IMPLEMENTING PCI EXPRESS
BRIDGING SOLUTIONS IN AN FPGA

A Lattice Semiconductor White Paper

July 2010

Lattice Semiconductor
5555 Northeast Moore Ct.
Hillsboro, Oregon 97124 USA
Telephone: (503) 268-8000
www.latticesemi.com

Introduction

Like its predecessor, the Peripheral Component Interconnect (PCI), PCI Express is becoming a ubiquitous system interface. Unlike PCI, PCI Express adopts a Serializer/Deserializer (SERDES) interface to provide users with the scalability required for future applications. As system bandwidths increase, more applications are moving to SERDES-based interfaces like PCI Express. In the past, ASICs or ASSPs typically have been used to implement next generation interface solutions. ASICs and ASSPs were popular choices because they provided a low cost, low power design solution. However, several new FPGAs families now offer very attractive options to designers. FPGAs provide an extremely flexible platform without the long lead times and large NREs typically associated with ASICs or the inflexibility of ASSPs. Newer generation FPGAs with embedded SERDES, like the LatticeECP2M and the LatticeECP3 devices, offer designers an extremely rich, high value programmable architecture, while also offering a low cost, low power solution for serial interfaces. The same FPGAs can be used to support a variety of serial protocols like PCI Express, Gigabit Ethernet, SGMII, XAUI, Serial RapidIO, and others, providing a single FPGA platform for multiple designs.

PCI Express is also becoming the interface of choice for control plane applications, replacing older parallel interfaces like PCI. Newer generation devices use one or more PCI Express links. In a majority of devices, the PCI Express core is implemented as a PCI Express endpoint. Designers often need to connect these devices to previous generation devices that have a parallel bus (e.g., microprocessors with parallel bus interfaces). Using a low cost, low power FPGA to bridge between PCI Express and a parallel interface provides designers the flexibility to solve this problem without exceeding their system cost and power budgets.

PCI Express Design Challenges

As designers migrate from PCI to PCI Express, the intricacies of the protocol, coupled with the complexities involved with SERDES-based designs, pose significant challenges. Fortunately, FPGAs, along with a full-featured PCI Express IP core, reference designs, hardware evaluation boards and associated demos, help smooth out the otherwise steep learning curve PCI Express designers face. Indeed, FPGAs are an ideal platform for PCI Express-based applications. Because they are programmable, FPGAs allow designers the flexibility to resolve design issues late in the design cycle with rapid turnaround times. Designers can also readily change or add features as individual design requirements evolve. FPGA designs also allow designers to make updates to accommodate changes to the specifications, enabling them to "future-proof" their designs against obsolescence. The programmable platform also allows designers to use the same FPGA to implement interface solutions that connect to a broad variety of other PCI Express chipsets: endpoints, root complex or switches. The designer can also integrate other functions required by the system in the FPGA, reducing the number of components on the board and further reducing the total cost of the system.

PCI Express Solutions

FPGAs provide an extremely flexible programmable platform for system designs. A comprehensive solution package that includes Intellectual Property cores, hardware platforms, demo designs, drivers and software enables designers to shrink their development cycles while reducing the complexity of the design. One common design requirement is for a PCI Express solution to bridge between PCI Express serial interfaces (endpoint devices) and legacy parallel bus interfaces, as shown in Figure 1. An FPGA with a PCI Express root complex IP core provides designers with the basic building blocks needed to implement such a solution. Alternatively, ASSPs and ASICs are also available that can implement this function. However, unlike FPGAs, these devices can implement only a fixed configuration that cannot be changed to accommodate the various parallel bus interfaces available. A programmable FPGA

Implementing PCI Express Bridging Solutions in an FPGA

A Lattice Semiconductor White Paper

platform, on the other hand, enables designers to make specific changes in their design to implement the specific bridge function that matches the interface available on their particular board. Designers also have the flexibility of implementing multiple bridges or different configurations of bridges in a single FPGA, thus reducing the total components on the board. An FPGA coupled with the PCI Express root complex IP core can enable various other bridging solutions as required by a design.
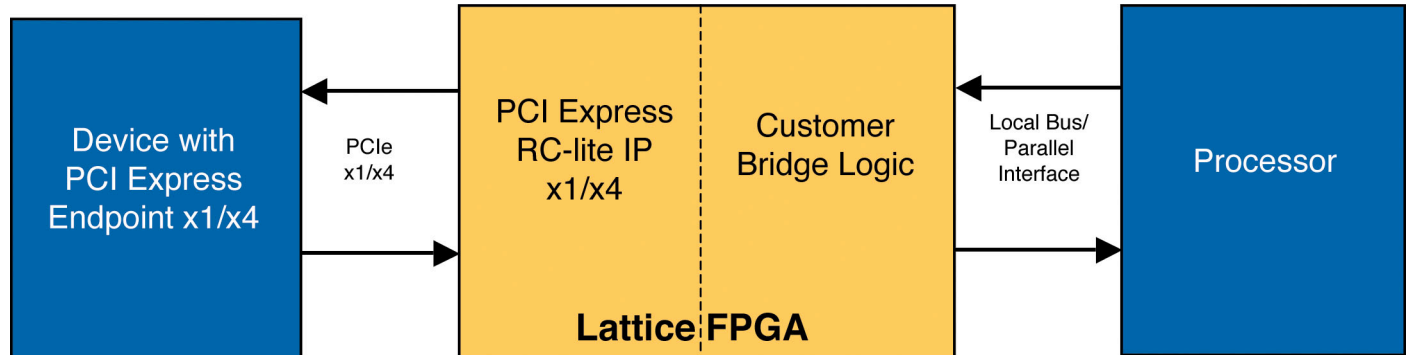
Figure 1 - PCI Express Bridging Solution

PCI Express Root Complex

A PCI Express endpoint operates as an upstream device, and cannot communicate downstream. This function can be performed by a root complex device, but a full-featured root complex implementation is quite expensive in terms of FPGA gates used. Instead, a "lightweight" root complex core with a sub-set of the transaction layer functionality is adequate for implementing most bridging functions. As shown in Figure 1, the bridge is comprised of two basic building blocks. The first block is the PCI Express root complex (or Root Complex-lite) IP core, which interfaces with the PCI Express endpoint device. The second block is the bridge logic that interfaces to the local bus/parallel interface. Since this implementation is in a programmable FPGA, the designer has a lot of flexibility to customize the design based on specific interface

needs. Other functionality can also be integrated into the same FPGA, eliminating other components on the board and reducing overall BOM costs.

PCI Express Root Complex IP Support

PCI Express is a complex protocol. Providing fully functional, fully validated PCI Express Intellectual Property cores significantly reduces the design complexity for the designer. For example, Lattice's PCI Express Root Complex Lite (RC-lite) core implements an x1 or x4 root complex function primarily for use in PCI Express bridging applications. As shown in Figure 2, all the PCI Express layers are implemented as a combination of embedded ASIC blocks and the PCI Express RC-lite soft intellectual property (IP) core, implemented in the FPGA. The various blocks include the electrical SERDES interface, the physical layer, the data link layer and a minimum transaction layer to support the protocol stacks required to implement a PCI express root complex function. This "lighter" IP is optimized for use in simple bridging applications between a PCI Express endpoint interface and a parallel local bus interface.
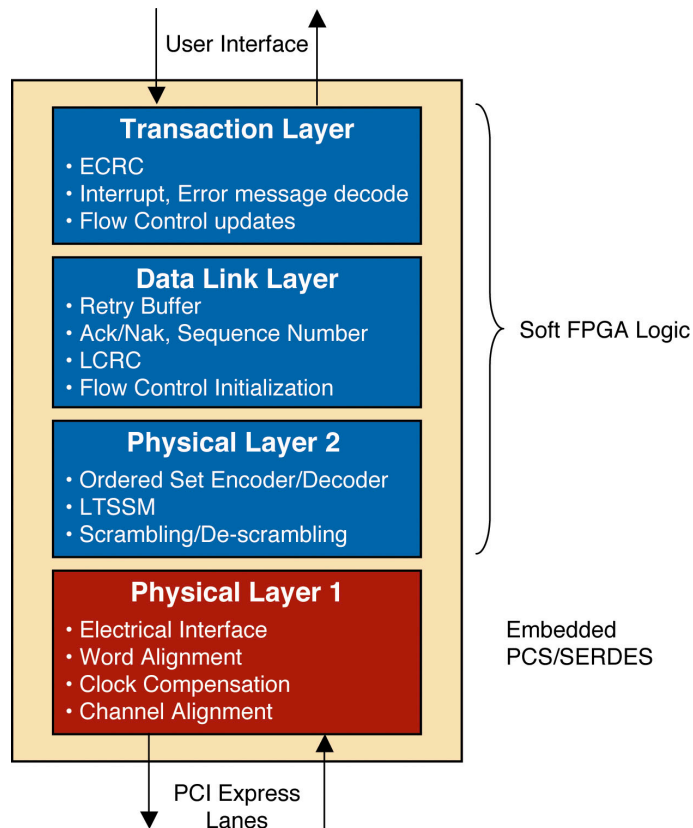
Figure 2 – PCI Express RC-lite IP Protocol Stack

The PCI Express RC-lite IP implemented in a LatticeECP2M or LatticeECP3 FPGA enables low cost, low power PCI Express bridging applications while providing designers the flexibility to customize the bridge interface. Additionally, PCI Express hardware evaluation boards and a variety of reference designs, demos and software drivers help designers kick-start their PCI Express designs and reduce time-to-market. Lattice also provides a hardware evaluation board for designers to test the RC-lite IP solution. Designers will be able to complete interoperability and verify the system level functionality of these solutions prior to actual system level deployment, saving the time and cost normally associated with post-design debug and performance enhancements.

Conclusion

PCI Express designs pose significant challenges to designers. The requirements for the interface are varied, depending on whether the PCI Express device has to connect to another end point, root complex or switch. Furthermore, in a number of cases the requirement is to connect between a PCI Express endpoint device and another device with a parallel bus interface. Designers can implement these functions in a low cost, low power FPGA platform while still having all the benefits of a flexible programmable architecture. For example, implementing a PCI Express root complex IP function in an FPGA provides an ideal platform to implement these bridging functions. *###*