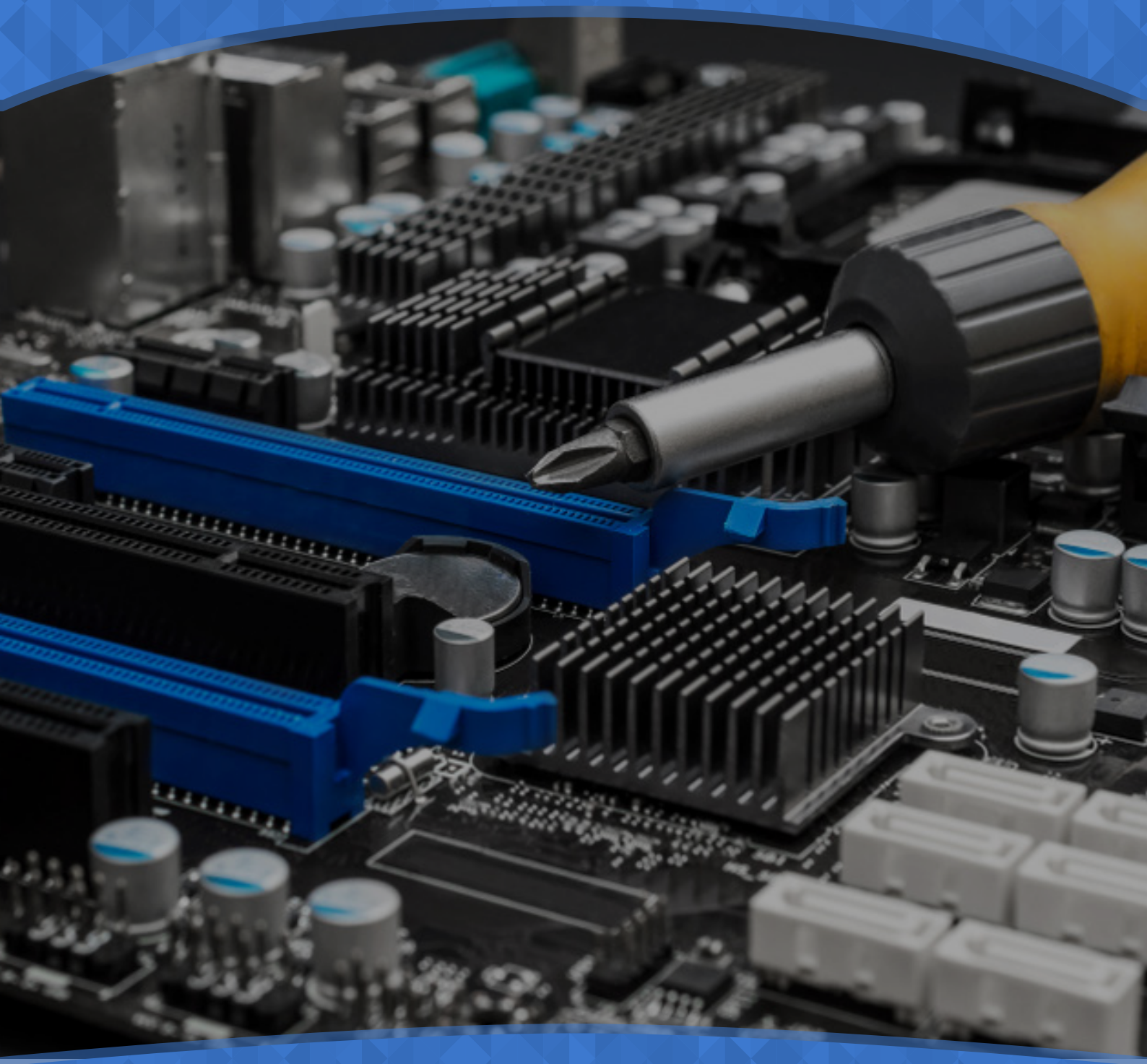
logic fruit Technologies

Whitepaper

PCIe VIRTUALIZATION STACK OVERVIEW



Here, the Root ports of the root complex are directly connected to the physical functions with no virtualization and resource sharing.

SINGLE ROOT IO VIRTUALIZATION

SRIOV- The capability for a single PCIe component to be used by more than one SI (System image). This functionality is defined in the Single Root I/O Virtualization and Sharing Specification.

In simple words - An OS or PC can be multiplied in the form of a virtual machine to reduce power consumption and cost. To give them network connectivity, they share the same NIC, but this will be a problem because network speed will be reduced and high CPU overhead. So to counter this, we virtualize NICs. NICs will have only one physical socket but will appear as multiple NICs, and all the virtual NICs can be given to individual Virtual machines.

- The single root I/O virtualization (SR-IOV) interface extends to the PCI Express (PCIe) specification. SR-IOV allows a device, such as a network adapter, to separate access to its resources among various PCIe hardware functions. These functions consist of the following types:
  - PF- This function is the device's primary function and advertises the device's SR-IOV capabilities. The PF is associated with the Hyper-V parent partition in a virtualized environment.
  - VF- Each VF is associated with the device's PF. A VF shares one or more physical resources of the device, such as a memory and a network port, with the PF and other VFs on the device.
- Single Root means SR-IOV device virtualization is possible only within one computer. It refers to the PCI Express root complex, the core PCI component that connects all PCI devices. PCI devices, bridges, and switches are cascaded off the root complex, creating a tree structure.
- Virtualization is used for improving server utilization, that is, by putting more software workloads onto a physical server to use up spare capacity. These software workloads are virtual machines running on top of a hypervisor like a kernel virtual machine or VMware. Virtualization reduces power consumption and costs and is a perfect fit for multi-core processors (MCPs) which often run one virtual machine (VM) per core.

What is a virtual machine?

Virtual Machine is an entirely separate individual operating system installation on your usual operating system. It is implemented by software emulation and hardware virtualization.

The main advantages of virtual machines:

- ☞ Multiple OS environments can exist simultaneously on the same machine, isolated from each other.
- ☞ Virtual machines can offer an instruction set architecture that differs from real computers.
- ☞ Easy maintenance, application provisioning, availability, and convenient recovery.
- ☞ Providing network connectivity to VMs on heavily virtualized servers is a challenge. The hypervisor can share NICs between the VMs using software, but at reduced network speed and with high CPU overhead.
- ☞ A better approach is to build a single NIC that appears as multiple NICs to the software. It has one physical ethernet socket but appears on the PCI Express bus as multiple NICs.
- ☞ Such an I/O-Virtualization-capable-NIC replicates the VM-facing hardware resources like ring buffers, interrupts, and Direct Memory Access (DMA) streams. The SR-IOV standard defines how these hardware resources are shared so hypervisors can find and use them in a standard way for all makes and models of SR-IOV-capable NICs. The SR-IOV standard calls the master NIC the Physical Function (PF) and its VM-facing virtual NICs the Virtual Functions (VFs).

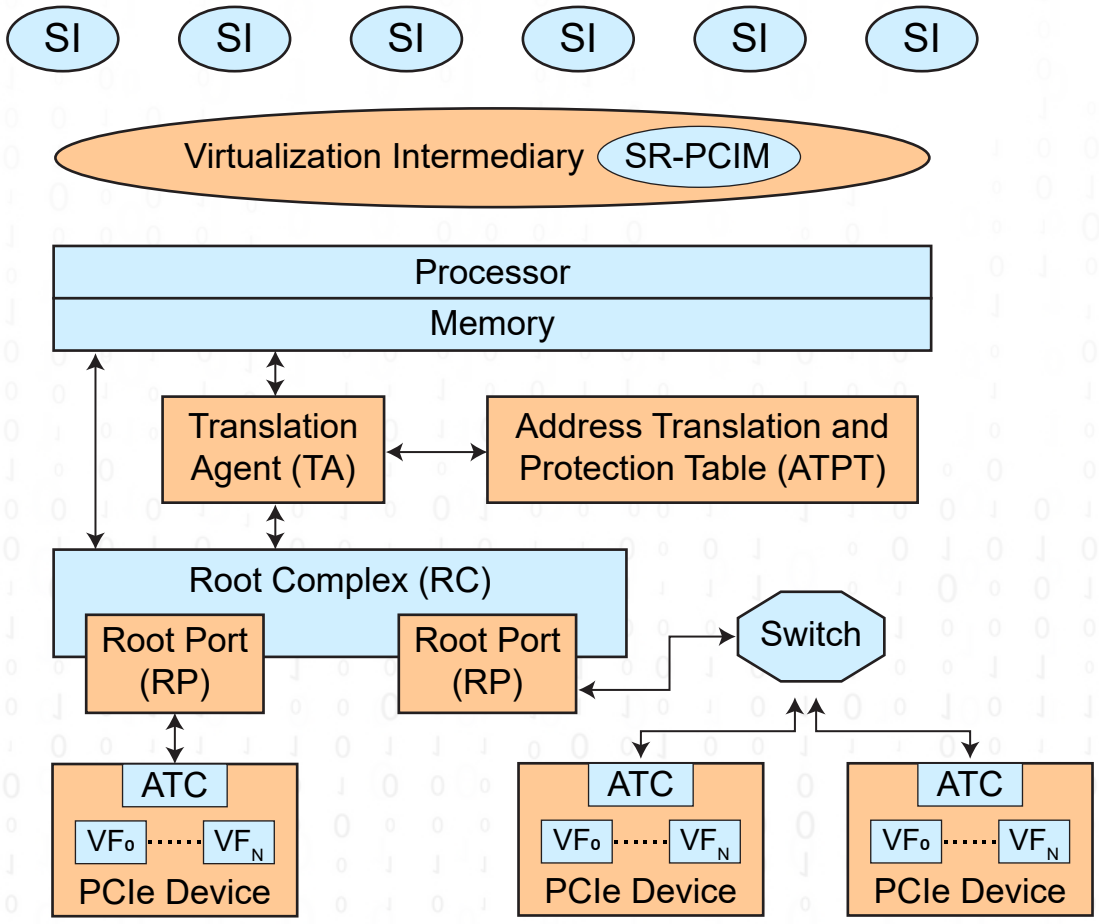


Figure - 02

In **fig. - 02** above, we can see that the physical functions are split and shared in virtual functions so that, at the same time, more system images can access the PCIe resource on the same hardware.

The PCIe Devices support the SR-IOV capability defined in the Single Root I/O Virtualization and Sharing Specification. SR-IOV enables a PCIe Device to support multiple Virtual Functions (VFs).

MRIOV - MULTI ROOT IO VIRTUALIZATION

MRIOV- The capability for a single PCIe component to be used by more than one Hierarchy Domain. Additionally, for Root Ports, the capability for a single Root Port to support more than one Hierarchy Domain.

In simple words - When we try to use PCIe hardware with multiple hosts, each host has a unique hierarchy. Suppose we have four hosts; then that implies we have four hierarchies. All the transactions will be based on hierarchy. For ex-, when host 0 (VH 0) sends a request, then EP will process it and send a completion. Now, this completion can be claimed by any of the four hosts; here, the hierarchy comes into play since the request had VH0, so it implies that host PC 0 was the requester.

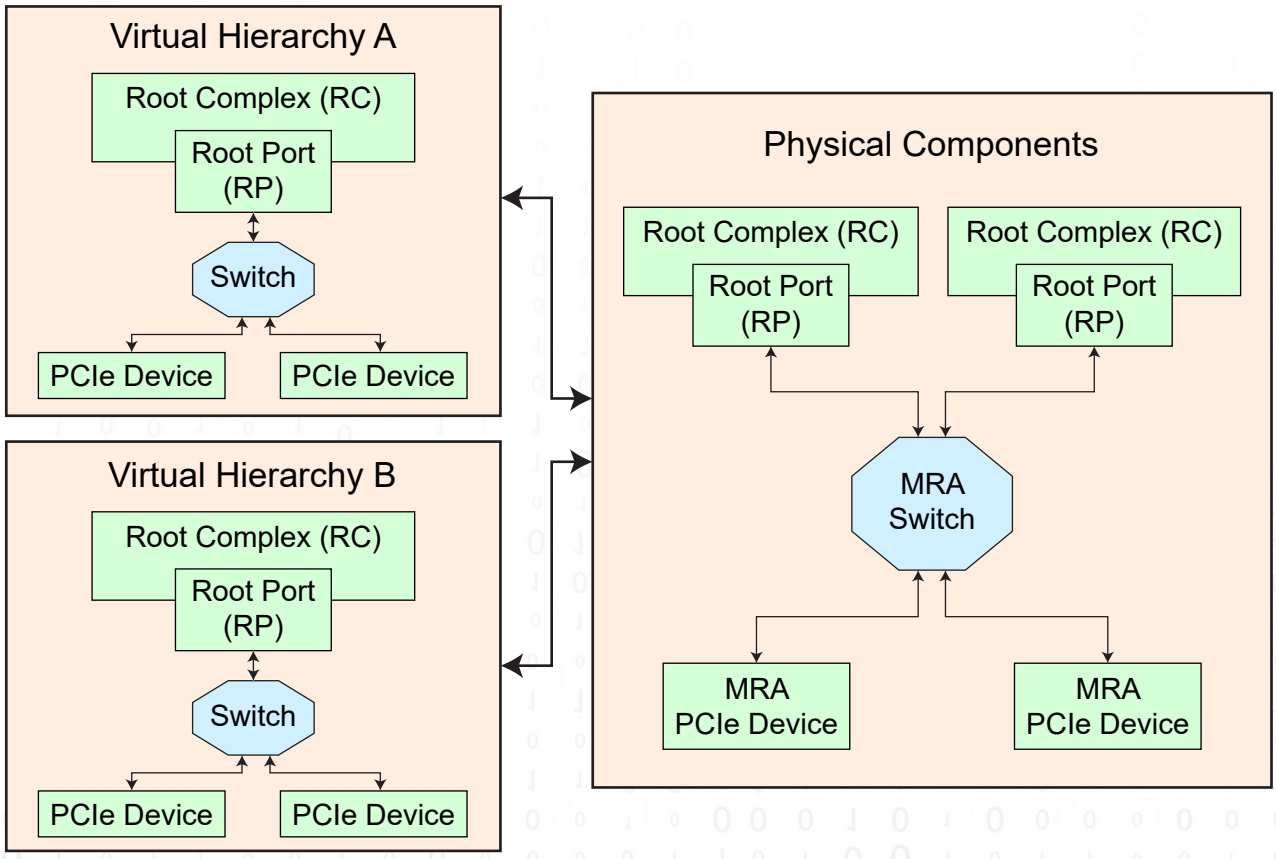


Figure - 03

In fig. 03 above, we see how to utilize the resource MRA switch to share the same resource with different ROOT PORTS.

How does multi-root virtualization work?

- ☞ Multiple hardware domains utilize the same IO Endpoint.
- ☞ EP will have Physical functions along with virtual functions.

Multi Root I/O Virtualization (MR-IOV) — by contrast — is concerned with sharing PCI Express devices among multiple computers.

Multiple MRA PCIe Switches can be provisioned and interconnected into root ports to access shared PCIe devices.

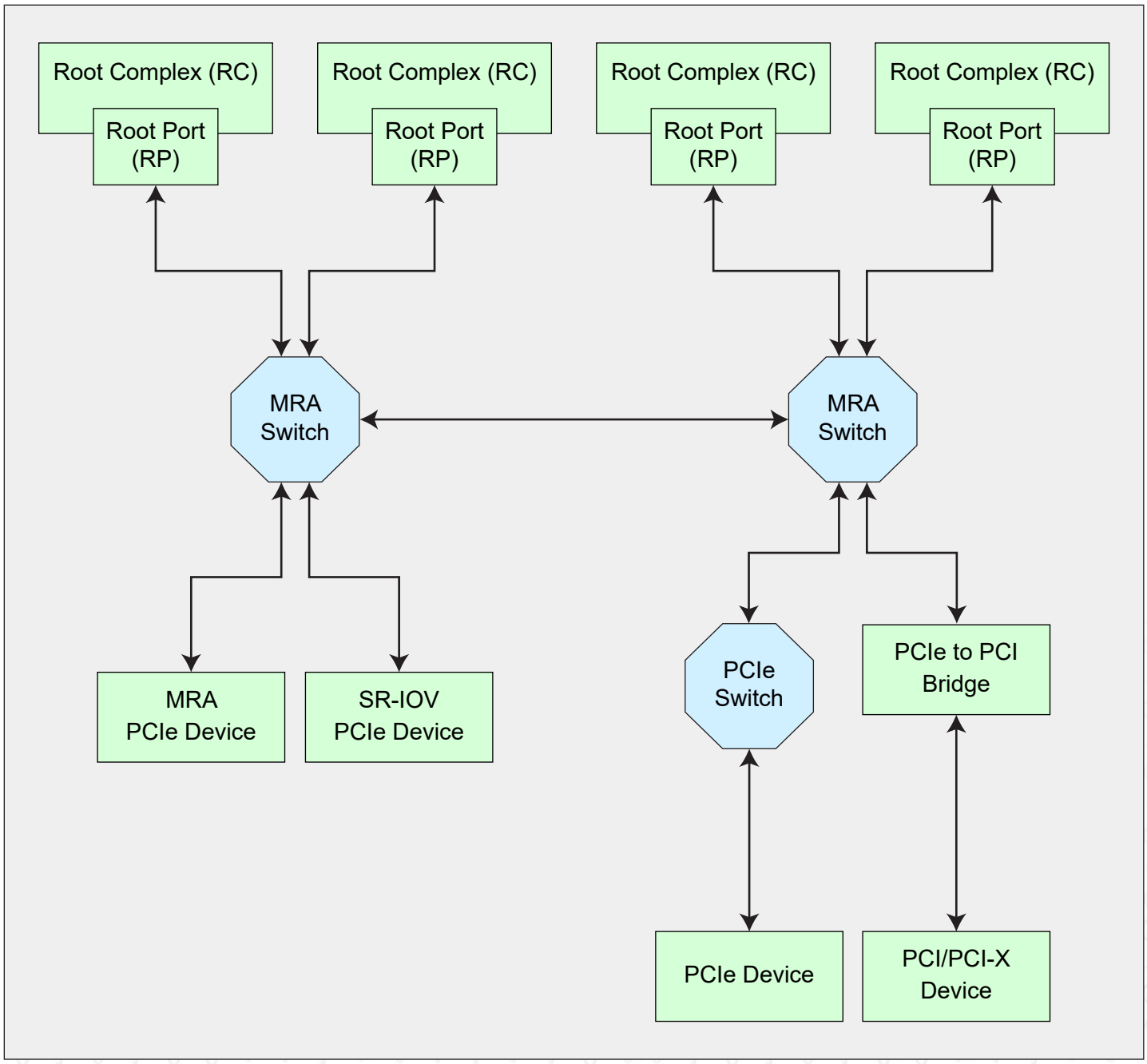


Figure - 04

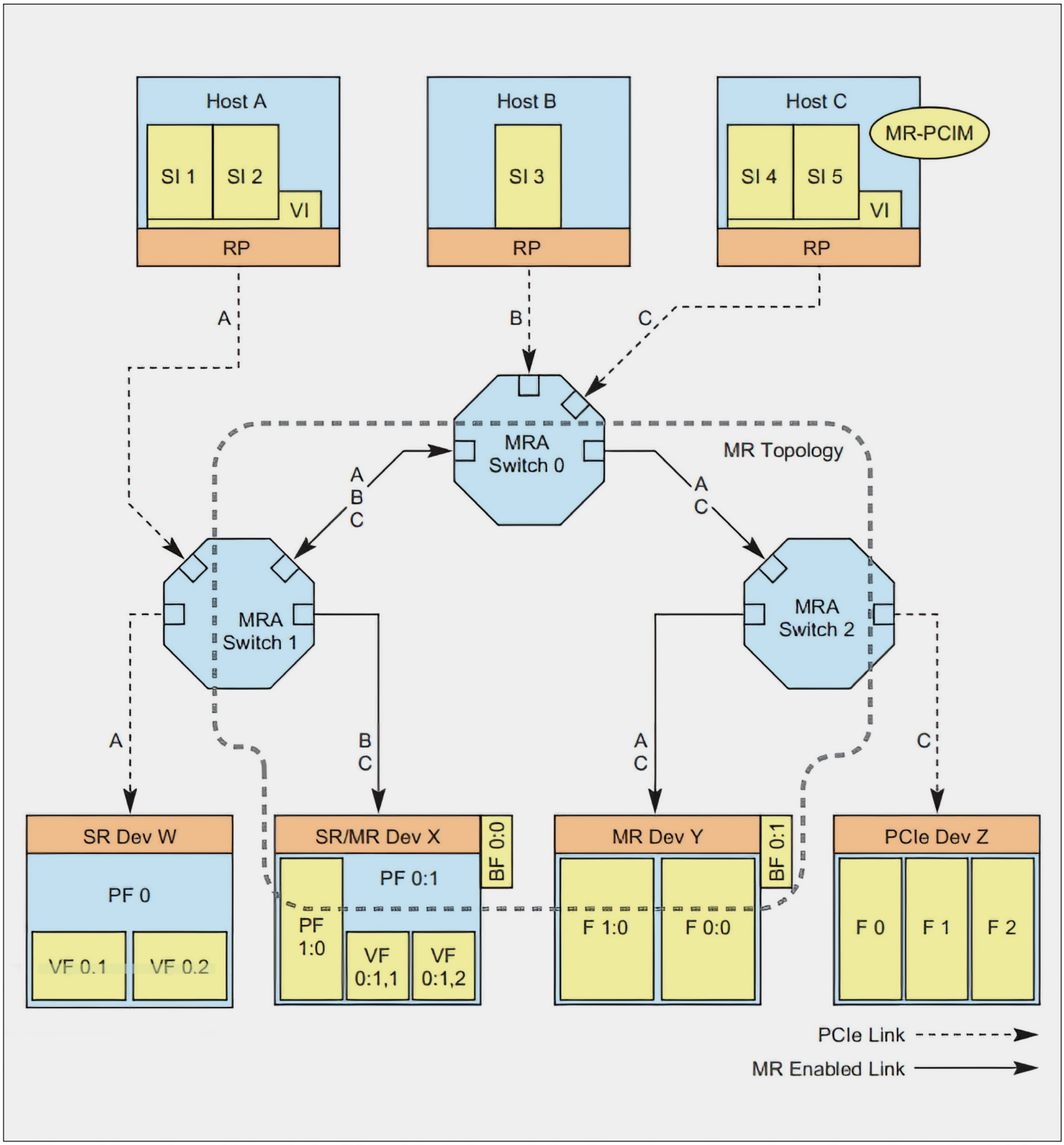


Figure - 05

Fig. 04 and 05 Above show how different hosts access the shared PCIe resources via MRA switches. All the hosts have a virtual hierarchy. In simple words, host PC 1 will have VH0; Host PC 2 will have VH1... and so on.

What difference Between the Different Physical Functions??

Each PF is independent of others and is seen by software separately. Each has a separate 256-byte config space (4Kb for extended cfg space).

Regarding the Physical layer capabilities like (PHY 16/32 GT ext capabilities, Lane margining), according to spec, they must be implemented in Function 0 (and only Function 0) of a Multi-Function Device associated with an Upstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s or higher.

The current implementation in the codebase : The connections are made only with Function 0.

The Lane Margining at the Receiver Extended Capability structure must be implemented in:

- ☞ A Function associated with a Downstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s or higher.
- ☞ A Function of a single-Function Device associated with an Upstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s or higher.
- ☞ Function 0 (and only Function () of a Multi-Function Device associated with an Upstream Port where the Supported Link Speeds Vector field indicates support for a Link speed of 16.0 GT/s or higher.

How is virtual function different from Physical Function??

Virtual functions are called lightweight PCIe PF because they do not have their own config space (256 bytes and 4kb ext space). They utilize the cfg space of PF only. But with one addition in PF cfg space. The addition is an extra capability.

The function which supports SR-IOV has one additional capability in PF's cfg space - SR-IOV extended capability. This capability has a specific command and status register that enables VF and tells the other devices how many VFs are supported in that specific function. Also, there are dedicated registers to program BAR of VF (similar to BAR In cfg space at offset 0x10, 0x14...) to allocate mem space and additional registers.

The VF has just enough resources to transmit and receive the data. And by the resources means queue, interrupts, and descriptors like BAR only.

The VM load up a VF driver when vm starts and reads the cfg space, and the hypervisor says it has virtual functions; the driver then loads up and fills up the descriptors like BARs to tell where the memory mapping is to be done. The VF provides queues. Formerly, a single core was used for all processing of VM but now, with the queue and sorting in VF itself, the hypervisor doesn't touch each pkt and copies it physically.

SPEC RELATED STATEMENTS

1. SR-PCIM does SRIOV resources management.

SR-IOV provides tools to reduce these platform resources overheads. The benefits of SR-IOV are:

- The ability to eliminate VI involvement in main data movement actions - DMA, Memory space access, interrupt processing, etc. Elimination of VI interception and processing of each I/O operation can provide significant application and platform performance improvements.

- Standardized method to control SR-IOV resource configuration and management through Single Root PCI Manager (SR-PCIM).
  - Due to a variety of implementation options - system firmware, VI, operating system, I/O drivers, etc. - SR-PCIM implementation is outside the scope of this specification.

- SR-PCIM - Software responsible for the configuration of the SR-IOV Extended Capability, management of Physical Functions and Virtual Functions, and processing of associated error events and overall device controls such as power management and hot-plug services.

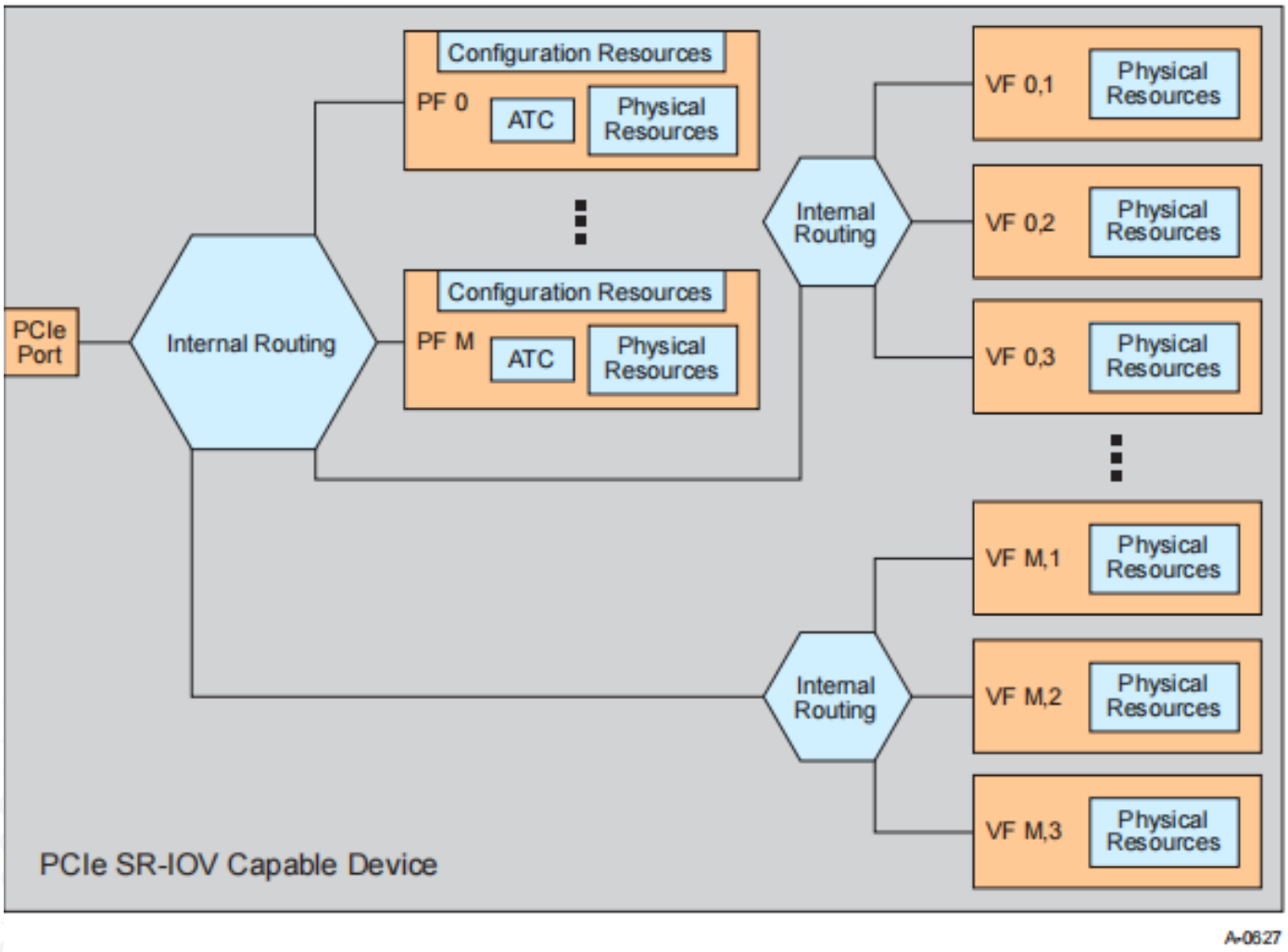
- Physical Function (PF) - A PF is a PCIe Function that supports the SR-IOV Extended Capability and is accessible to an SR-PCIM, a VI, or an SI.

- Virtual Function (VF) - A VF is a "light-weight“ PCIe Function that is directly accessible by an SI.
  - Minimally, resources associated with the main data movement of the Function are available to the SI. Configuration resources should be restricted to a trusted software component such as a VI or SR-PCIM.

☞ Each VF shares a number of common configuration space fields with the PF; (i.e., where the fields are applicable to all VF and controlled through a single PF. Sharing reduces the hardware resource requirements to implement each VF.)

- AVF uses the same configuration mechanisms and header types as a PF.
- All VFs associated with a given PF share a VF BAR set (see Section 9.3.3.14) and share a VF Memory Space Enable (MSE) bit in the SR-IOV extended capability (see Section 9.3.3.3.4) that controls access to the VF Memory space. That is, if the VF MSE bit is Clear, the memory mapped space allocated for all VFs is disabled.

☞ Each VF contains a non-shared set of physical resources required to deliver Function-specific services, (e.g., resources such as work queues, data buffers, etc.) These resources can be directly accessed by an Sl without requiring VI or SR-PCIM intervention.



Config resources are given only to PF in the figure above.

9.2.1.1 Configuring SR-IOV Capabilities

This section describes the fields that must be configured before enabling a PF's IOV Capabilities. The VFs are enabled by Setting the PF's VF Enable bit (see Section 9.3.3.3.1) in the SR-IOV extended capability.

The NumVFs field (see Section 9.3.3.7) defines the number of VFs that are enabled when VF Enable is Set in the associated PF.

How Resource Sharing is Happening for Virtual as well as Physical Function??

The VF shares the same cfg space as the PF, and SR-IOV capability tells the other device during enumeration that the device has VF enabled along with other information.

The memory space gets allotted based on the BAR programmed in cfg space for PF and in BAR of SR-IOV capability for VF.

Logic Fruit Expertise

Logic Fruit has vast experience and expertise in PCIe Virtualization and, thus, can support the following:

1. Development of an SRIOV/MRIOV related Product,
2. Validation of an SRIOV/MRIOV-related Product,
3. Prototyping for a specific requirement.

Logic Fruit will also come up with solutions for Virtualization soon.

logic fruit Technologies

Thank You!

Does anyone have any questions?

Contact Us

Gurugram (Headquarter)

806, 8th Floor
BPTP Park Centra Sector–30,
NH–8 Gurgaon – 122001
Haryana (India)

info@logic-fruit.com

+91-124 4643950

Bengaluru (R&D House)

Sy. No 118, 3rd Floor,
Gayathri Lakefront,
Outer Ring Road, Hebbal,
Bangalore - 560 024

sales@logic-fruit.com

+91 80-69019700/01

United States (Sales Office)

Logic Fruit Technologies
INC 691 S Milpitas Blvd
Ste 217 (Room 9) Milpitas
CA 95035

info@logic-fruit.com

+1-408 338 9743