Multicore Processing: Virtualization and Data Center

Syed Shah, Nikolay Guenov

Virtualization and Its Impact

Virtualization is a combination of software and hardware features that creates virtual CPUs (vCPU) or virtual systems-on-chip (vSoC). These vCPUs or vSoCs are generally referred to as virtual machines (VM). Each VM is an abstraction of the physical SoC, complete with its view of the interfaces, memory and other resources within the physical SoC. Virtualization provides the required level of isolation and partitioning of resources to enable each VM. Each VM is protected from interference from another VM. The virtualization layer is generally called the virtual machine monitor (VMM).

Why Virtualization?

Virtualization has already impacted the server and IT industries in a significant way. IT organizations are using it to reduce power consumption and building space, providing high availability for critical applications and streamlining application deployment and migration. The trends to adopt virtualization in the server space also are being driven by the desire to support multiple operating systems and consolidation of services on a single server by defining multiple VMs. Each VM operates as a stand-alone device. Because multiple VMs can run on a single server (provided the server has enough processing capacity), IT gains the advantage of reduced server inventory and better server utilization.

Virtualization helps to:

- Run multiple operating systems on a single computer, including Windows®, Linux® and more
- Increase energy efficiency, reduce hardware requirements and thereby reduce overall capital expenditure
- Determine highest availability and performance for enterprise applications
- Use computing resources efficiently

Virtualization in Embedded Space

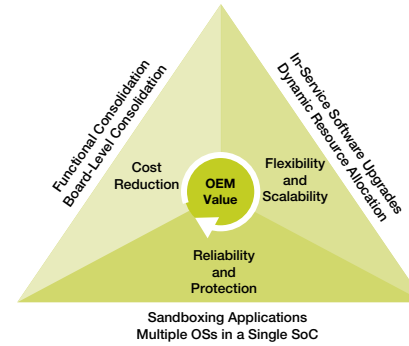
Although they are not mainstream, these IT industry trends are trickling down into the embedded space as well. Observed trends for virtualization in the embedded space include:

- The concept of having a sea of processors, and the associated processing capacity sliced and diced among applications and processes, is no longer science fiction.
- The challenges of extracting higher utilization from the processors, and consolidation triggered by cost reduction, are driving the adoption of virtualization in embedded systems.
- With virtualization, one can merge the control and data plane processing onto the same SoC. Previous approaches used separate discrete devices for these functions.

Virtualization offers three major benefits to the embedded industry:

1. Cost reduction via consolidation
2. Flexibility and scalability
3. Reliability and protection

Figure 1: Benefits of Virtualization

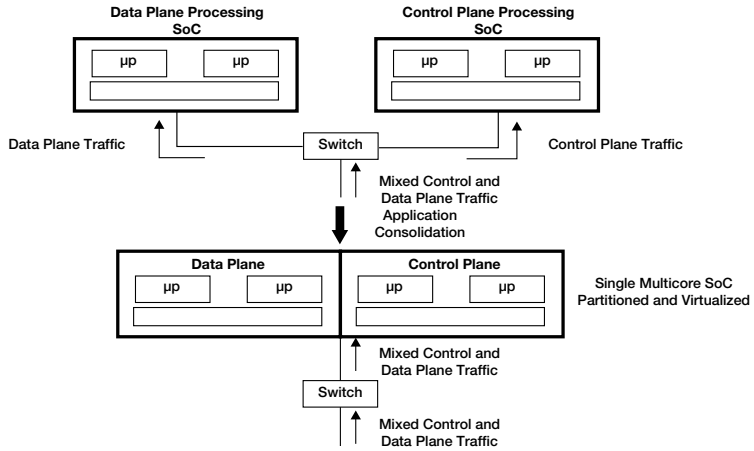


Virtualization and Multicore Processing

With multicore SoCs, given enough processing capacity and virtualization, control plane applications and data plane applications can be run without one affecting the other. Data plane and control plane applications, in most cases, will be mapped to different cores in the multicore SoC as shown in Figure 2.

Control and data plane applications are not the only application-level consolidation that will occur. Virtualization and partitioning will allow OEMs to enable their customers to customize service offerings by adding their own applications and operating systems to the base system on the same SoC, rather than using another discrete processor to handle it. Data or control traffic that is relevant to the customized application and operating system (OS) can be directed to the appropriate virtualized core without impacting or compromising the rest of the system.

Figure 2: Control and Data Plane Application Consolidation in a Virtualized Multicore SoC



Another example of consolidation of functions is board-level consolidation. Functions that were previously implemented on different boards now can be consolidated onto a single card and a single multicore SoC. Virtualization can present different virtual SoCs to the applications. With increasing SoC and application complexity, the probability of failures due to software bugs and SoC mis-configuration are greater than purely hardware-based failures. In such a paradigm, it may make sense to consolidate application-level fault tolerance onto a single multicore SoC, where a fraction of the cores are set aside in hot standby mode. While such a scheme will save the cost of having to develop a standby board or at the very least another SoC, it would require the SoC to be able to virtualize not only the core complex but also the inputs/outputs (I/Os).

Although virtualization has its advantages, it comes with new challenges and considerations, including partitioning, fair sharing and protection of resources between multiple/competing applications and operating systems. The following sections will discuss how virtualization technology addresses these challenges.

Addressing Challenges with Virtualization

Addressing Partitioning Challenge

Partitioning can be defined as subdividing resources of a SoC in a manner that allows the partitioned resources to operate independently of one another. Partitioned resources can be mapped either explicitly to the actual hardware or to the virtualized hardware. Note that the system can be partitioned without being virtualized. For example, in a SoC that allows partitioning but not virtualization, each Ethernet interface can be assigned to a partition but a single Ethernet interface cannot be assigned to two different partitions at same time. However, if the SoC also provides virtualization capabilities, then a single Ethernet interface can be virtualized and each virtual Ethernet interface can be presented to a different partition.

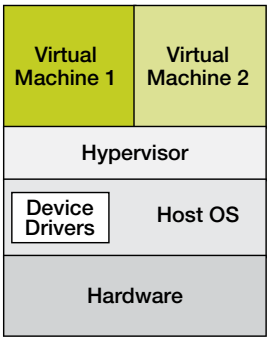
Addressing Fair Sharing and Protection Challenge

A hypervisor is a software program that works in concert with hardware virtualization features to present the VM to the guest OS. It is the hypervisor that creates the virtualization layer. There are two broad architectural approaches when it comes to virtualizing the system: 1) OS-hosted and 2) bare-metal hypervisor. Each approach has its pros and cons and the choice would depend on the applications and the market segments.

OS-Hosted Hypervisor

The hypervisor schedules the guest operating systems based on the scheduling policies in the hypervisor. Scheduling of the VMM and guest operating systems is dependent on the scheduling policies of the host OS, because they run on top of the host OS.

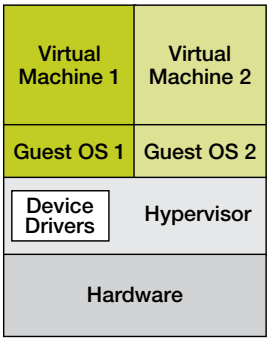
Figure 3: OS-Hosted Hypervisor



Bare-Metal Hypervisor

The bare-metal hypervisor approach does not depend on the host OS and runs directly on the physical hardware (bare metal). The hypervisor fully controls the SoC, enabling it to provide quality of service guarantees to the guest operating systems.

Figure 4: Bare-Metal Hypervisor



Different I/O handling approaches can be used by the hypervisor, including fully virtualized I/O, dedicated I/O and para-virtualized I/O. In the fully virtualized approach, the hypervisor virtualizes the I/O by emulating the devices in software. The software overhead thus created reduces the efficiency of the system. In the para-virtualized approach, the I/O interfaces are not fully virtualized.

The key difference between full I/O virtualization and para-virtualization is that not all functions are emulated in para-virtualization; hence, this approach reduces software overhead at the cost of OS portability. In the dedicated approach, each VM is assigned a dedicated I/O in its own partition and does not have to go through the hypervisor for I/O transactions once set up, resulting in the lowest software overhead.

Unlike servers or compute-centric systems, one key design metric for embedded systems is performance of the system per watt of power dissipation. That is, the system should be optimized to extract the best possible performance within a given power budget. Usually the power budget of embedded systems is more constrained than that of the servers or compute-centric systems. While portability and flexibility are important, often they are not the number one concern. As such, the bare-metal hypervisor approach offers the best virtualization solution for embedded systems.

In summary, while the OS-hosted approach offers the greatest application and guest OS portability, the bare-metal hypervisor approach offers the best performance and the lowest virtualization overhead.

Advantages of Virtualization

On the processor, a new hypervisor state introduced by Freescale and some traditional compute-centric companies automatically traps privileged and sensitive calls by the guest OS to the hypervisor, removing the need for binary code rewriting or para-virtualization. This reduces the complexity and overhead introduced by the hypervisor and also improves overall performance of the system.

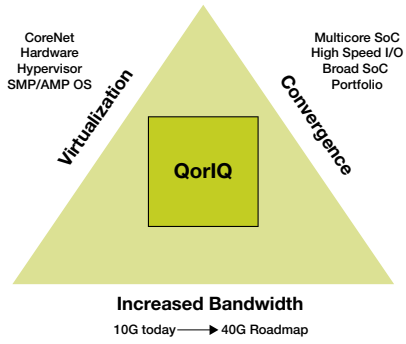
The second area where hardware support improves hypervisor performance is memory management. In an un-virtualized system, a virtual memory address is formed by using the effective address. The virtual address is then translated to the physical address by the memory management unit. The system uses the physical address to either fetch or store the data or instruction from memory. If the system has multiple processes running on it, another level of abstraction is introduced, and the virtual address is calculated using the effective address and the process ID. That is, different processes having the same effective address will be represented by a different virtual address and consequently a different physical address.

A virtualized system can have many VMs running on a single system and each VM can have multiple processes running on it. Memory is allocated to each VM and in turn to the different processes running on each VM. A virtualized system, therefore, introduces yet another level of memory abstraction. While this abstraction can be accomplished using software, hardware assistance will greatly improve the performance of the system. Freescale and some compute-centric companies have introduced enhancements to their memory management units that allow multiple levels of memory abstractions.

Based on the system and the application, memory associated with a VM may need to be protected and partitioned from the other VMs. For example, in a mixed control and data plane system where control and data plane applications run on the same SoC, it is imperative that memory be protected and partitioned so that these applications run completely independently and the state of one partition cannot erroneously or maliciously corrupt the state of the other partition.

Finally, the last major area where hardware support for virtualization will greatly reduce hypervisor overhead is the I/O. With increasing desire to consolidate applications and workloads onto single multicore SoCs, the density of VMs per SoC is bound to increase. However, this trend can have significant performance impact as more and more applications become bound by network I/O. In traditional virtualized systems, the hypervisor manages I/O activity of different VMs by using a virtual switch. The virtual switch resides inside the hypervisor and is responsible for negotiating traffic exchange between the VMs and the I/Os. The virtual switch parses and classifies incoming packets and forwards them to the appropriate VM. The virtual switch does the same for the packets it receives from the VMs, and forwards them to the appropriate network I/O. As the number of VMs increase and network I/O speeds move from 1 Gigabit Ethernet (GbE) to 10 GbE, the number of CPU cycles required by the virtual switch to forward packets to and from the VMs and the I/O will increase substantially—reducing the amount of CPU cycles available for the applications running on the VMs. Hardware support that eliminates the need for the virtual switch, or significantly reduces the burden on it, will inevitably improve the performance of the system.

Figure 5: Key Features of Freescale's Data Center Solution



Freescale's Virtualization Solution

Discrete hardware-based solutions have been proposed by some companies in compute-centric industry. Freescale, on the other hand, has taken an integrated data path architecture approach for the embedded market in its QorIQ products. QorIQ processors provide an integrated on-chip solution using a combination of in-line hardware accelerators to parse, classify and queue packets to different VMs and processors. Traffic from multiple network interfaces can be directed to different partitions and VMs on the systems. QorIQ processors also offer built-in scheduling and priority mechanisms for the system to fairly distribute traffic among different partitions and VMs, as well as to allow policy-based sharing of the network I/Os by different VMs. In a nutshell, Freescale's approach is to allow partitioning and virtualization of SoC, provide performance isolation between different partitions and virtualized SoC and manage and protect those resources.

Multicore Processors in New Generation Data Center Solutions

Figure 5 shows key aspects of Freescale's data center solution based on virtualization.

- The CoreNet, hardware hypervisor and SMP/AMP OS technologies of QorIQ processors help enable virtualization required for data center networking.
- With its multicore architecture, high-speed I/O and broad SoC portfolio, QorIQ processors facilitate convergence.
- With the progressively increasing speed and processing capacities of QorIQ processors, Freescale is targeting 40G standards from the current mark of 10G today.

Key differentiating features of Freescale's data center networking solution are:

- Data centers focused on reducing power and cost
- Network node consolidation driving multiple functions into fewer platforms
- Integrated service routers adding appliance capabilities
- Management of multiple devices is usually difficult and time consuming

Freescale Enterprise Networking and Data Center Strategy

Freescale's data center approach provides embedded solutions for:

- Switching platforms
- Storage platforms
- Application delivery controllers
- Intelligent network interface controllers (NICs) and converged network adapter (CNAs)
- Low power servers

Freescale is uniquely positioned to enable common platforms through a broad portfolio of integrated SoC solutions for control and data path processing. Scalable multicore processor solutions facilitate common platform architectures across OEM portfolios, maximizing OEM hardware and software investments with best-in-class tools and software ecosystem including commercial solutions from Freescale VortiQa and third-party partners.

Figure 6: Freescale's Enterprise Networking and Data Center Strategy

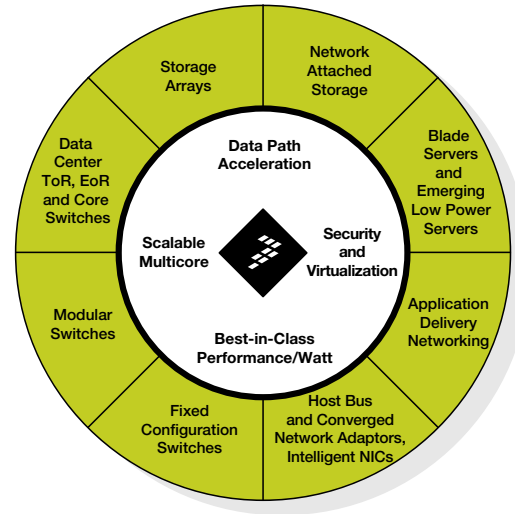


Table 1: Freescale's Enterprise Networking and Data Center Product Offerings

	Main Functions	New Technologies	Freescale Offering
Network	• IP switching • WAN optimization • Application delivery	• High bandwidth and throughput virtualization	• MPC83xx, MPC85xx, P10xx, P20xx, P30xx, P4080, P5020
Security	• Security/UTM • Appliance • High-bandwidth firewall	• IDS/IPS • High-bandwidth SSL	• MPC8572, P1020, P2020
Storage	• SAN • Storage controllers • NAS	• Virtualization of resources • On-demand provisioning • Improved performance	• P3041, P4080, P5020
Computing	• Server controllers	• High bandwidth • Virtualization	• MPC85xx -> P10xx -> P2040 • P3041 -> P4080 • P5020
Application	• Server processors	• Large multicore complex cluster cores	• P4080

In the current era of cloud computing, there has been increased demand to have a more robust and secure data center. Understandably, one of the primary concerns that a company has while implementing a data center is to sustain business continuity. Because every company has a tremendous reliance on its IT operations and because many of these IT operations rely on data centers, it is extremely vital for these data centers to be available all the time throughout the year. With its broad range of current and future QorIQ and PowerQUICC III processors and cutting edge technologies such as high bandwidth and throughput, virtualization, high bandwidth SSL and on-demand provisioning for improved performance, Freescale has the capacity to meet the requirements of the most robust data centers (see Table 1).

Conclusion

Proliferation of multicore processors in embedded markets and the desire to consolidate applications and functionality will push the embedded industry into embracing virtualization in much the same way as it occurred with the server and compute-centric markets.

Differences in the characteristics of embedded and compute-centric markets warrant different virtualization approaches. The embedded market, unlike the server and compute-centric markets, is sensitive to the power envelope a particular device can dissipate and usually has very constrained power budgets as compared to those in the compute-centric space. One of the primary design objectives in the embedded market is to maximize performance per watt, so it is desirable to offload as many functions to hardware as possible and free CPU cycles to be allocated to applications. When it comes to virtualization, the same philosophy is applied. While software-based solutions would work fine in the server and compute-centric markets, they are more likely to degrade the performance of an embedded system to unacceptable levels. In embedded markets, the bare-metal hypervisor-based approach coupled with hardware virtualization assists in the core, the memory subsystem and the I/O appears to offer the greatest performance over other approaches.

How to Reach Us:

Home Page:
freescale.com

Power Architecture Portfolio Information:
freescale.com/power

e-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
1-800-521-6274
480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014
+81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate,
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor
Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447
303-675-2140
Fax: 303-675 2150
LDCForFreescaleSemiconductor@hibbertgroup.com

freescale™

For more information, visit freescale.com/power

Document Number: PWRARBYNDBITSMP REV 0

BUILT ON Power™