT)

March 13, 2017

Microsoft Corporation

Core System Resources (CSRs) are shared hardware functions such as interrupt controllers, timers, DMA controllers, non-architectural platform security requirements, and possibly others that the OS must manage. Standard CSRs, like the GIC and APIC, are described in standard ACPI tables. However, to support the diversity of CSRs on SoCs, there is a need to describe non-standard devices to the OS as well.

Microsoft is using an OS-vendor defined table to provide this information. The table will have a reserved signature ("CSRT") in the ACPI specification, and must be included in the RSDT if non-standard CSRs are used on the platform.

NOTICE

The information contained in this document is subject to change without notice. Microsoft irrevocably promises, with respect to the CSRT Specification, not to assert any of their respective Necessary Claims against any Implementers of the CSRT Specification for making, using, selling, offering for sale, importing or distributing any implementation of the CSRT Specification to the extent such implementation conforms to the CSRT Specification, and is compliant with all of the required parts of the mandatory provisions of the CSRT Specification ("Covered Implementation"), except where the Implementer asserts a claim against, or files, maintains or voluntarily participates in a lawsuit, arbitration, administrative action or other judicial or quasi-judicial proceeding against a Microsoft Covered Implementation. Here, "Implementer" means any party to this promise that implements a Covered Implementation. "Necessary Claims" are those claims of Microsoft-owned or Microsoft-controlled patents that are necessary to implement the required portions (which also include the required elements of optional portions) of the Specification that are described in detail and not merely referenced in the Specification. A patent is "controlled" by a party if (a) the patent is owned by an Affiliate of that party or (b) that party or its Affiliate, is the exclusive licensee of the patent; and that party, or its Affiliate, has the right to make this promise regarding the patent without a duty or obligation to pay a royalty or other fee to a third party. "Affiliate" means, in relation to a party, any entity controlled, directly or indirectly, by that party, any entity that controls, directly or indirectly, that party or any entity directly or indirectly under common control with that party.

This promise is not an assurance either (i) that any of Microsoft's issued patent claims covers a Covered Implementation or are enforceable or (ii) that a Covered Implementation will not infringe the patents or other intellectual property rights of any third party. No other rights except those expressly stated in this promise shall be deemed granted, waived or received by implication, exhaustion, estoppel or otherwise."

1   CSRT Table Definition

To enumerate any non-standard CSRs, the system will use a "Core System Resource Table" (CSRT). The CSRT consists of a standard ACPI Header followed by a set of variable-length Resource Groups. Each Resource Group starts with a Resource Group Header which includes Resource Group identification information and the size of the group, followed by an optional Resource Group Shared Info section containing vendor-defined information about the group, and then followed by one or more Resource Descriptors. Each resource descriptor defines a subdevice (an interrupt controller, timer, DMA controller, DMA channel, or platform security device) associated with the Resource Group. The Resource Descriptor is further broken down into a Resource Descriptor Header followed by Silicon Vendor defined information which is meaningful only to the vendor-provided software module provided for the platform.

Figure 1-1 Core System Resource Table Structure
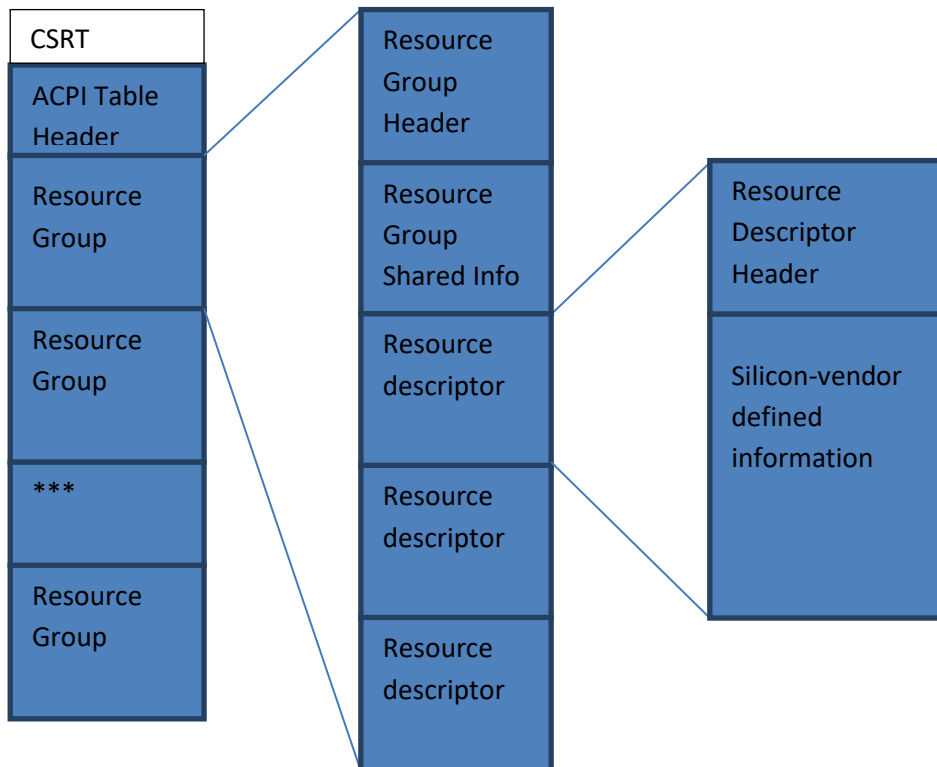
Windows will support exactly one CSRT table. If more than one CSRT table is implemented by the firmware, the behavior is undefined.

2   ACPI Table Header

ACPI Tables always start with the ACPI Table Header structure. For the Core System Resource Table, this header will have the following values:

Table 2-1. ACPI Table Header Format

Field	Byte Length	Byte Offset	Value	Comment
ACPI Header				36 bytes total
Signature	4	0	"CSRT"	Signature for the table.
Length	4	4	36 + <sum of all Resource Groups>	Length, in bytes, of the entire table
Revision	1	8	0	Revision
Checksum	1	9	<checksum>	Entire table must sum to zero.
OEMID	6	10	<firmware-specific>	OEM ID
OEM Table ID	8	16	<firmware-specific>	The table ID is the manufacturer model ID.
OEM Revision	4	24	<firmware-specific>	OEM revision for supplied OEM Table ID.
Creator ID	4	28	<firmware-specific>	Vendor ID of utility that created the table.
Creator Revision	4	32	<firmware-specific>	Revision of utility that created the table.

3   Resource Groups

A Resource Group conceptually represents a group of devices sharing the same IP, hardware resources, or registers. Each Resource Group in the CSRT begins with a Resource Group Header:

Table 3-1. Resource Group Header Format

Field	Byte Length	Byte Offset	Description
Resource Group Header			**24 bytes total**
Length	4	0	Length of this Resource Group, including the size of the Resource Group header.
Vendor ID	4	4	4-byte vendor identifier. Little-endian order: the string "ABCD" as a 32-bit value is DCBA.
Subvendor ID	4	8	4-byte subvendor identifier. Little-endian order: the string "ABCD" as a 32-bit value is DCBA. (Optional). If 0, it, and Subdevice ID, is ignored.
Device ID	2	12	2-byte (16 bit) device ID.
Subdevice ID	2	14	2-byte (16 bit) subdevice ID. Ignored if Subvendor ID is 0.
Revision	2	16	2-byte (16 bit) revision ID.
Reserved	2	18	Must be zero
SharedInfoLength	4	20	4-byte (32 bit) value which specifies the length of any additional data appended to the Resource Group Header prior to any Resource Descriptors. This additional data might include configuration information shared by all the resource descriptors within this group. If no shared data is included, this field should be set to 0.

3.1   Resource Group Shared info

The optional Resource Group Shared Info section may be used to store configuration information that is global to the associated Resource Group. This may include information like peripheral base addresses, shared interrupt lines, or any other information that the implementer deems should not be associated with any one Resource Descriptor. This section is optional; if it is not include, the SharedInfoLength field of the Resource Group Header should be set to 0.

4   Resource Descriptor

Resource Descriptor defines a single device which is part of a Resource Group. Each Resource Descriptor begins with the following header:

Table 4-1. Resource Descriptor Format

Field	Byte Length	Byte Offset	Description
Resource Descriptor Header			**12 bytes total**
Length	4	0	Length of this Resource Descriptor, including the size of the Resource Descriptor header.
Resource Type	2	4	Type for this resource. This value will have corresponding Subtype and value as shown in the Type and Subtype Table, below.
Resource Subtype	2	6	Subtype for this resource. See the Type and Subtype Table, below.
UID	4	8	32-bit resource identifier. This UID must be a number, and must be unique within the Resource Group. The UID may be used to uniquely identify a specific device within the group. 0xFFFFFFFF: Reserved
Silicon vendor defined information	N	12	Variable length information that is specific to this resource, this information is opaque to the OS and passes through to the vendor-provided module.

Table 4.2. Resource Types and Subtypes

Resource	Type	SubType	Description
Reserved	0x0000	0x0000	Do not use
Interrupt	0x0001	0x0000	Interrupt Line
		0x0001	Interrupt controller
Timer	0x0002	0x0000	Timer
DMA	0x0003	0x0000	DMA channel
		0x0001	DMA controller
Platform Security	0x0004	0x0000	Reserved
		0x0001	Platform security
		0x0002-0xFFFF	Reserved
Reserved	0x0005-0x07FF	0x0000	Reserved
Reserved	0x0800 – 0xFFFF	0xFFFF	Do not use

While it is expected that commonly Resource Groups are made up of Resource Descriptors of the same type (e.g. a block of timers), there is no requirement that this be so. In some situations, a Resource Group may need to represent a heterogeneous collection of Resource Descriptors (e.g. an interrupt controller with a built-in timer).

5   Parsing Resource Groups

The relationship between the various headers and the length fields in the CSRT is illustrated in the following diagram, which shows a CSRT containing three Resource Groups:

Figure 5-1 Core System Resource Table Size Calculation

Field	Byte Length	Value	Description
ACPI Header	36		36 bytes total "CSRT" signature
…			
Length		36 +24+L +24+M +24+N	Total size of CSRT table
…			
Resource Group Header 0	24		
Length		24+L	Size of this record.
Resource Descriptor(s) 0	L		L bytes
Resource Group Header 1	24		
Length		24+M	Size of this record.
Resource Descriptor(s) 1	M		M bytes
Resource Group Header 2	24		
Length		24+N	Size of this record.
Resource Descriptor(s) 2	N		N bytes