Contents

[1. PCIe Introduction 2](#_Toc126529290)

[2. PIO and MMIO 2](#_Toc126529291)

[3. PCIe Lane Negotiation 2](#_Toc126529292)

# PCIe Introduction

The PCI Express bus is a backwards compatible, high performance, general purpose I/O interconnect bus, and was designed for a range of computing platforms. One of the key improvements of PCI Express, over the [PCI Local Bus](https://wiki.osdev.org/PCI), is that it now uses a serial interface (compared to the parallel interface used by PCI).

For specifications, refer - https://pcisig.com/specifications/

# PIO and MMIO

PIO and MMIO operate in different address spaces.

PIO –

PIO (port I/O) uses special assembly instructions to communicate with a dedicated IO bus. All devices attached to the PIO bus has a small 16-bit address.

MMIO –

In case of MMIO, device’s registers are also mapped into RAM’s address space.

# PCIe Lane Negotiation

* PCIe link between two devices can be 1 to 32.
* In multi lane link, packet data is stripped across lanes.
* Lane count is automatically negotiated during device initialization and can be restricted by Endpoint due to this single lane device can be used in multi-lane slot.
* Number of lanes in a link can also be reduced dynamically.

# Configuration space size: -

For each function in a device, 4K size is allocated for configuration space. If a device is having multiple functions, then for each function 4k bytes space will be allocated.

For example, if a device has three functions, then 3x4K = 12K bytes of space will be allocated for configuration space.

# PCIe topology: -

CPU

RC: 0:0:0

Device 1

Bridge

Device 2

DS

Bridge

Device 3

Device 4

US

US

DS

EP

Bus - 0

Bus - 1

0:1:0

0:2:0

0:3:0

1:0:0

1:1:0

1:2:0

Bus - 2

Legend: B:D:F(Bus Device Function)

If we want to see tree output of PCI topology in our target, then we can run below command:

# lspci -tv

# PCI Utilities

Repository for PIC utilities like “lspci”, “setpci” etc is maintained in: -

<https://github.com/pciutils/pciutils>

When we clone this, then we also get “lib/” folder in repository. Cloned source code also has a file – example.c, which shows how to use libpci.

# Why Multiple BARs in a device

PCIe Endpoints can have maximum 6 BARs. But Bridges can have only 2 BARs. Address assigned to BAR is physical address.

Note – Non prefetchable means it can’t use caches for this memory.

Text, letter

Description automatically generated