Linux Plumbers Conference
Dublin, Ireland  September 12-14, 2022
CXL Type-3 device discovery, configuration in firmware and prepare ACPI tables for kernel usage

Sayanta Pattanayak
14/09/2022
Required ACPI tables for Kernel

Memory Expansion:
- A Memory NUMA node describing System locality, resource affinity, hot plug capability. SRAT ACPI structure provides these boot time description for Processor and Memory Ranges.
  - In addition, information regarding memory latency and bandwidth attributes between system localities help kernel manage resources more efficiently. HMAT ACPI structures helps in describing this information.

CXL Root Device:
- A CXL aware Kernel needs presence of Root device in ACPI namespace for enumerating CXL devices present downstream, understanding properties related to interleaving across host bridges.
  - An ACPI object with HID=ACPI0017 in firmware will allow kernel to know about the presence of CXL Root device. It also indicates presence of CEDT tables.
  - An ACPI object with HID=ACPI0016 would make kernel aware of Host bridge presence and kernel can utilize it for enumeration and configuration of the same.
Required ACPI tables for Kernel

CEDT (CXL Early Discovery Table):

- Kernel CXL driver needs information about each CXL Host bridge present in system and also the memory window where software would map HPA to CXL device memory.

- ACPI object of type CEDT – CHBS would provide the pointer to CHBCR block and enabling kernel to program any necessary HDM decoder configuration.

- ACPI object of type CEDT – CFMWS would describe the HPA memory window which kernel drivers can map to CXL Device memory based on discovery.
Development Platform for CXL

Hardware

• Fixed Virtual Platform (FVP) is the test platform that are complete simulations of an Arm system, including processor, memory and peripherals.

• Neoverse RD-N2 FVP is one of the reference design platform.

• CXL 2.0 support continues to evolve on the RD-N2 FVP.
  • Supports DVSEC, Mailbox, CDAT, DOE
  • HDM (Interleaving support implementation is ongoing).
Development Platform for CXL

Neoverse Reference Design

CMN700
  - SBSX
  - HNP
  - RNI
  - RND
  - CCG

SCP

CSS

MMU700

T2C400

CXL

CXL Root Port

CXL IO

CXL Mem

DBAM

RoS

CXL Type-3 Device

CXL Device

MMIO space

(Component Registers)

HDM Memory

CXL

PCIe

Linux Plumbers Conference | Dublin, Ireland Sept. 12-14, 2022
System Control Processor (SCP) firmware
- Interconnect configuration, CXL device discovery.
- Find out device capabilities, DOE operations.
- Configure Interconnect based on device memory range.

EDK2
- During Enumeration process discovers PCIe device with CXL and DOE capability.
- DOE operation, fetch CDAT(DSMAS, DSEMTS) tables. Find out supported Device memory range, type and associated attributes.
- Platform DXE prepares SRAT, HMAT tables based on remote CXL memory ranges discovered by CXL DXE.
- Prepare CEDT and CXL root device structures.
- [https://github.com/SayantaP-arm/edk2-platforms/tree/cxl-type-3](https://github.com/SayantaP-arm/edk2-platforms/tree/cxl-type-3)

Kernel
- As of now, utilizing the well covered CXL framework present in Kernel.
- CXL device memory is used as separate NUMA node.
- [https://github.com/SayantaP-arm/edk2/tree/cxl](https://github.com/SayantaP-arm/edk2/tree/cxl)
CXL with CEDT and Decoder Config

**CXL ACPI Hierarchy**
- Host
- ACPI 0016
- Root Port
- USP (Switch)
- DSP (Switch) → Endpoint
- CEDT ACPI 0017

**Decoder Configuration**
- Decoder count: 1
- Target count: 1
- Decoder.Base: 0x3FE000000000
- Decoder.Size: 0x200000000
- Decoder.IG: x
- Decoder.IW: 0 (1 way)
- Decoder.Type: 1
- Decoder.Target List: 1

- HDM Decoder
  - 1 Way or Non Interleave
  - Type-3 memory Device
Status & Planned Tasks

Completed

- Firmware development for discovering CXL device, mapping memory regions.

- DOE implementation in firmware for fetching CDAT structures. Prepare SRAT, HMAT tables to present CXL Type-3 memory as separate NUMA node in kernel.

- CEDT and other ACPI structures preparation and use of kernel CXL drivers for Host bridge non-interleaving HDM configuration.

Ongoing and Future

- Development of interleaving capability in FVP and software.

- Continue engagement with CXL developer community and contribute to upstream CXL firmware and kernel support.

- Memory pooling is one of the next architectural work being investigated.

- SBBR coverage.

- Continue using upstream kernel for validating firmware work on the FVP platform.
References

- Release gitlab link
  - [RD-N2 Cfg1 Platform Software guide](#)

- FVP download link
  - [Arm Ecosystem FVPs – Arm Developer](#)
THANK YOU