**# Title:**

Allow FFixedHW OpRegion

**# Status:**

Draft

**# Document:**

ACPI 6.4 specification

**# License:**

SPDX-License-Identifier: CC-BY-4.0

**# Submitters:**

* Andrei Warkentin, VMware
* Samer El-Haj-Mahmoud, ARM
* TianoCore Community (<https://www.tianocore.org>)

**# Summary of the change**

Allow FFixedHW as an OpReg

**# Benefits of the change**

Today FFixedHW is only usable inside a Register descriptor (i.e. in a ResourceTemplate). Exposing FFixedHW via an OpRegion enables advanced use of FFH on some architectures. For example, it could be used to easily proxy AML code to architecture-specific behavior (to ensure it is OS initiated)

I.e.:

Add FFixedHW to RegionSpaceKeyword

Add FFixedHW (0x0C) as an OperationRegion address space identifier

Actual behavior of course would depend on FFH bindings for a specific architecture. (e.g. look in Arm FFH spec to understand how to handle AML code performing accesses to a certain offset)

**# Impact of the change**

* New OpRegion feature that requires implementation in platform FW, OSes, ACPICA, FWTS test suite, etc..
* No impact for platforms that do not wish to use the new FFH OpRegion

**# References**

* ACPI 6.4
* Arm FFH spec: https://developer.arm.com/documentation/den0048/latest

**# Detailed description of the change [normative updates]**

* Insertions in **green**
* Removals in ~~red~~

5.5.2.4 Access to Operation Regions

5.5.2.4.1 Operation Regions

*…*

**Table 5-149 Operation Region Address Space Identifiers**

|  |  |  |
| --- | --- | --- |
| **Value** | **Name (RegionSpace Keyword)** | **Reference** |
| … | … | … |
| 0x0B | PlatformRtMechanism | ~~PlatformRtMechanism (Reserved for future use by a mechanism developed in the code-first approach).~~ Operation Region used by the Platform Runtime Mechanism Table. See [Links to ACPI-Related Documents](https://uefi.org/acpi) under the heading “Platform Runtime Mechanism Table”. |
| 0x0C | FFixedHW | See Declaring Functional Fixed Hardware (FFH) Operation Regions. |
| 0x0~~B~~D-0x7F | Reserved |  |
| 0x80 to 0xFF | OEM defined |  |

…

* + - * 1. **Declaring Functional Fixed Hardware (FFH) Operation Regions**

The syntax for declaring and using Functional Fixed Hardware (FFH) Operation Region is architecture specific. Please refer to architecture specific documentation for the definition. For ARM FFixedHW Operation Region definition, see [Links to ACPI-Related Documents](https://uefi.org/acpi) under the heading “ARM FFH Specification”.)

The use of functional fixed hardware carries with it a reliance on OS specific software that must be considered. OEMs should consult OS vendors to ensure that specific functional fixed hardware interfaces are supported by specific operating systems. The OS and the platform can handshake on support for the FFH Operation Regions using the \_OSC method as described in Platform-Wide OSPM Capabilities.

…

**6.2.11.2 Platform-Wide OSPM Capabilities**

**…**

**Table 6.13: Platform-Wide \_OSC Capabilities DWORD 2**

|  |  |  |
| --- | --- | --- |
| **Bits** | **Field Name** | **Definition** |
| … | … | … |
| 21 | Platform Runtime Mechanism | ~~Reserved for future use -~~ The OS sets this bit to indicate support for Platform Runtime Mechanism (PRM). See [Links to ACPI-Related Documents](https://uefi.org/acpi) under the heading “Platform Runtime Mechanism Table”. |
| 22 | Functional Fixed Hardware | The OS sets this bit to indicate support for the usage of Functional Fixed Hardware FFixedHW) Operation Regions. |
| 31:~~21~~23 |  | Reserved (must be 0) |

**19.2.7 ASL Parameter Keyword Terms**

AddressSpaceKeyword := RegionSpaceKeyword ~~| FFixedHW~~

**…**

RegionSpaceKeyword := SystemIO | SystemMemory | PCI\_Config | EmbeddedControl | SMBus | SystemCMOS

| PciBarTarget | IPMI | GeneralPurposeIO | GenericSerialBus | PCC | PRM | FFixedHW