Socket SP3 Platform NUMA Topology for AMD Family 17h Models 30h–3Fh

Publication #	**56338** Revision: **1.00**
Issue Date:	**October 2019**

Advanced Micro Devices

Trademarks

Table of Contents

NUMA Topology in "Rome" SoC—SP3 Package 6
NUMA Nodes Per Socket (NPSx) 7
BIOS Implementation for NPSx 8
Interleaving Options 8
2-Channel Interleaving (per Quadrant)—NPS4, and Fallback for NPS2/1/0 8
4-Channel Interleaving (per Half-Socket)—NPS2 9
8-Channel Interleaving (per Socket)—NPS1 9
Socket Interleaving (2P Only)—NPS0 10
System Locality Distance Information Table (SLIT) 11
Quadrants as NUMA Nodes: 4 Nodes Per Socket (NPS4) 11
Halves as NUMA Nodes: 2 Nodes Per Socket (NPS2) 12
Sockets as NUMA Nodes: 1 Node per Socket (NPS1) 13
Last-Level Cache (L3) as NUMA Node 13
BIOS Implementation for L3AsNumaNode 13
_PXM (Proximity) 15
L3AsNUMA Disabled Configuration 15
L3AsNUMA Enabled Configuration 17

List of Tables

Table 1. NPSx Options per Model Number 7
Table 2. Interleaving Options 8

List of Figures

Figure 1. Illustration of the “Rome” CCD and CCX 6
Figure 2. Two Socket (2P) System—Quadrants as NUMA Nodes 11
Figure 3. Two Socket (2P) System—Halves as NUMA Nodes 12
Figure 4. Two Socket (2P) System—Sockets as NUMA Nodes 13
Figure 5. Single Socket (1P) System—L3AsNumaNode, NPS4 14
Figure 6. L3AsNUMA Disabled, NPS4 (4 Nodes per Socket) 15
Figure 7. L3AsNuma Disabled, NPS2 (2 Nodes per Socket) 15
Figure 8. L3AsNUMA Disabled, NPS1 (1 Node per Socket) 16
Figure 9. _PXM Assignment at the Root-Complex Level 16
Figure 10. L3AsNUMA Enabled 17
Figure 11. Sample Root Complex 17
Figure 12. _PXM Assignment at the Root-Port Level 18

Revision History

Date	Revision	Description
October 2019	1.00	Initial public release

Audience

This document is for platform firmware architects and platform BIOS developers.

References

[1] *Processor Programming Reference (PPR) for AMD Family 17h, Models 30h–3Fh Processors*, order # 55803

[2] *Advanced Configuration and Power Interface Specification* (version 6.2—May 2017)

http://www.uefi.org/sites/default/files/resources/ACPI_6_2.pdf

NUMA Topology in "Rome" SoC—SP3 Package

- Package divided into 4 quadrants, with up to 2 CCDs per quadrant.
- 2, 3, 4, 6 or 8 Core/Cache Die (CCDs), and 1 I/O die per package.
- 2 Core-Complexes (CCXs) per CCD.
- Up to 4 cores per CCX sharing an L3 cache. All CCXs configured equally.
- 2 SMT Threads (T0, T1) per core sharing an L2 cache.
- 8 Memory Channels (MC0..7) per package with up to 2 DIMMs per channel.
- Platform support for one or two sockets (1P or 2P).

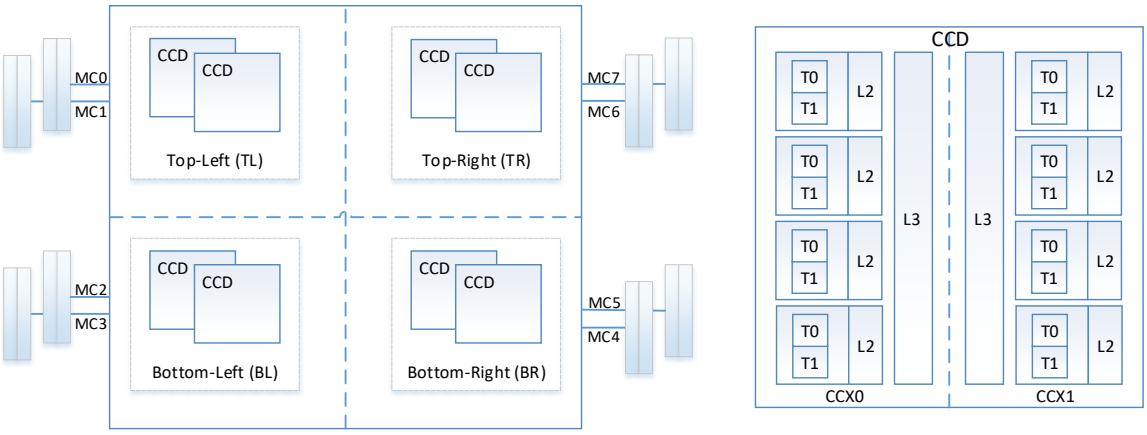


Figure 1. Illustration of the "Rome" CCD and CCX

NUMA Nodes Per Socket (NPSx)

- NPS4 - Four NUMA nodes per socket, one per Quadrant.
  - Requires symmetrical CCD configuration across Quadrants of the SoC.
  - Preferred Interleaving: 2-channel interleaving using channels from each quadrant.
- NPS2 - Two NUMA nodes per socket, one per Left/Right Half of the SoC.
  - Requires symmetrical CCD configuration across Left/Right Halves of the SoC.
  - Preferred Interleaving: 4-channel interleaving using channels from each half.
- NPS1 - One NUMA node per socket.
  - Available for any CCD configuration in the SoC.
  - Preferred Interleaving: 8-channel interleaving using all channels in the socket.
- NPS0 - One NUMA node per system.
  - Available only on a 2P system.
  - Preferred Interleaving: 16-channel interleaving using all channels in the system.

Table 1. Available NPSx Options per Model Number

Model Number	NPSx Options
7H12	4, 2, 1, 0
7742	4, 2, 1, 0
7702	4, 2, 1, 0
7702P	4, 2, 1
7662	4, 2, 1, 0
7642	2, 1, 0
7552	2, 1, 0
7542	4, 2, 1, 0
7532	2, 1, 0
7502	4, 2, 1, 0
7502P	4, 2, 1

Model Number	NPSx Options
7452	4, 2, 1, 0
7402	4, 2, 1, 0
7402P	4, 2, 1
7352	4, 2, 1, 0
7302	4, 2, 1, 0
7302P	4, 2, 1
7282	1, 0
7272	1, 0
7262	4, 2, 1, 0
7252	1, 0
7232P	1

***NOTES**:*

1. *The model number is detected based on fusing. For CCD configuration of each model number, consult Power and Thermal Data Sheet for AMD Family 17h Models 30h–3Fh Socket SP3 Processors, order # 56585.*
2. *If the CCD configuration is altered by software (e.g., BIOS Setup Option), NPS4 and NSP2 options may not be available based on the Symmetry requirements noted above.*

BIOS Implementation for NPSx

- The BIOS Setup menu shall present the applicable NPSx options based on the underlying model number. A change to the current NPSx is communicated to pre-BIOS firmware to take effect on the next boot.
- During boot, if the previously chosen NPSx option is not allowed for the model number (e.g., model number change between reboots), pre-BIOS firmware falls back to the NPS1 option, and BIOS presents an error message.
- During boot, if the preferred interleaving for the current NPSx is not possible (e.g., the memory population is inconsistent with the preferred interleaving), pre-BIOS firmware falls back to 2-Channel Interleaving per quadrant, and BIOS presents a warning message.

Interleaving Options

As shown in the following table, based on the NPSx selection, the pre-BIOS firmware chooses the corresponding preferred memory interleaving. If the memory configuration does not allow for the preferred option (e.g., a channel is not populated or one or more DIMMs on a channel does not initialize or train properly), the pre-BIOS firmware chooses the corresponding alternate memory interleaving option.

Table 2. Interleaving Options

Interleaving Options Based on NPSx		
NPSx	**Preferred**	**Alternate**
4	2-channel	None
2	4-channel	2-channel
1	8-channel	4-channel, 2-channel
0	16-channel, 2P	2-channel

***Note**: In a 2P system both sockets must be in the same interleaving mode if both sockets have memory populated.*

2-Channel Interleaving (per Quadrant)—NPS4, and Fallback for NPS2/1/0

- This interleaves two channels in each quadrant.
- Does not require the memory to be equal on both channels of a quadrant. Any non-symmetrical DIMM is stacked on top.
- Any quadrant where one of the two channels is not populated is not interleaved.

- There is no alternate, as all configurations can be mapped into this mode.

4-Channel Interleaving (per Half-Socket)—NPS2

- This interleaves the four channels on the left or right half of a socket. As an alternative option from NPS1 only, the four channels {CS 2, 3, 4, 5} may be interleaved.
- Requires all four channels to be populated with equal size memory.
- There is no requirement that the two halves have equal size memory with respect to each other.
- The system has support for one half to have no memory.

- In a 2P system:
    - There is no requirement that both sockets have the same number of halves populated.
    - There is no requirement that each of the four halves has the same amount of memory with respect to each other.
    - The system allows for one of the sockets to have no memory

8-Channel Interleaving (per Socket)—NPS1

- This interleaves eight channels in a socket.
- If only four channels per socket are populated, the system will alternate to 4-channel interleaving. It is supported for the following channel configurations: {CS 0, 1, 2, 3}, {CS 2, 3, 4, 5}, and {CS 4, 5, 6, 7}.
    - Note: While supported, {CS 0, 1, 2, 3} and {CS 4, 5, 6, 7} is recommended as a memory population only if all eight channels are populated as NPS2. The recommended memory population for four channels only is {CS 2, 3, 4, 5}.
- Requires all populated channels in a socket to have equal size memory.
- In a 2P system:
    - There is no requirement for both sockets to have equal size memory.
    - If both sockets have memory, the interleaving mode must be the same for both sockets.
    - The system allows for one of the sockets to have no memory.
- In a 1P system, creates a single NUMA node for the system, in which case SRAT and SLIT table are not required.

Socket Interleaving (2P Only)—NPS0

- This interleaves all sixteen channels (eight per socket) in a 2P system.
- Requires all channels in the system to be populated with equal size memory.
- Creates a single NUMA node for the system, in which case SRAT and SLIT table are not required.

System Locality Distance Information Table (SLIT)

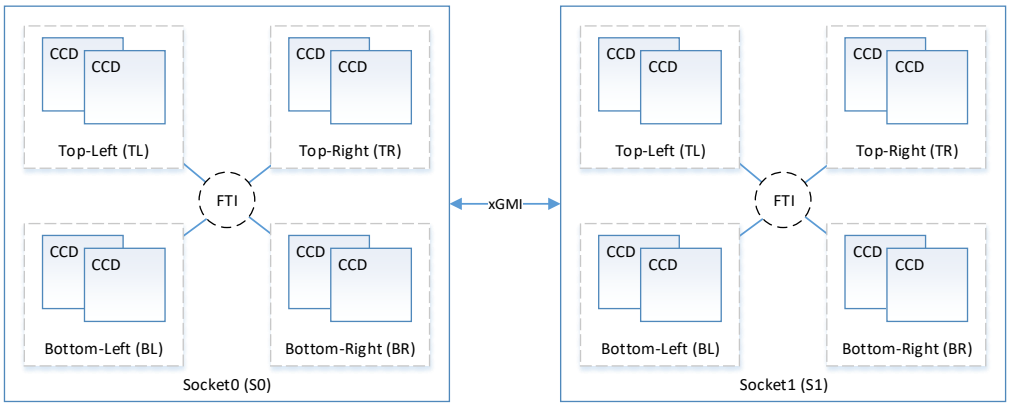
The Linux OS defines a RECLAIM_DISTANCE value (30) to represent the distance between nodes from which it is deemed too costly to allocate. The OS prefers to reclaim from a local node before falling back to allocating on a remote node with such a distance.

Hence, for applications where pinning execution to nodes within a socket is desirable, the recommended Far-Hop Distance value is 32 (larger than 30). Moreover, the BIOS Setup should present a **PinNodesToSocket** (Boolean) option to set the Far-Hop Distance value (20 or 32).

Quadrants as NUMA Nodes: 4 Nodes Per Socket (NPS4)

Each quadrant is configured as a NUMA node: four NUMA nodes per socket.

- Local node is 10
- Near-Hop Distance (on same socket): 1-hop FTI (+2) = 12
- Far-Hop Distance (on other socket): 1-hop xGMI (+10) = PinNodesToSocket ? 32 : 20



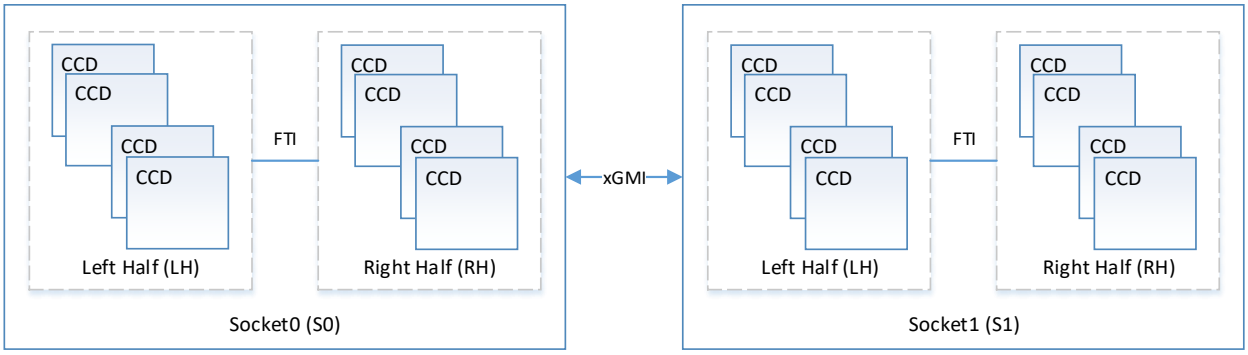
SP3-2P	S0, BR	S0, TR	S0, TL	S0, BL	S1, BR	S1, TR	S1, TL	S1, BL
S0, BR	**10**	12	12	12	**20/32**	**20/32**	**20/32**	**20/32**
S0, TR	12	**10**	12	12	**20/32**	**20/32**	**20/32**	**20/32**
S0, TL	12	12	**10**	12	**20/32**	**20/32**	**20/32**	**20/32**
S0, BL	12	12	12	**10**	**20/32**	**20/32**	**20/32**	**20/32**
S1, BR	**20/32**	**20/32**	**20/32**	**20/32**	**10**	12	12	12
S1, TR	**20/32**	**20/32**	**20/32**	**20/32**	12	**10**	12	12
S1, TL	**20/32**	**20/32**	**20/32**	**20/32**	12	12	**10**	12
S1, BL	**20/32**	**20/32**	**20/32**	**20/32**	12	12	12	**10**

Figure 2. Two Socket (2P) System—Quadrants as NUMA Nodes

Halves as NUMA Nodes: 2 Nodes Per Socket (NPS2)

Each half is configured as a NUMA node: 2 NUMA nodes per socket.

- The local node is 10
- Near-Hop Distance (on same socket): 1-hop FTI (+2) = 12
- Far-Hop Distance (on other socket): 1-hop xGMI (+10) = PinNodesToSocket ? 32 : 20

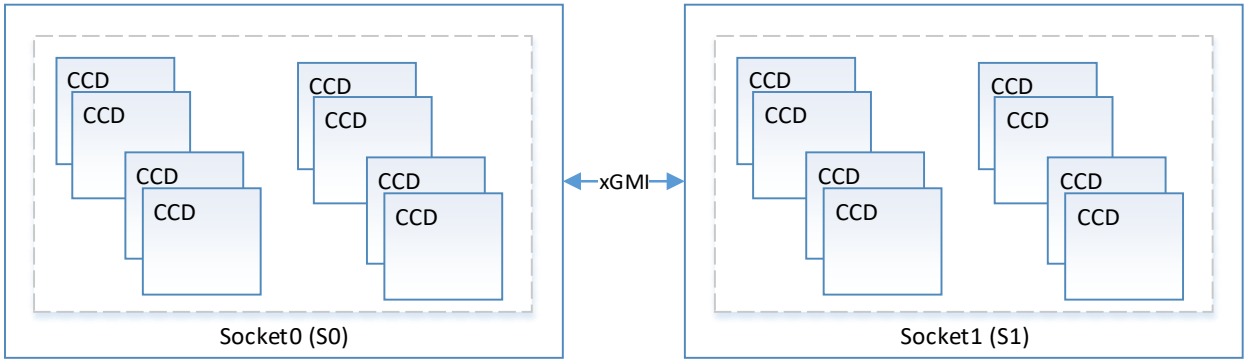


SP3-2P	S0, RH	S0, LH	S1, RH	S1, LH
S0, RH	**10**	12	**20/32**	**20/32**
S0, LH	12	**10**	**20/32**	**20/32**
S0, RH	**20/32**	**20/32**	**10**	12
S1, LH	**20/32**	**20/32**	12	**10**

Figure 3. Two Socket (2P) System—Halves as NUMA Nodes

Sockets as NUMA Nodes: 1 Node per Socket (NPS1)

- The local node is 10
- Far-Hop Distance (on other socket): 1-hop xGMI (+10) = PinNodesToSocket ? 32 : 20



SP3-2P	S0	S1
S0	10	20/32
S1	20/32	10

Figure 4. Two Socket (2P) System—Sockets as NUMA Nodes

Last-Level Cache (L3) as NUMA Node

In certain Datacenter applications where workloads are managed by a remote Job scheduler, it is desirable to pin execution to a single NUMA node, and preferably to share a single Last-Level cache within that node. Hence BIOS Setup should support an **L3AsNumaNode** (Boolean) option to create a NUMA node for each CCX (L3 Cache) in the system.

BIOS Implementation for L3AsNumaNode

- CPU and memory resources are partitioned based on the chosen **NPSx** option:
  - **NPS4**—4 partitions per socket, with each partition consisting of an SoC quadrant.
  - **NPS2**—2 partitions per socket, with each partition consisting of an SoC half.
  - **NPS1**—1 partition per socket.
  - **NPS0**—1 partition per system (2P only).
- The number of NUMA nodes is derived from the number of CCXs in the system, because each CCX contains an LLC/L3 cache.
- The CPUs assigned to each NUMA node are simply the CPUs contained in each CCX.
- The memory assigned to each NUMA node is divided equally from the total amount of memory in the partition to which each CCX belongs.

For example, in an **NPS4** configuration the total amount of memory in each quadrant is divided equally between the (four at most) CCXs in the quadrant.

- Memory interleaving for each partition is derived based on the chosen **NPSx** option, as described in the *Interleaving Options* section.
- SLIT distances are derived as follows:
  - The local node is 10.
  - Nodes within a given partition are 11.
  - Nodes in other partitions within the same socket are 12.
  - Nodes in the other socket are 20 or 32, per **PinNodesToSocket** in the *System Locality Distance Information Table (SLIT)* section.

For example, a fully populated 1P system configured in **NPS4** would consist of: four partitions, with four CCXs per partition, for a total of 16 NUMA nodes.

LLC-1P	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	**10**	11	11	11	12	12	12	12	12	12	12	12	12	12	12	12
1	11	**10**	11	11	12	12	12	12	12	12	12	12	12	12	12	12
2	11	11	**10**	11	12	12	12	12	12	12	12	12	12	12	12	12
3	11	11	11	**10**	12	12	12	12	12	12	12	12	12	12	12	12
4	12	12	12	12	**10**	11	11	11	12	12	12	12	12	12	12	12
5	12	12	12	12	11	**10**	11	11	12	12	12	12	12	12	12	12
6	12	12	12	12	11	11	**10**	10	12	12	12	12	12	12	12	12
7	12	12	12	12	11	11	11	**10**	12	12	12	12	12	12	12	12
8	12	12	12	12	12	12	12	12	**10**	11	11	11	12	12	12	12
9	12	12	12	12	12	12	12	12	11	**10**	11	11	12	12	12	12
10	12	12	12	12	12	12	12	12	11	11	**10**	11	12	12	12	12
11	12	12	12	12	12	12	12	12	11	11	11	**10**	12	12	12	12
12	12	12	12	12	12	12	12	12	12	12	12	12	**10**	11	11	11
13	12	12	12	12	12	12	12	12	12	12	12	12	11	**10**	11	11
14	12	12	12	12	12	12	12	12	12	12	12	12	11	11	**10**	11
15	12	12	12	12	12	12	12	12	12	12	12	12	11	11	11	**10**

Figure 5. Single Socket (1P) System—L3AsNumaNode, NPS4

_PXM (Proximity)

This optional object is used to describe proximity domain associations within a machine in the ACPI namespace (DSDT). _PXM evaluates to an integer that identifies a device as belonging to a Proximity Domain defined in the System Resource Affinity Table (SRAT).

In the Rome SoC there are four PCIe root-complexes per socket (PCIe RC0..3), each of which has physical proximity to a CCD complex on each quadrant.

L3AsNUMA Disabled Configuration

In the L3AsNUMA Disabled configuration, each root-complex is associated with at most a single NUMA node, as shown in Figure 6 through Figure 8.

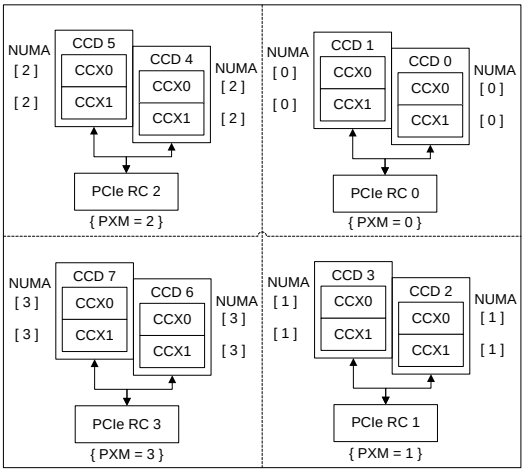


Figure 6. L3AsNUMA Disabled, NPS4 (4 Nodes per Socket)

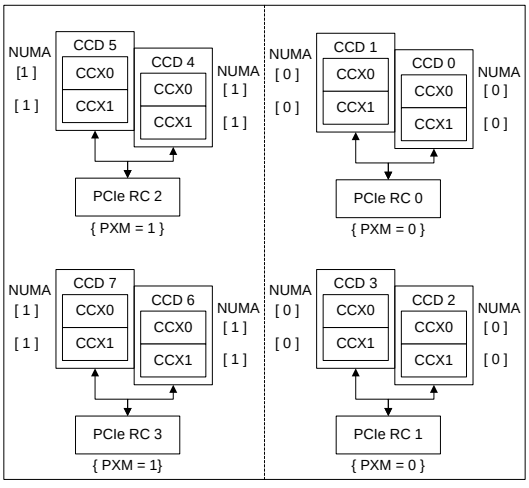


Figure 7. L3AsNuma Disabled, NPS2 (2 Nodes per Socket)

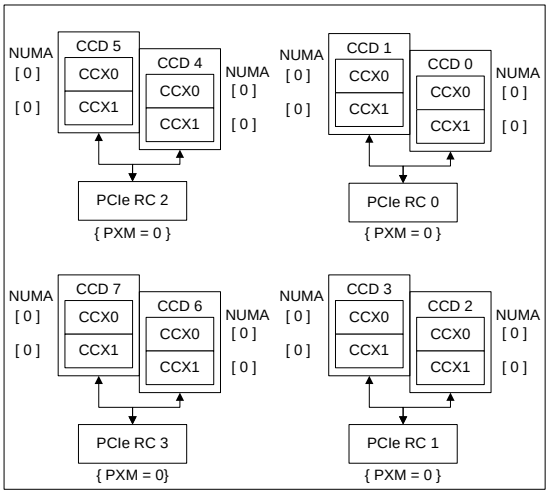


Figure 8. L3AsNUMA Disabled, NPS1 (1 Node per Socket)

Hence, _PXM assignments can be made at root-complex level in the ACPI namespace (DSDT), as shown in Figure 9.

```
Scope (\_SB) {
  Device (PCI0) { // Root PCI Bus (Host-Bridge)
    Name (_HID, EISAID("PNP0A08"))
    Name (_CID, EISAID("PNP0A03"))
    Name (_BBN, 0)
    Method (_CRS,0) {
      // Return current resources for host bridge 0
    }
    Name (_PRT, Package() {
     // Package with PCI IRQ routing table information
    })
    Method (_PXM, 0, NotSerialized) {
         Return (PXM0)
    }
  }
  // ...
}
```

Figure 9. _PXM Assignment at the Root-Complex Level

L3AsNUMA Enabled Configuration

As shown in Figure 10, in the L3AsNUMA Enabled configuration, each root-complex may be associated with multiple NUMA nodes, because each CCX represents a NUMA node.

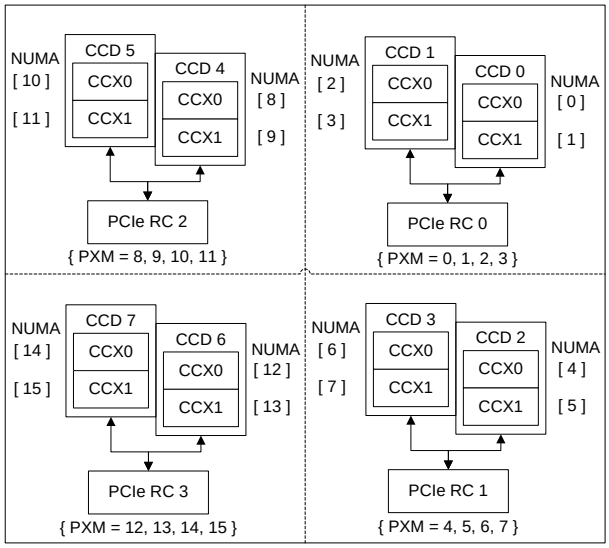


Figure 10. L3AsNUMA Enabled

As shown in Figure 11, each root complex may have multiple root ports. Hence _PXM assignments can be made at root-port (P2P Bridge) level in the ACPI namespace (DSDT), as shown in Figure 12.

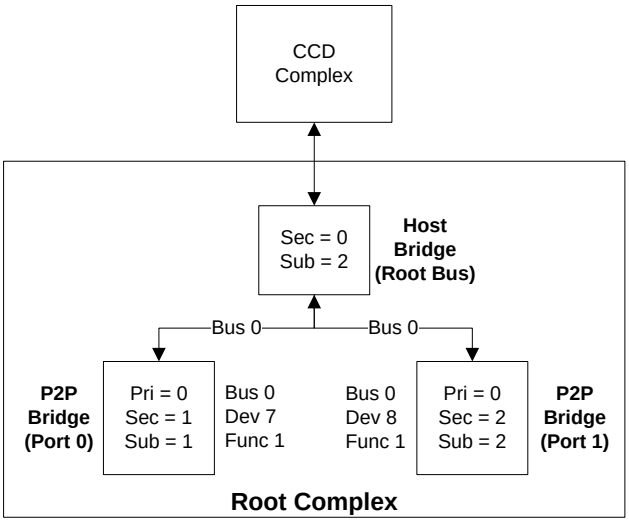


Figure 11. Sample Root Complex

```
Scope (\_SB) {
  Device (PCI0) { // Root PCI Bus (Host-Bridge)
    Name (_HID, EISAID ("PNP0A08"))
    Name (_CID, EISAID ("PNP0A03"))
    Name (_BBN, 0)
    Method (_CRS,0) {
      // Return current resources for host bridge 0
    }
    Name (_PRT, Package() {
      // Package with PCI IRQ routing table information
    })

    Device (P2P0) { // First PCI-to-PCI bridge (Port0)
      Name (_ADR, 0x00070001) // Device#7h, Func#1 on bus PCI0
      Name (_PRT, Package() {
        // Package with PCI IRQ routing table information
      })
      Method (_PXM, 0, NotSerialized) {
          Return (PXM0)
      }
    }

    Device (P2P1) { // Second PCI-to-PCI bridge (Port1)
      Name (_ADR ,0x00080001) // Device#8h, Func#1 on bus PCI0
      Name (_PRT, Package() {
        // Package with PCI IRQ routing table information
      })
      Method (_PXM, 0, NotSerialized) {
        Return (PXM1)
      }
    }
  }
  // ...
}
```

Figure 12. _PXM Assignment at the Root-Port Level