100Gbps PCI-Express Readout for the LHCb Upgrade

Paolo Durante, Niko Neufeld, Rainer Schwemmer, Umberto Marconi, Gabriele Balbi, and Ignazio Lax

Abstract—We present a new data acquisition system under development for the next upgrade of the LHCb experiment at CERN. We focus in particular on the design of a new generation of readout boards, the PCIe40, and on the viability of PCI-express as an interconnect technology for high speed readout. We show throughput measurements across the PCI-express bus on Altera Stratix 5 devices, using Direct Memory Access. Finally we discuss hardware and software design considerations necessary to achieve a throughput of 100Gbps per readout board.

I. INTRODUCTION

The LHCb experiment is currently undergoing an upgrade in anticipation for higher luminosity operation after LS2. Part of this upgrade involves a complete overhaul of its data acquisition system in order to allow a “triggerless” readout of the entire experiment at the LHC bunch crossing rate of 40MHz. As a direct consequence of the increase in luminosity and the elimination of the hardware trigger, the new readout system has to be able to accommodate a sustained aggregate throughput of the order of tens of terabits per second.

This requirement is very demanding for all the elements of the readout system, from the sub-detectors, to the long-distance optical links, to the readout boards, to the computer network used to assemble, process and filter physics events as they are received.

This document will focus on the challenges that such performance requirements impose on the design of a family of custom-made readout boards, henceforth collectively referred to as the PCIe40, and in particular on the implementation of an appropriate protocol for communication between a readout board and its associated event builder unit.

II. OVERVIEW OF THE PCIe40

As already mentioned, the PCIe40 will not be a one-size-fits-all design but will encompass a family of board designs each tailored for a particular experiment sub-system, however all boards are designed around a common platform that can be seen in Fig.1.

The FPGA at the core of every board is meant to be a 10-series device by Altera Corp. more specifically a high-end device like the Arria 10 or the upcoming Stratix 10, which offer up to 1150k reconfigurable logic elements. Reconfigurable logic was obviously chosen for its unparalleled performance and flexibility and will allow every sub-detector to customize the data-processing logic according to their needs.

Each board will be installed in a dedicated readout-unit, fundamentally consisting of a server-grade CPU with very fast memory and network interfaces appropriately dimensioned to match the output data rate produced by the readout board.

By taking into consideration the current state of the art and projected advancements in interconnect technology, a performance figure of 100Gbps for the link between the readout-board and the readout-unit seems to result in the most cost-effective readout system [1].

On its front panel, the PCIe40 receives raw physics data from the various sub-detector front-end chips over long-distance optical links using a special-purpose protocol developed at CERN [2]. The number of links is a function of the requirements of the particular sub-system connected to the board and can vary (nominally between 24 and 48) depending on individual link speed and on sub-detector occupancy, this allows all sub-detectors to optimize the number of boards to be produced and to exploit the full output bandwidth and logic resources available, as those are meant to be fixed across the entire family of boards.

The second external interface on the PCIe40 consists of a 16-lane PCI-express connector with each lane supporting a signaling speed of 8GT/s, as specified by the PCI-express Gen3 protocol. According to specification, a 16-lane Gen3 link is perfectly adequate to meet our performance target, creating such a link requires however a particular configuration that will be described in the following section.

III. SWITCHING CONFIGURATION

PCI-express is a mature protocol with readily available IP for modern FPGAs like those used in the PCIe40, however Gen3 is a relatively recent revision and modern FPGAs only

Fig. 1. High-level schematic of the PCIe40 board

P. Durante, N. Neufeld, and R. Schwemmer are with CERN, Switzerland
U. Marconi, G. Balbi, and I. Lax are with INFN, Italy
provide hardened IPs for configurations up to 8 lanes rather than 16.

Hardened IPs are however advantageous both in terms of power consumption and logic utilization, and are already available “for free” on the device. A possible solution to exploit this silicon for our application is to logically pair two such x8 links and expose them as a single, high-speed, x16 link.

This is possible because PCI-express is a point-to-point, packet-oriented protocol designed to interconnect potentially complex and heterogeneous topologies such as the one shown in Fig.2. Within a topology, dedicated devices called switches can be used to multiplex access to the interconnect fabric between multiple endpoints.

In our case, using a discrete PCI-express switch, the PCIe40 would be able to bond two x8 interfaces from the FPGA onto a x16 slot in the readout-unit. In order to evaluate the performance characteristics of such a solution, a testbed was set up using an ASUS GTX 690 board. This board contains two GPGPUs behind a PEX8747 PCI-e switch from PLX Technologies, each GPGPU was programmed with a data generator emulating a simple DAQ flow and the throughput seen downstream from the switch was measured to consistently exceed 110Gbps thus satisfying our requirements [3].

In a switched scenario, the next open question is whether a single 8-lane link implemented using Altera’s PCI-express HardIP is able to absorb at least half of the readout bandwidth, answering this question is the focus of the next section.

IV. DMA ARCHITECTURE

Direct Memory Access, or DMA, is the most efficient mechanism to transfer data from an external peripheral like the PCIe40 to the main memory of a computer like the aforementioned readout-unit, this is because DMA does not need to preempt the CPU at any point of the memory transfer. In order to evaluate the performance of the Altera PCI-express interface, a specialized DMA controller was implemented.

In order to stress test the DMA controller, a simple data generator was implemented. Data is generated as a 256bit wide stream of events connected to the DMA controller, the data generator works at a frequency of 250Mhz and is therefore able to achieve a maximum data rate of 64Gbps.

In order to guarantee data integrity and prevent data loss (which is particularly critical in a data acquisition scenario) the DMA controller has to carefully synchronize memory access at two important boundaries:

- the small internal buffer on the FPGA, where data from the input stream is written at the same time as it is fetched by the DMA engine
- the larger event buffer in the main memory of the readout-unit, which is both written by the DMA engine and read by the event building application

In the first case this is accomplished by maintaining a dedicated state machine for each section of the internal buffer, in the second by keeping a pair of internal memory guard registers, for read and write access respectively, that both the FPGA and the event building software have to update in order to coordinate access to the event buffer.

In order to maximize throughput in such a concurrent read/write scenario, access to the internal buffer must be highly granular, this allows for example one segment of the buffer to be written with new data, while another segment is pinned waiting to be transferred, while yet another segment that has already been transmitted is awaiting for acknowledgment by the PCI-e root complex.

This results in an internal bookkeeping system where to each segment of the internal buffer corresponds a state variable which, in response the behavior of the input event stream and the DMA engine, can assume one of the following values:

- FREE the segment is available and can be overwritten with new data
- FILL the segment is being filled by new data from the input stream
- SEND the segment is ready for transmission to the DMA engine
- WACK the DMA request has been issued but its completion has not yet been acknowledged

An additional benefit of this fine granularity is that it allows the internal buffer to be extremely small: thanks to the small packet size used by the PCI-express Data Link Layer, each buffer segment can be very small and it is stored in FPGA memory only for as long as it is in transit across the PCI-express link without locking the entire buffer or a substantial portion thereof. This exploits the memory and the memory interface on the readout-unit to the fullest extent and simplifies the design of the readout board considerably by not requiring the integration of an external memory interface on the board.

Since DMA transfers do not preempt the host CPU there still needs to be a mechanism to notify the readout-unit of the availability of new data, two obvious options are active polling by the CPU or the generation of interrupt messages from the FPGA peripheral towards the CPU. The second approach is by far the most efficient and is the one implemented in our solution.

V. DMA PERFORMANCE

The aforementioned DMA controller has been implemented and successfully tested on a setup consisting of an Altera
The implemented DMA controller exhibits very consistent performance over several days of operation which is of course highly desirable in a sustained data acquisition scenario. Another positive aspect of our implementation is that it requires only 64KiB of on-chip memory for DMA buffering at the cost of only 5% logic utilization on a Stratix 5 FPGA. The Arria 10 chosen for the PCIe40 almost doubles the available logic elements which would result in even lower logic utilization.

VI. CONTROL & STATUS INTERFACE

An additional factor that must be taken into consideration in order to validate the choice of PCI-express for our purposes, and in particular Altera’s implementation of PCI-express, is the fact that, in addition to DMA transactions transporting physics events from the readout board to the readout unit, the LHCb Experiment Control System also requires access to the board in order to configure and monitor its internal state. This is accomplished by executing one instance of the distributed control system on the readout-unit itself, the control system would then have immediate access to the PCIe40 register space through the same PCI-express link used for data acquisition.

In order to adopt such a solution it is however necessary to prove that this out-of-band control traffic would not disruptively interfere with the data acquisition traffic. In order to test this assumption the DMA firmware was extended with an additional memory mapped interface, independent from the DMA and connected to a set of 32-bit internal FPGA registers. At the same time the software driver was also augmented to provide memory mapped access to this internal register space from the readout-unit. A configurable memory access generator was implemented in order to stimulate different memory access patterns to the FPGA registers, both in the read and write direction.

By stress testing the DMA controller in parallel with this control system emulator, generating a pseudo-random read/write access pattern to the internal FPGA register file at a speed of 1Gbps, we observed a proportional decrease in DMA performance to 54.6Gbps, still able to satisfy our requirements.

It has to be pointed out that more regular access patterns, for example where the read/write memory location is fixed or monotonically increased after each operation by one register offset across the entire address space do seem to cause particular contention on the PCI-express link, to the point where average DMA performance drops to about 49.8Gbps. Understanding this behavior will require deeper knowledge or the PCI-express memory ordering model, this kind of access patterns are however degenerate cases far from the typical behavior of a control system and these results should be considered accordingly.

VII. CONCLUSIONS

We have implemented a high-performance streaming DMA controller for DAQ workloads. The FPGA firmware has been optimized for throughput and resource utilization, the observed throughput is able to satisfy the very demanding requirements of the LHCb experiment upgrade and the resource utilization is kept to a minimum in order to maximize logic resources available for physics data processing. In conjunction with the FPGA firmware, low-level driver software and user libraries were also implemented for integration in the future event building and experiment control system.

REFERENCES

[1] C. LHCb, “LHCb Trigger and Online Upgrade Technical Design Report,” CERN, Geneva, Tech. Rep. CERN-LHCC-2014-016. LHCB-TDR-016, May 2014.
[2] C. Soos, M. Barros Marin, S. Detraz, L. Olanter, C. Sigaud, S. Storey, J. Troska, F. Vasey, and P. Vichoudis, “The Versatile transceiver: Towards production readiness,” J. Instrum., vol. 8, p. C03004, 2013.
[3] N. Neufeld, D. Galli, V. Vagnoni, B. Jost, I. Lax, G. Liu, R. Schemmer, P. Durante, and U. Marconi, “A PCIe Gen3 based readout for the LHCb upgrade,” in 20th International Conference on Computing in High Energy and Nuclear Physics, 2013.