# # Title:

EINJ Updates for CXL

# # Status:

Draft

# # Document:

ACPI Specification Version 6.*next*

# # License:

SPDX-License-Identifier: CC-BY-4.0

# # Submitter:

* Mahesh Natu (Intel)
* Shubhada Pugaonkar (Intel)
* Samer El-Haj Mahmoud (Arm)
* Thanu Rangarajan (Arm)
* TianoCore Community ([https://www.tianocore.org](https://www.tianocore.org/))

# # Summary of the change

This ECR is related to a proposal to update the current EINJ table to support error injection on CXL. Specifically, the ECR includes the following changes:

1. Updates to section 18.6.1 (Error Injection Table (EINJ)) to support error injection on CXL.

# # Benefits of the change

Enables support for error injection on systems that support CXL.

# # Impact of the change

Platforms firmware will have to understand and support the new format of the EINJ structure, and OS’s will also require a driver to support the new format. These are both fundamentally new code.

# # Detailed description of the change [normative updates]

Existing text

New text

~~Deleted text~~

**18.6 Error Injection**

…

**Table 18.29 Error Type Definition**

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | | | **Description** |
| 0 | | | Processor Correctable |
| 1 | | | Processor Uncorrectable non-fatal |
| 2 | | | Processor Uncorrectable fatal |
| 3 | | | Memory Correctable |
| 4 | | | Memory Uncorrectable non-fatal |
| 5 | | | Memory Uncorrectable fatal |
| 6 | | | PCI Express Correctable |
| 7 | | | PCI Express Uncorrectable non-fatal |
| 8 | | | PCI Express Uncorrectable fatal |
| 9 | | | Platform Correctable |
| 10 | | | Platform Uncorrectable non-fatal |
| 11 | | | Platform Uncorrectable fatal |
|  | 12 |  | CXL.cache Protocol Correctable |
|  | 13 |  | CXL.cache Protocol Uncorrectable non-fatal |
|  | 14 |  | CXL.cache Protocol Uncorrectable fatal |
|  | 15 |  | CXL.mem Protocol Correctable |
|  | 16 |  | CXL.mem Protocol Uncorrectable non-fatal |
|  | 17 |  | CXL.mem Protocol Uncorrectable fatal |
| ~~12~~18:30 | | | RESERVED |
| 31 | | | Vendor Defined Error Type. If this bit is set, then the Error types and related data structures are defined by the Vendor, as shown in the Vendor  Error Type Extension Structure |