White Paper

Freescale's Embedded Hypervisor for QorIQ™ P4 Series Communications Platform

Document Number: EMHYPQIQTP4CPWP
Rev 1
10/2008

freescale™ semiconductor

Overview

Freescale Semiconductor's QorIQ™ communications platform P4 series processors deliver industry-leading performance in the under 30-watt power category. These multicore processors combine eight Power Architecture® e500mc cores—operating at frequencies up to 1.5 GHz—with high-performance datapath acceleration logic, as well as networking I/O and other peripheral bus interfaces.

The QorIQ P4 series processors feature hypervisor software, which allows cores to run different operating systems. A hypervisor is a special low-level software program that facilitates secure partitioning. It acts as the partitions' resources and security manager, presenting a virtual machine to the OS running in each partition. A hypervisor may manage multiple virtual machines and partitions, even on a single core, in a manner similar to how an operating system switches processes.

Contents

1 Introduction

To realize the full potential of multicore systems, many usage scenarios require using multiple secure computing domains, often managed by different operating systems, on the same physical device. This can be accomplished by dividing the cores, memory and I/O devices of a system into secure, logical partitions. The partitions need to operate independently of each other, and operating systems must be able to access and manage the hardware resources belonging to the partition with little or no overhead.

A hypervisor is a special low-level software program that facilitates secure partitioning. It acts as the partitions' resources and security manager, presenting a virtual machine to the OS running in each partition. A hypervisor may manage multiple virtual machines and partitions, even on a single core, in a manner similar to how an operating system switches processes.

1.1 A Case for Partitioning

There are a variety of use cases that require multiple heterogeneous computing domains on a single multicore processor.

- A system with a control plane and data plane that was previously spread across multiple discrete chips can be consolidated into a single multicore processor.
- An open OS, such as Linux®, can be run alongside a proprietary OS, allowing the system the benefits of both operating systems.
- In order to support end-user installed software, partitioning may be required to isolate the untrusted software.
- Sensitive security tasks may need to be partitioned away from other partitions.

This model could potentially be addressed cooperatively, where all operating systems agree to stay within their boundaries and behave as peers. Unfortunately, the cooperative scheme is difficult to enforce. One OS failure can bring down another, and shared resources, such as interrupt controllers, can become problematic. Some applications require memory sharing between partitions while keeping other resources private to the partition and secure from interference. Using a hypervisor can more effectively isolate partitions and provide better security while managing resources shared among the partitions.

A hypervisor presents a virtual machine to each partition and allocates resources among the partitions. For example, one partition could be running a version of Linux while another partition could be running a real-time OS and a real-time application. What's more, each partition may be completely unaware of the other's existence. It's possible that the security and resource management can be accomplished with few changes to the operating systems, but the number of changes required depends on the design of the hypervisor.

2 Freescale QorIQ Platform Support for Virtualization and Partitioning

Freescale QorIQ multicore communications platforms are the next-generation evolution of the popular PowerQUICC® communications processors. Built using high-performance e500mc cores based on Power Architecture® technology, Freescale QorIQ platforms enable advanced networking innovations where reliability, security and quality of service are the primary design considerations.

The e500mc cores, developed specifically for QorIQ multicore platforms, operate at frequencies up to 1.5 GHz and feature embedded hypervisor technology, network and peripheral bus interfaces and high-performance datapath acceleration logic. Datapath acceleration IP manages packet routing, security, quality-of-service (QoS) and deep packet inspection, which frees the core to focus on value-added services and application processing. Each e500mc core has a 32 KB instruction and data L1 cache and a private 128 KB L2 cache.

The e500mc embedded hypervisor architecture introduces the concept of partitions with several architectural changes from previous generations of the e500 family of CPUs. The changes—processor (machine) modes and states, memory management, interrupts and others—are defined in such a way that a previous operating system that was programmed to run on an existing core bare-metal will continue to run bare-metal, even on a core that has implemented the embedded hypervisor features. This allows ISVs to continue to provide systems software in the same environment that existed before the embedded hypervisor was introduced. In addition, the Freescale-developed hypervisor presents a virtual machine very close to the actual underlying hardware, allowing operating systems to be implemented so that they can run under the control of a hypervisor or on bare-metal with few or no changes.

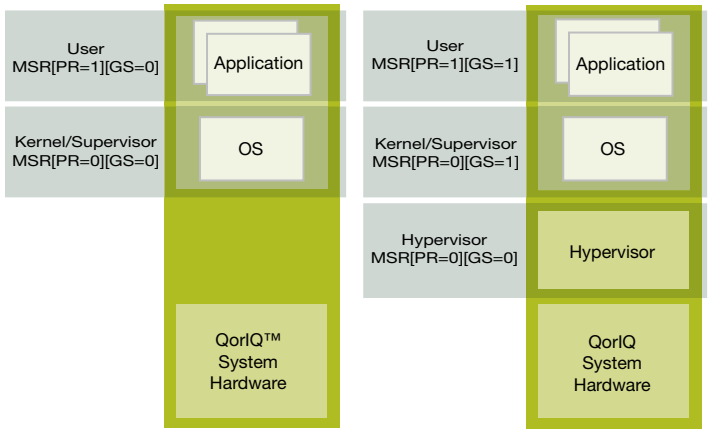
Changes introduced by the e500mc embedded hypervisor architecture effect the following groups:

- Processor (machine) modes and states
- Memory management
- Interrupts

2.1 Guest State

The e500mc embedded hypervisor architecture introduces a third privilege level called 'guest state' (GS bit in the e500mc machine state register). When a hypervisor is hosting an operating system, it runs in hypervisor state, and the hosted operating system and its applications run in guest state.

Figure 2.1: Freescale e500mc Embedded Hypervisor Privilege Levels



Freescale technology

For improved performance, the e500mc implements shadowed registers where certain performance-critical CPU registers are duplicated, with one designated for use by the hypervisor and the other by guests. In addition, they are remapped so that guest software need not be modified to use them. For example, SRR0 and GSRR0 registers are present, with GSRR0 accessible by guest software. The CPU automatically remaps guest accesses to SRR0 to GSRR0.

2.2 Memory Management

The e500mc embedded hypervisor architecture extends the virtual address space of the CPU to include a logical partition ID. This provides a hypervisor with a large virtual address space, which can be used to efficiently create partitions.

The QorIQ communications processor contains a peripheral access memory management unit (PAMU) that provides I/O device-to-memory access control, protection and address translation. This is a critical component in creating a securely partitioned system. The QorIQ communications processor can be configured so that for memory accesses all DMA-capable I/O devices must go through the PAMU. The PAMU accesses a set of software-configured tables that describe what physical address ranges the device is permitted to access.

2.3 Interrupts

The e500mc allows that some interrupts may be selectively directed to the guest state without any involvement by hypervisor software. These interrupts may be performance-critical, based on the software behavior as a whole and on the strategies that a hypervisor uses for memory management.

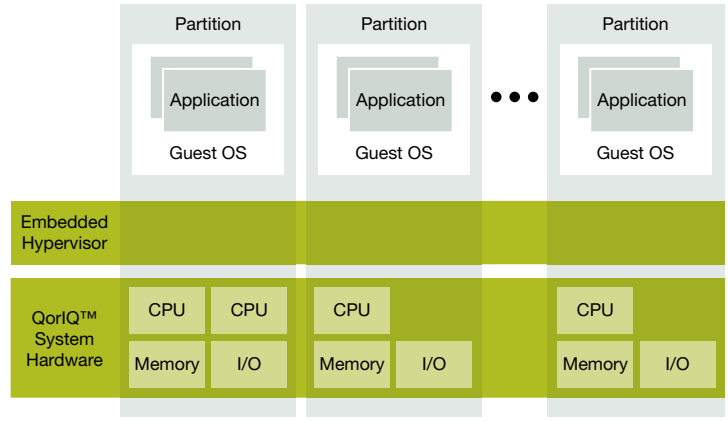
The following interrupts can be configured to go directly to guest state:

- External Input
- Data TLB Errors
- Instruction TLB Errors
- Data Storage
- Instruction Storage

3 Freescale Embedded Hypervisor Software

The Freescale embedded hypervisor is a layer of software that enables the efficient and secure partitioning of a multicore system. A system's CPUs, memory and I/O devices can be divided into partitions, with each partition capable of executing a guest operating system.

Figure 3.1: Freescale Embedded Hypervisor Software Layer



Freescale technology

Key features of the embedded hypervisor software architecture are summarized below.

- **Partitioning:** Support for partitioning of CPUs, memory and I/O devices
  - **CPUs:** Each partition is assigned one or more CPU cores in the system.
  - **Memory:** Each partition has a private memory region that is only accessible to the partition that is assigned the memory. In addition, shared memory regions can be created and shared among multiple partitions.
  - **I/O devices:** P4 series QorIQ processor I/O devices may be assigned directly to a partition (direct I/O), making the device a private resource of the partition, which provides optimal performance.

- **Protection and isolation:** The hypervisor provides complete partition isolation so that one partition cannot access the private resources of another. The QorIQ communications processor contains a PAMU which is used by the embedded hypervisor software to ensure device-to-memory accesses are constrained to allowed memory regions only.
- **Sharing:** Mechanisms are provided to selectively enable partitions to share certain hardware resources, such as memory.
- **Virtualization:** This provides support for mechanisms that enable the sharing of certain devices among partitions, such as the system interrupt controller.
- **Performance:** The hypervisor uses the features of the Freescale QorIQ P4 series processor to provide security and isolation with very low overhead. Guest operating systems take external interrupts directly without hypervisor involvement, providing very low interrupt latency.
- **Ease of migration:** The hypervisor uses a combination of full emulation and para-virtualization to maintain high performance while requiring minimal guest OS changes when migrating code from an e500mc CPU to the embedded hypervisor.

The Freescale embedded hypervisor software is unlike traditional hypervisors in that it does not provide scheduling services that enable multiple operating systems to run on a single physical CPU at the same time. The hypervisor exclusively addresses the problem of partitioning and isolation.

The hypervisor provides services to guest software through two mechanisms: emulation and a special system call referred to as a hypercall (or hcall).

Emulation is where normal CPU instructions and registers are used to provide a service where the guest software is unaware that it is running under a hypervisor. For example, the execution of a *tlbwe* instruction traps to the hypervisor, since it is a 'hypervisor privileged' instruction. The hypervisor then performs the *tlbwe* on behalf of the guest and then returns to the guest, which is unaware of the service that was performed on its behalf.

Hypercalls provide a mechanism by which the hypervisor presents an API to guest software for various services.

3.1 Virtual CPU

Guest software running under the control of the embedded hypervisor software-created partition sees a set of Power Architecture instructions and registers that is like an e500mc CPU but with some resources removed. This is referred to as the virtual CPU. Specifically, the virtualization extensions in the e500mc are removed and not accessible to guest software. Normal, supervisor-level software targeting the e500mc core can, in most cases, be run unmodified under the hypervisor.

Standard e500mc CPU features can be used without changing guest software, including supervisor-privileged instructions and registers, exception and interrupt handling, CPU timer facilities and the memory management unit. However, in order to provide the same programming model between the guest and the CPU for supervisor-level code, the hypervisor must emulate some features of the bare-metal architecture, including the interrupt controller and debug functions.

3.1.1 Debug

The embedded hypervisor software provides two options for debugging guest software:

- In one mode the virtual CPU makes the complete set of e500mc debug resources and interrupts available to guest software. This enables a guest debug agent or guest-based debugger to use e500mc debug facilities without change.
- In another mode the hypervisor software takes control of the e500mc debug resources. A debug agent that is part of the hypervisor provides an interface for a host debugger, such as gdb, to debug guest software using a byte-channel as the communications channel.

3.2 I/O and Interrupts

Typically, I/O devices will be assigned directly to and owned by a partition. This means the I/O device is that partition's private resource and is not shared. This enables an operating system to directly access an I/O device with a native device driver with little or no overhead.

The embedded hypervisor software configures the QorIQ communications processor so that external interrupts are received directly by partitions and can be handled by guest software with no added latency introduced by the hypervisor. The QorIQ P4 series processor provides an advanced feature called an External Proxy Register (EPR) designed to enable guest interrupt handlers to read the interrupt vector for an external interrupt without requiring an access to the interrupt controller.

3.3 Partition Management Services

The Freescale embedded hypervisor software provides mechanisms that enable one partition to manage other partitions. A manager partition has the capability to start, stop and restart other partitions using hcalls.

For managed partitions that are 'stopped,' a manager partition can copy data to and from the managed partition, thus enabling the manager to load OS images, etc., into the managed partition.

A manager partition may receive interrupts related to managed partitions:

- Notification on managed partition state change to the running or stopped state
- Notification on managed partition watchdog expiration
- Notification on managed partition restart request

3.4 Byte-Channel Services

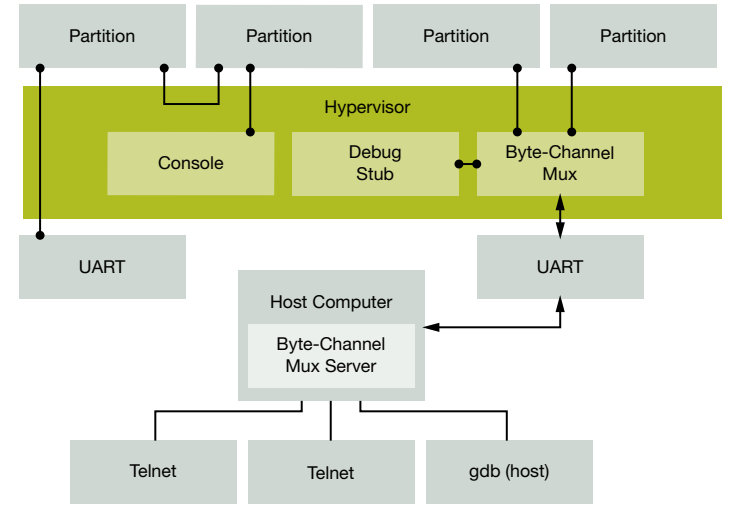
The hypervisor provides an hcall-based service called a byte-channel that provides an interrupt-driven character-based I/O channel. The functionality is similar to a UART. Byte-channels provide a mechanism for each partition to have a console.

Each byte-channel has an endpoint, and a flexible set of endpoints are supported:

- A byte-channel-to-UART multiplexer
- A physical UART on the QorIQ P4 series processor
- Another byte-channel endpoint
- A hypervisor debug stub
- The hypervisor console

The byte-channel to UART multiplexer provides the capability to multiplex multiple byte-channel streams over a physical UART to a host system. The host system runs a mux server that de-multiplexes the streams and makes them available through network ports.

Figure 3.2: Freescale Embedded Hypervisor Software Byte Channels



3.5 Other Hypervisor Services

The hypervisor also provides the following hcall-based services:

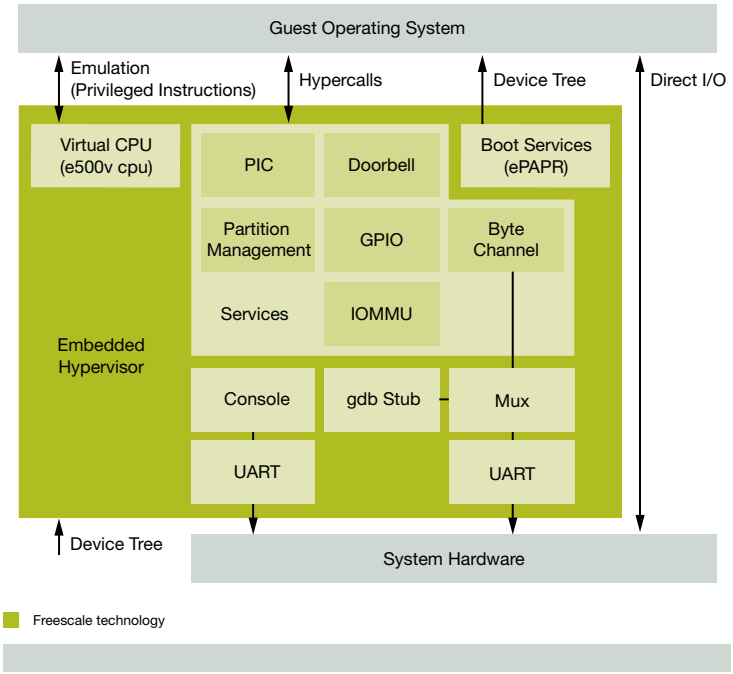
- Interrupt controller services for processing and managing hardware interrupt services
- Inter-partition doorbell services that enable one partition to interrupt another partition
- GPIO services that enable the partitioning and assignment of individual general purpose I/O pins to partitions
- Power management services that enable partitions to access power saving and management capabilities of the QorIQ communications processor

3.6 ePAPR Boot Architecture

The embedded hypervisor adopts the power.org ePAPR (Embedded Power Architecture Platform Requirements) architecture. This architecture defines two aspects of how operating systems are booted: device trees and multi-CPU boot.

A device tree is a data structure passed to a guest operating system at boot that defines the physical and virtual resources that make up the partition. The device tree is how an operating system discovers what resources are available to it. The multi-CPU boot architecture defines mechanisms on how secondary CPUs are released.

Figure 3.3: Freescale Embedded Hypervisor Software Architecture



4 Conclusion

The QorIQ P4 series multicore processor is extremely flexible and can be configured to meet many system application needs. Freescale's innovative CoreNet™ fabric technology is a key design component of the QorIQ processor platform. It enables highly scalable on-chip connectivity by allowing concurrent traffic to enter and exit the system from any point within the fabric rather than through a single point, eliminating bus contention, bottlenecks and latency issues associated with scaling shared bus/shared memory architectures that are common in other multicore approaches.

Leveraging the advantages of the CoreNet fabric and a ground-breaking three-tiered cache hierarchy, the advanced virtualization capabilities of the embedded hypervisor enables the e500mc cores to be combined as a fully symmetric multi-processing system-on-chip, or they can be operated with varying degrees of independence to perform asymmetric multiprocessing (AMP).

The advanced virtualization technology brings a new level of hardware partitioning through the embedded hypervisor that allows system developers to ensure software running on any CPU only accesses the resources that it is explicitly authorized to access. The embedded hypervisor enables safe and autonomous operation of multiple individual operating systems, allowing them to share system resources, including processor cores, memory and other on-chip functions.

The ability of the cores to run different operating systems provides the user with significant flexibility in partitioning control and datapath and applications processing. It also simplifies the consolidation of functions onto a single device where previously they were spread across multiple discrete processors.

How to Reach Us:

Home Page:
www.freescale.com

QorIQ Platform Information:
www.freescale.com/QorIQ

Power Architecture Information:
www.freescale.com/PowerArchitecture

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

Document Number: EMHYPQIQTP4CPWP
REV 1

freescale semiconductor