A multi-port 10GbE PCIe NIC featuring UDP offload and GPUDirect capabilities.

2015 J. Phys.: Conf. Ser. 664 092002

(http://iopscience.iop.org/1742-6596/664/9/092002)

Journal of Physics: Conference Series **664** (2015) 092002 doi:10.1088/1742-6596/664/9/092002

A multi-port 10GbE PCIe NIC featuring UDP offload and GPUDirect capabilities.

Roberto Ammendola[1], Andrea Biagioni[2], Ottorino Frezza[2], Gianluca Lamanna[3], Francesca Lo Cicero[2], Alessandro Lonardo[2], Michele Martinelli[2], Pier Stanislao Paolucci[2], Elena Pastorelli[2], Luca Pontisso[4], Davide Rossetti[5], Francesco Simula[2], Marco Sozzi[6 4], Laura Tosoratto[2], Piero Vicini[2]

[1] INFN, Sezione di Roma Tor Vergata, Italy
[2] INFN, Sezione di Roma, Italy
[3] INFN, Laboratori Nazionali di Frascati, Italy
[4] INFN, Sezione di Pisa, Italy
[5] NVIDIA Corp, Santa Clara, California
[6] Università, Pisa, Italy

E-mail: andrea.biagioni@roma1.infn.it

Abstract. NaNet-10 is a four-ports 10GbE PCIe Network Interface Card designed for low-latency real-time operations with GPU systems. To this purpose the design includes an UDP offload module, for fast and clock-cycle deterministic handling of the transport layer protocol, plus a GPUDirect P2P/RDMA engine for low-latency communication with NVIDIA Tesla GPU devices. A dedicated module (Multi-Stream) can optionally process input UDP streams before data is delivered through PCIe DMA to their destination devices, re-organizing data from different streams guaranteeing computational optimization. NaNet-10 is going to be integrated in the NA62 CERN experiment in order to assess the suitability of GPGPU systems as real-time triggers; results and lessons learned while performing this activity will be reported herein.

1. Introduction

The GPGPU paradigm, the employ of graphical processing units for fulfillment of traditional computing tasks, is strongly emerging in several fields of computational physics. GPUs are definitively integrated in the high-level trigger of ALICE at LHC [1] while other HEP experiments as ATLAS [2], CMS [3] and LHCb [4] are evaluating GPU adoption in order to achieve the target computing power to cope with the LHC luminosity increase expected in 2018.

A different approach — based on the exploitation of GPUs also in low-level triggers [5] or in fast control systems [6] of HEP experiments — is recently being investigated; in this latter, the major hurdle to overcome is the hard real-time constraint. Although GPUs show a mostly deterministic behaviour in terms of latency while performing computational tasks, the data streams coming from the experimental apparatus which flow into their local memories represent a source of non-deterministic fluctuations in overall system response.

The NaNet project arises in this context with the goal of designing a low-latency and high-throughput data transport mechanism for real-time systems based on CPU/GPUs. Since

2012, two PCIe Gen2 network interface cards based on Altera FPGA have been developed: NaNet-1 [7] for the NA62 experiment at CERN [8] and NaNet^3 [9] for the KM3NeT-IT underwater neutrino telescope [10]. NaNet-1 offers a GbE channel and three optional proprietary channels (APElink [11]) with bandwidth up to 34 Gbps and implements a direct communication mechanism with NVIDIA Fermi- and Kepler-class GPUs [12] (GPUDirect V2/RDMA). NaNet^3 implements 4 I/O optical channels with deterministic latency and bandwidth up to 2.5 Gbps for the data transport and slow control management of the experiment.

In this paper we present NaNet-10, the third generation of the NaNet NIC family as an upgrade to the NaNet-1 board. In Sec. 3 a brief description of NaNet architecture is given. Current hardware and software integration details are presented in Sec. 4. Preliminary benchmarks are showed in Sec. 6

2. Related Works

In the High Energy Physics field, several experiments have FPGA boards in development for their DAQ network.

In [13] the FEROL card is described — based on Altera Arria II FPGA — which is developed within the CMS experiment and implements the TCP/IP protocol suite over 10 Gbps Ethernet, providing a reliable connection between the front-end readout system and the event-builder network via two 6 Gbps SlinkXpress interfaces.

The FELIX project [14, 15] aims at providing a high-throughput interface to the upgraded version of ATLAS front-end electronics. A baseline implementation of a FELIX node consists of a PCIe Gen3 card based on Xilinx Virtex-7 with high-density optical interfaces towards at least 24 GBT links and a PCIe dual NIC implementing commodity network technology (56 Gbps InfiniBand).

C-RORC [16] (Common Read-Out Receiver Card), developed by ALICE, implements a PCIe Gen 2 x8 network adapter based on Xilinx Virtex-6 and interfaces to 12 optical links via three QSFP transceivers capable of up to 6.6 Gbps each.

The baseline design for the LHCb readout is an ATCA board requiring dedicated crates. The readout boards (TELL40 [17]) consist of an ATCA-compliant carrier board hosting up to four AMC40-card plugged onto it. Each AMC40-card is equipped with a single, high-end FPGA (ALTERA Stratix V or newer) providing 24 GBT-link input and 12 LAN-link output (10GbE and UDP network protocol).

Keeping with the idea of an FPGA supporting the DAQ network in the context of a physics experiment of a completely different nature is PRANA [18], a pathfinder for a reliable, scalable and energy-efficient, GPU-based real-time controller for adaptive optics able to target extremely large telescopes such as the future E-ELT.

3. NaNet Architecture Overview

NaNet is a modular design of a low-latency PCIe RDMA NIC supporting different network link technologies and partitioned into 4 main modules: *I/O Interface*, *Router*, *Network Interface* and *PCIe Core* (see Fig. 1).

The *Network Interface*, *Router* and *PCIe Core* are inherited from APEnet+ [19], a parallel development line dedicated to the HPC environment, preserving the key features and the achieved reliability degree.

The *PCIe Core* module is built upon a high-end commercial core from PLDA that sports a simplified but efficient back-end interface and multiple DMA engines.

The *Network Interface* guarantees a GPU/CPU memory write bandwidth of 2.8 GB/s benefiting of a 128-entries Translation Look-aside Buffer [20]. The CPU memory read bandwidth has a capability of 3.0 GB/s and the same is true for the Kepler-class GPU exploiting the GPUDirect RDMA features thanks to the implementation of a double DMA channel. Backward

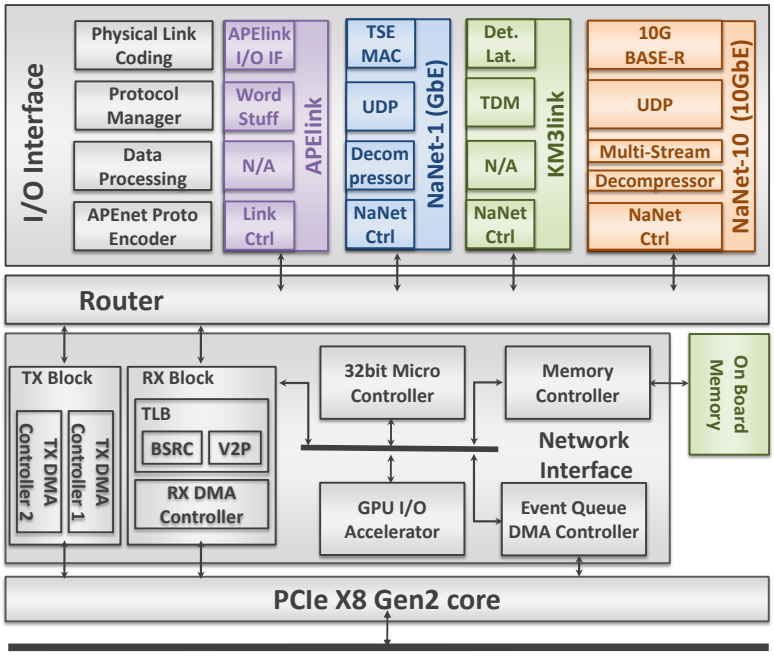

Figure 1. NaNet architecture schematic.

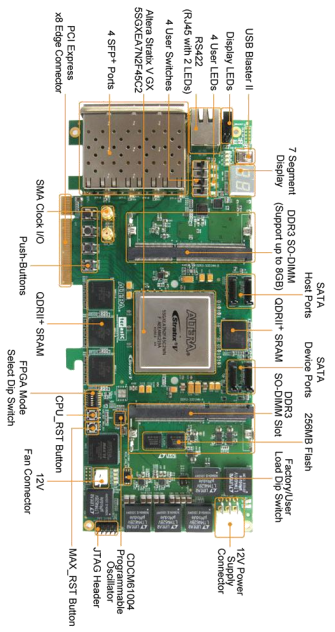

Figure 2. NaNet-10 board.

compatibility with Fermi-class GPUs is guaranteed by the implementation of a GPUDirect V2 custom protocol with 1.5 GB/s as upper limit.

The *Router* is able to sustain up to eight data streams at 2.8 GB/s applying a deterministic dimension-ordered routing policy.

The *I/O Interface* is the discriminating component among the cards in the NaNet family. It is each time re-designed in order to satisfy the requirements of the readout system data transmission protocol optimizing the data movement process for the different experiments. The I/O Interface module performs a 4-stages processing on the data stream: following the OSI Model, the Physical Link Coding stage implements, as the name suggests, the channel physical layer (*e.g.* 1000BASE-T) while the Protocol Manager one handles data/network/transport layers (*e.g.* Time Division Multiplexing or UDP), depending on the kind of channel; the Data Processing stage implements application-dependent reshuffling on data streams (*e.g.* performing de/compression) while the APEnet Protocol Encoder performs protocol adaptation, encapsulating inbound payload data into the APElink packet protocol — used in the inner NaNet logic — and decapsulating outbound APElink packets before re-encapsulating their payload into the output channel transport protocol (*e.g.* UDP).

On table 1 we show a recap of the used FPGA logic resources as measured by the synthesis software.

4. NaNet-10

NaNet-10 is a PCIe Gen2 x8 Network interface card implemented on the Terasic DE5-net board equipped with an Altera Stratix V FPGA and featuring four 10GbE SFP+ ports (see Fig. 2). The network adapter offers hardware support for either direct CPU/GPU memory access and the offloading engine managing the network stack protocol. Data streams coming from the transmission channel are directly routed towards the target memory without any involvement of the CPUs, avoiding OS jitter effects and guaranteeing a stable, low-latency transmission. TTC signals are received from the 2 SMA connectors on board.

Table 1. An overview of NaNet resource consumption.

Project	**FPGA**	**Comb. ALUT**	**Register**	**Memory [MB]**
NaNet-1	EP4SGX230KF40C2	50502 (28%)	52369 (29%)	1.06 (58%)
NaNet^3	5SGXEA7N2F45C2	72364 (26%)	118455 (13%)	1.16 (18%)
NaNet-10	5SGXEA7N2F45C2	83611 (30%)	132194 (14%)	1.17 (18%)
Logic Block		**Comb. ALUT**	**Register**	**Memory [MB]**
PCIe		3902	4449	—
Network Interface		31033	34389	0.825
Router		7954	7957	0.338
10GbE I/O (4 ports)		40503	84935	0.007
GbE I/O (1 port)		5026	7125	0.026
APElink I/O (6 ports)		31349	25487	0.006
KM3link I/O (4 ports)		29540	71057	0.005

4.1. Hardware Overview

In this section we focus on the novel UDP/IP 10GbE I/O data transmission system as the rest of the architecture is widely described in previous literature.

Referring to Fig. 3 and following the design guidelines for the NaNet I/O interface described in Sec. 3, the Physical Link Coding is implemented by two Altera IPs, the 10GBASE-R PHY and the 10 Gbps MAC. The 10GBASE-R PHY IP delivers serialized data to an optical module that drives optical fiber at a line rate of 10.3125 Gbps. PCS and PMA are implemented as hard IP blocks in Stratix V devices, using dedicated FPGA resources. The 10 Gbps MAC supports 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps operating modes with Avalon-Streaming up to 64-bit wide client interface running at 156.25 MHz and MII/GMII/SDR XGMII on the network side.

We developed a custom 10 Gbps UDP/IP Core as a Protocol Manager of the I/O interface, providing full UDP, IPv4 and ARP protocols. It is derived and adapted from the FPGA-proven 1 Gbps UDP/IP open core [21] and provides an AXI-based 64-bit data interface at an operating frequency of 156.25 MHz. Several registers are exposed for UDP header settings (*e.g.* source/destination port and destination IP address) both in the transmit and receive side. IP and MAC address are also fully customizable. The core offers ARP level functionalities, with a 256-entries cache for IP-to-MAC address translation. Underlying ARP communication is automatic when first packet transfer occurs and sender and receiver mutually exchange informations about their own IP and MAC addresses. There is no data buffering internally, allowing zero latency between the Data Processing block and the Physical layer. For this reason packet segmentation and reassembly are not supported.

The Multi-Stream and Decompressor hardware components apply application-dependent modifications to accelerate the GPU computing task. Multi-Stream module analyses the received data stream and separates the packets according to the UDP destination port. The experimental requirements lead to generate four data streams that will be redirected to different GPU memory buffers. A Decompressor stage was added in the I/O interface to reformat events data in a GPU-friendly fashion on the fly.

The NaNet Transmission Control Logic (NaNet TCL) encapsulates the received streams into the APEnet Protocol allowing for reuse of the overall APEnet+ architecture. Several parameters are used to configure the NaNet TCL (*i.e.* packet size, port id, target device) and whatever is needed to fulfill the key task of virtual address generation for the APEnet packets. All the information for the virtual memory management is provided by the on-board micro-controller (base address, buffer size, number of available buffers); more details are in Sec 4.2.

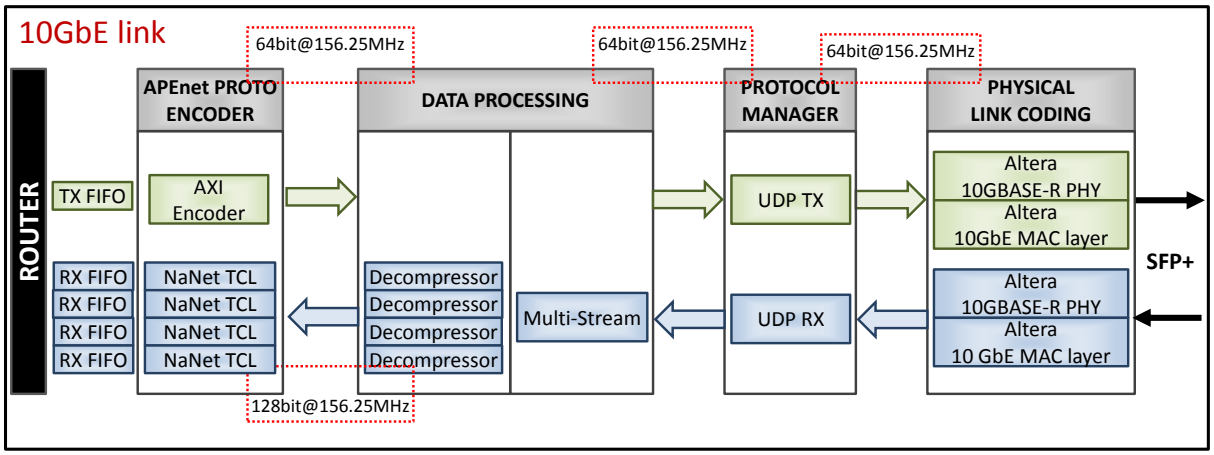

Figure 3. 10GbE I/O data transmission system block diagram.

4.2. Software Overview

NaNet software stack is distributed among the x86 host and the Nios II FPGA-embedded μcontroller. A single process program running on the Nios II is in charge of system initialization and configuration tasks. On the host side a dedicated Linux kernel driver offers its services to an application level library, which provides to the user a series of functions to: open/close the NaNet device, register and de-register circular lists ff persistent data receiving buffers (CLOPs) in GPU and/or host memory, and manage software events generated when a receiving CLOP buffer is full (or when a configurable timeout is reached) and received data are ready to be processed.

4.3. NaNet-10 configuration and data flow

To clarify the NaNet-10 overall functioning we focus on the three main tasks *HW/SW Initialization*, *Data Reception and Processing* and *Data Transmission*, providing a brief description of the hardware/software major operations.

During *HW/SW Initialization* (Fig. 4), the NaNet driver initializes the configuration registers of the board enabling the involved hardware component. CLOP buffers are registered and memory pages are pinned and locked. Assignments of physical memory to virtual memory are communicated to the card through the Nios II. A custom hardware module directly transmits the CLOP settings to the 10GbE links. In this way, links can drive data reception and transmission without need for further software action (network stack protocol is totally hardware-managed).

In the *Data Reception and Processing* stage (Fig. 5), incoming data is reordered according to application-specific demands and encapsulated into a packet-based protocol optimizing PCIe communication. The destination virtual address in CPU or GPU memory for the payload data is generated in hardware and patched into the packet header. The *Router* dispatches data to the receiving port (the same one for both GPU and CPU transactions). The receiving modules operate Virtual-to-Physical translations exploiting a TLB whose entries are registered/deregistered via the Nios II. Data is DMA-written into GPU memory and the completions of the writing process are stored in an event queue; notifications of new events are notified to the driver via interrupt. The driver signals the reception of new frames to the application kicking off the computing process.

The *Data Transmission* (Fig. 6) is handled by the kernel driver. It manages the TX ring where commands for the DMA memory read process and data transmission are queued. The transmitting modules manage the DMA memory read transactions. Data is routed to the

corresponding port, encapsulated into UDP protocol and passed along to the network.

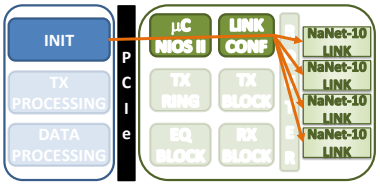

Figure 4. *HW/SW Initialization.*

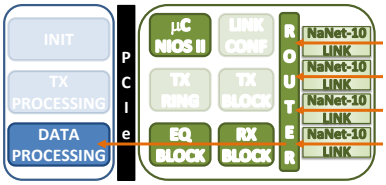

Figure 5. *Data Reception/Processing.*

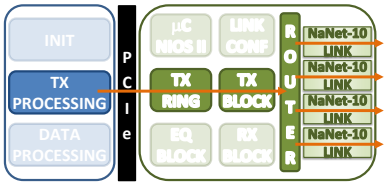

Figure 6. *Data Transmission.*

5. Case Study

NaNet-10 is designed in the context of a pilot project within the NA62 experiment, investigating the feasibility of introducing a GPGPU system as L0 trigger processor (GL0TP). As a first example we studied the possibility to reconstruct rings in the RICH detectors.

Data communication between the TEL62 readout boards and the L0 trigger processor (L0TP) occurs over multiple GbE links using UDP streams. The main requisite for the communication system comes from the request for <1 ms and deterministic response latency of the L0TP: communication latency and its fluctuations are to be kept under control. In the final system, 4 GbE links will be used to extract primitive data from the readout board towards the L0TP satisfying the requisite on bandwidth. Therefore, a single NaNet-10 10GbE channel is enough to manage the entire data stream en-route to the GPU-based L0 trigger.

6. NaNet-10 Performance

NaNet-10 performances are assessed on a SuperMicro Server. The setup comprises a X9DRG-HF dual socket motherboard — Intel C602 Patsburg chipset — populated with Intel Xeon E5-2630 @2.60 GHz CPUs (*i.e.* Ivy Bridge micro-architecture), 64 GB of DDR3 memory and a Kepler-class NVIDIA K40m GPU.

Measurements are performed in "loop-back" configuration closing 2 out of 4 available ports on the Terasic DE5-net board. The outgoing packet payload is generated via a custom hardware module directly feeding the UDP TX. Packet size, number of transmitted packets, delay between packets and UDP protocol destination port are configured setting several NaNet-10 registers.

First we measure the time-of-flight from UDP TX to UDP RX exploiting the SignalTap II Logic Analyzer tool of the Altera Quartus II suite: this is 64 clock cycles @156.25 MHz (409.6 ns).

The receiving hardware path traversal latency is profiled via cycle counter values recorded at different stages during packet processing. The values are patched into the completion payload and stored in the event queue. The adoption of a data transmission custom hardware module ensures that results are not affected by external latency sources (*i.e.* DMA memory reading process). The custom module is always ready to send data exploiting the entire link capability mimicking the detector readout system "worst-case".

A comparison between NaNet-10 and NaNet-1 latency results in the range of interest is shown in Fig. 7. The NaNet-10 NIC experiences sub-microsecond hardware latency moving data to the GPU/CPU memory for buffer sizes up to ∼1kByte.

Focusing on bandwidth, the maximum capability of the data transmission system, 10 Gbps, is already reached for a ∼1kByte buffer size (Fig. 8). NaNet-10 satisfies the capability requirement of the read-out system, four GbE channels from the TEL62, starting from a ∼256 bytes buffer size.

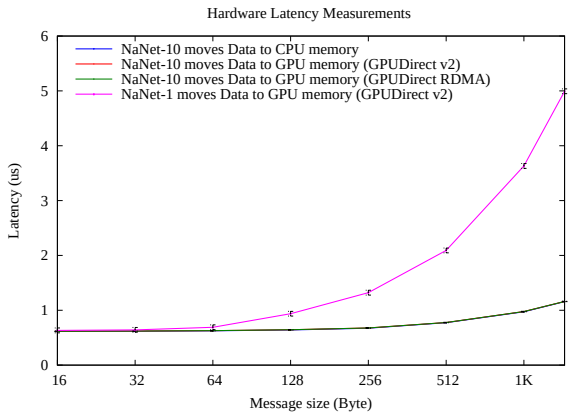

Figure 7. A comparison between NaNet-10 and NaNet-1 hardware latency.

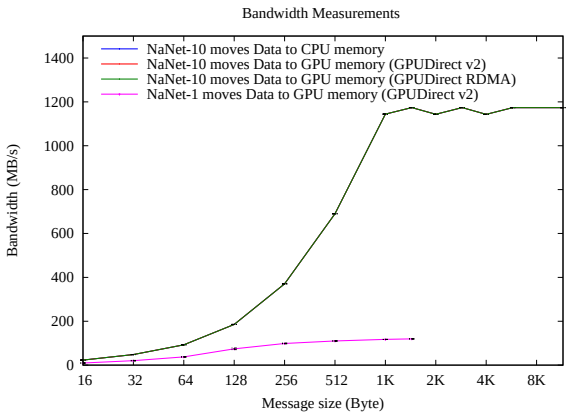

Figure 8. A comparison between NaNet-10 and NaNet-1 bandwidth.

Finally, we notice that performance of moving data from the I/O interface of the NIC to target device memory are the same for both the CPU and the GPU.

7. Conclusion

In this paper we presented the NaNet-10, a low-latency network interface card dedicated to real-time systems offering four 10GbE I/O channels. The Network adapter is built upon the RDMA paradigm between CPU and GPU, offers hardware support for the management of the network protocol and guarantees a sub-microsecond hardware latency when moving incoming data into the target device memory. Some modifications are foreseen in order to optimize transmission of the results from the GPU-based L0 trigger to the upper trigger levels. Implementation of PCIe Gen3 is mandatory in the next future in order to exploit the 40 Gbps aggregate bandwidth of the I/O interface.

The current development status is considered to closely resemble what will be achieved in the final NIC implementation and thus to be a close estimate for its consumption of FPGA resources; referring to table 1, we do not thereby foresee hitting any limit when integrating new functionalities in NaNet-10. On the contrary, exploitation of a more cost-efficient device could be taken into account for final deployment.

Acknowledgments

This work was partially supported by the EU Framework Programme 7 project EURETILE under grant number 247846 and MIUR (Italy) through INFN SUMA project. G. Lamanna, L. Pontisso and M. Sozzi thank the GAP project, partially supported by MIUR under grant RBFR12JF2Z “Futuro in ricerca 2012”.

References

[1] Rohr D, Gorbunov S, Szostak A, Kretz M, Kollegger T, Breitner T and Alt T 2012 *Journal of Physics: Conference Series* **396** 012044 URL `http://stacks.iop.org/1742-6596/396/i=1/a=012044`

[2] Clark P J, Jones C, Emeilyanov D, Rovatsou M, Washbrook A and the ATLAS collaboration 2011 *Journal of Physics: Conference Series* **331** 022031 URL `http://stacks.iop.org/1742-6596/331/i=2/a=022031`

[3] Halyo V, Hunt A, Jindal P, LeGresley P and Lujan P 2013 *Journal of Instrumentation* **8** P10005 URL `http://stacks.iop.org/1748-0221/8/i=10/a=P10005`

[4] Badalov A, Campora D, Collazuol G, Corvo M, Gallorini S, Gianelle A, Golobardes E, Lucchesi D, Lupato A, Neufeld N, Schwemmer R, Sestini L and Vilasis-Cardona X 2014 GPGPU opportunities at the LHCb trigger Tech. Rep. LHCb-PUB-2014-034. CERN-LHCb-PUB-2014-034 CERN Geneva URL `https://cds.cern.ch/record/1698101`

[5] Collazuol G, Lamanna G, Pinzino J and Sozzi M S 2012 *Nuclear Instruments and Methods in Physics Research A* **662** 49–54
[6] Kato S, Aumiller J and Brandt S 2013 *Cyber-Physical Systems (ICCPS), 2013 ACM/IEEE International Conference on* pp 170–178
[7] Ammendola R, Biagioni A, Frezza O, Lamanna G, Lonardo A, Lo Cicero F, Paolucci P S, Pantaleo F, Rossetti D, Simula F, Sozzi M, Tosoratto L and Vicini P 2014 *Journal of Instrumentation* **9** C02023 URL http://stacks.iop.org/1748-0221/9/i=02/a=C02023
[8] Lamanna G 2011 *Journal of Physics: Conference Series* **335** 012071 URL http://stacks.iop.org/1742-6596/335/i=1/a=012071
[9] Lonardo A, Ameli F, Ammendola R, Biagioni A, Ramusino A C, Fiorini M, Frezza O, Lamanna G, Lo Cicero F, Martinelli M, Neri I, Paolucci P, Pastorelli E, Pontisso L, Rossetti D, Simeone F, Simula F, Sozzi M, Tosoratto L and Vicini P 2015 *Journal of Instrumentation* **10** C04011 URL http://stacks.iop.org/1748-0221/10/i=04/a=C04011
[10] Margiotta A 2014 *Journal of Instrumentation* **9** C04020 URL http://stacks.iop.org/1748-0221/9/i=04/a=C04020
[11] Ammendola R, Biagioni A, Frezza O, Lonardo A, Lo Cicero F, Paolucci P, Rossetti D, Simula F, Tosoratto L and Vicini P 2013 *Journal of Instrumentation* **8** C12022 URL http://stacks.iop.org/1748-0221/8/i=12/a=C12022
[12] Ammendola R, Bernaschi M, Biagioni A, Bisson M, Fatica M, Frezza O, Lo Cicero F, Lonardo A, Mastrostefano E, Paolucci P S, Rossetti D, Simula F, Tosoratto L and Vicini P 2013 *Parallel and Distributed Processing Symposium Workshops PhD Forum (IPDPSW), 2013 IEEE 27th International* pp 806–815
[13] Bauer G, Bawej T, Behrens U, Branson J, Chaze O, Cittolin S, Coarasa J A, Darlea G L, Deldicque C, Dobson M, Dupont A, Erhan S, Gigi D, Glege F, Gomez-Ceballos G, Gomez-Reino R, Hartl C, Hegeman J, Holzner A, Masetti L, Meijers F, Meschi E, Mommsen R K, Morovic S, Nunez-Barranco-Fernandez C, O'Dell V, Orsini L, Ozga W, Paus C, Petrucci A, Pieri M, Racz A, Raginel O, Sakulin H, Sani M, Schwick C, Spataru A C, Stieger B, Sumorok K, Veverka J, Wakefield C C and Zejdl P 2013 *Journal of Instrumentation* **8** C12039 URL http://stacks.iop.org/1748-0221/8/i=12/a=C12039
[14] ATLAS FELIX developer Team 2015 A high-throughput network approach for interfacing to front end electronics for atlas upgrades http://indico.cern.ch/event/304944/session/1/contribution/41/material/slides/2.pdf
[15] Bartoldus R S, Bee C M C, Francis D C, Gee N R, George S L R, Hauser R M S, Middleton R R, Pauly T C, Sasaki O K, Strom D O, Vari R R I and Veneziano S R I 2013 Technical Design Report for the Phase-I Upgrade of the ATLAS TDAQ System Tech. Rep. CERN-LHCC-2013-018. ATLAS-TDR-023 CERN Geneva URL http://cds.cern.ch/record/1602235
[16] Borga A, Costa F, Crone G, Engel H, Eschweiler D, Francis D, Green B, Joos M, Kebschull U, Kiss T, Kugel A, Vazquez J P, Soos C, Teixeira-Dias P, Tremblet L, Vyvre P V, Vandelli W, Vermeulen J, Werner P and Wickens F 2015 *Journal of Instrumentation* **10** C02022 URL http://stacks.iop.org/1748-0221/10/i=02/a=C02022
[17] Bellato M, Collazuol G, D'Antone I, Durante P, Galli D, Jost B, Lax I, Liu G, Marconi U, Neufeld N, Schwemmer R and Vagnoni V 2014 *Journal of Physics: Conference Series* **513** 012023 URL http://stacks.iop.org/1742-6596/513/i=1/a=012023
[18] Sevin A, Perret D, Gratadour D, Lain M, Brul J and Le Ruyet B 2014 Enabling technologies for gpu driven adaptive optics real-time control URL http://dx.doi.org/10.1117/12.2055770
[19] Ammendola R, Biagioni A, Frezza O, Lonardo A, Lo Cicero F, Martinelli M, Paolucci P, Pastorelli E, Rossetti D, Simula F, Tosoratto L and Vicini P 2015 *Journal of Instrumentation* **10** C02005 URL http://stacks.iop.org/1748-0221/10/i=02/a=C02005
[20] Ammendola R, Biagioni A, Frezza O, Lo Cicero F, Lonardo A, Paolucci P S, Rossetti D, Simula F, Tosoratto L and Vicini P 2013 *Field-Programmable Technology (FPT), 2013 International Conference on* pp 58–65
[21] 1g eth udp/ip stack http://opencores.org/project,udp_ip_stack