ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THBPA03

THE BACK-END COMPUTER SYSTEM FOR THE MEDIPIX BASED PI-MEGA X-RAY CAMERA

H. D. de Almeida†, M. A. L. Moraes, LNLS, Campinas, Brazil
J. M. Polli, D. P. Magalhães, LNLS, Campinas, Brazil

Abstract

The Brazilian Synchrotron, in partnership with BrPhotonics and CPqD, is designing and developing π-MƎGA ("pi-mega"), a new X-Ray camera using Medipix 3RX chips, with the goal of building very large and fast cameras to supply Sirius' new demands. This work describes the design and testing of the back-end computer system that will receive, process and store images. The back-end system will use RDMA over Ethernet technology and must be able to process data at a rate ranging from 50 Gbps to 100 Gbps per pi-mega element. Multiple pi-mega elements may be combined to produce a large camera. Initial applications include tomographic reconstruction and coherent diffraction imaging techniques.

PI-MEGA OVERVIEW

The new pi-mega X-Ray camera was designed to support the demands of experiments in Sirius. In its current prototype stage, the pi-mega is a 1.536x1.536 (2,4 MPixel, Fig. 1) X-Ray camera acquiring 24 bit images at 1.000 frames per second. Pi-mega cameras are also composable, with many of them forming a matrix to increase the pixel count, with the initial goal being a 2x2 pi-mega array (9,4 MPixel). The camera was designed with the following features:

- Fast readout (up to 2.000 continuous fps @ 12 bit)
- Dark image-free hybrid technology
- 55 µm square pixel size
- Gap minimizing (no dead area)
- Easy maintenance and replacement

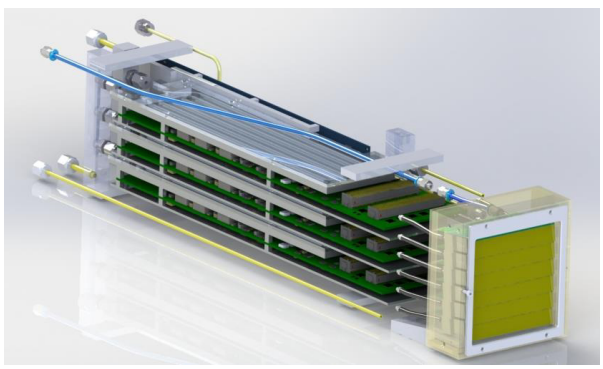

Figure 1: pi-mega design showing the square sensor, the medipix [1] frame boards (green) and the chassis (gray)

The camera is in prototyping stage and it was designed by the LNLS Detectors Group.

† henrique.almeida@lnls.br

SINGLE PI-MEGA BACK-END DESIGN

For a single pi-mega, it's possible to design a back-end computer including a 100 Gbps network card and 1 high-end GPU. In this system, the data flow can be understood as the following (Fig. 2):

1. The raw image data leaves pi-mega and arrives at the network card
2. It then goes to the GPU through some path (for example, it's stored in host memory, then copied to the GPU).
3. The GPU does some processing on the raw data and generates the resulting cooked image data.
4. The resulting data goes back to the network card through some path.
5. The data leaves the network card and goes to storage.

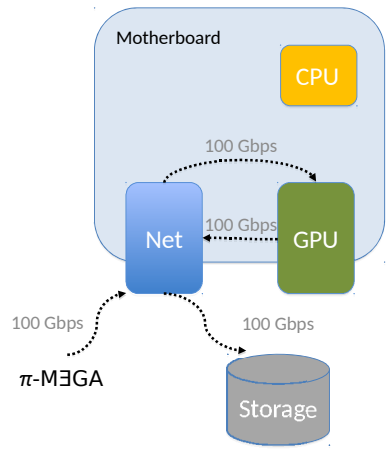

Figure 2: The high level data flow for a single pi-mega.

In the data flow, the incoming raw data may arrive at up to 56,6 Gbps (2,4 MPixel x 24 bit x 1.000 fps).

As a way to make the above data flow as efficient as possible, the RDMA technology (Remote Direct Memory Access [2, 3]) was chosen for both the pi-mega front-end and the network card. RDMA was specifically designed for high performance, low overhead network transfers by offloading the packet processing to hardware. With RDMA, the CPU only deals with the control flow, requesting transfers and receiving completion events from the network card, while the data packet building and processing is delegated to the card [4].

There are different types network cards that support RDMA transferm with different advantages and disadvantages:

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THBPA03

- InfiniBand: All network cards compatible with InfiniBand networks support RDMA [2].
- RoCE v1: RDMA over Converged Ethernet is a hardware accelerated RDMA implementation for Ethernet cards [5]. RoCE v1 implements the InfiniBand network and transport protocols on top of Ethernet. Given Ethernet support, it's easy to implement RoCE in FPGA.
- RoCE v2 [6]: Similar to RoCE v1, but implements the InfiniBand transport protocol on top of UDP/IP, on top of Ethernet. It has the advantage that it's routable through standard IP networks. It's also somewhat easy to implement in hardware.
- iWARP: Ethernet cards with iWARP implement RDMA on top of MPA/TCP (Marker PDU Aligned Framing for TCP, a packet protocol on top of TCP) or SCTP [3]. The main advantage of iWARP cards is to reuse TCP and automatically gain all its benefits, like using a familiar, supported, protocol with reliable flow control, while the disadvantage is that it's harder to implement in hardare.
- OmniPath [7]: Intel's proprietary technology also supports hardware accelerated RDMA.
- SoftROCE: A software based RoCE implementation, useful for testing RDMA without a hardware accelerated Ethernet card. It was merged in linux 4.8 kernel [8].

The RoCE v1 protocol was chosen for the pi-mega for two reasons: first because, unlike the other technologies, it's not expected for the Ethernet standard to become obsolete in the next decades, and second, because it was the simplest protocol to implement in pi-mega front-end.

MULTIPLE PI-MEGA BACK-END

The NVLink interconnect [9] was chosen as the local bus to be used in a multiple pi-mega implementation. The initial target is a camera composed of a 2x2 pi-mega array that will be connected to a machine with 2 PCIe network cards and up to 4 NVLink GPUs. In this setup two pi-megas send data to each of the network cards with the help of a network switch. The maximum allowed frame rate is reduced accordingly to fit data from all 4 pi-megas in the 200 Gbps bandwidth. The main benefit from the NVLink bus is that it's fast enough for GPUs to receive data from both network cards and to exchange data between GPUs. The machine tested, the IBM/OpenPOWER S822LC HPC [10], supports 4 NVIDIA Pascal P100 GPUs with 320 Gbps NVLink connections between the GPUs and also 320 Gbps between GPUs and CPUs (Fig. 3).

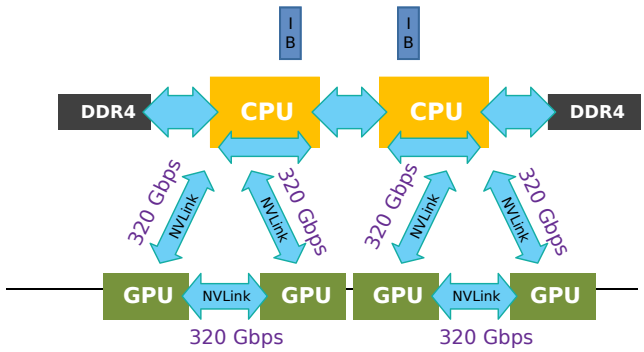

Figure 3: The IBM S822LC HPC bus architecture.

TESTS

Performance tests were conducted remotely at the IBM Benchmark Center at NY, USA. A cluster of S822LC HPC with 4x EDR (100 Gbps) InfiniBand links was available. A benchmark application was developed that was capable of simultaneously saturate the machine data path (Fig. 4) including the network card PCIe bus, the GPU, finishing in the host memory (but excluding the final write to the storage). Multiple runs were executed with pairs of machines, one to simulate a single pi-mega camera generating data and the other representing the back-end receiving and processing the data. In maximum throughput mode, the back-end machines received 256 GiB of data, copying it to the GPU. The GPU calculated a checksum and copied both the data and the checksums to CPU memory. The calculated checksum was validated with a simple, but slow, CPU-only program to make sure there were no software bugs in GPU code. The final throughput achieved was 78 Gbps, which more than what's required by a single pi-mega.

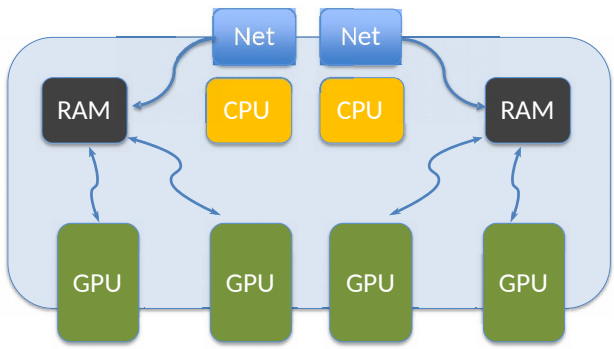

Figure 4: The data flow in the NVLink system.

The benchmark program, called rdmacp, which also provides a general purpose RDMA library, was made available online [11].

NEXT STEPS

The storage at IBM center was tested with an isolated benchmark program only and it needs to be integrated to rdmacp. The RDMA library must also be expanded to support multiple pi-megas, so that a more realistic bandwidth test, where the pi-megas compete for the network bandwidth, can be done.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THBPA03

REFERENCES

[1] R. Ballabriga *et al.*, "The Medipix3RX: a high resolution, zero dead-time pixel detector readout chip allowing spectroscopic imaging", Feb. 2013

[2] *InfiniBand Architecture Specification Volume 1 Release 1.3*, InfiniBand Trade Association, Mar. 2015, pp. 97-101.

[3] *A Remote Direct Memory Access Protocol Specification,* Network Working Group IETF, Oct. 2007, pp. 4-6.

[4] *RDMA Aware Networks Programming User Manual, Rev 1.7,* Mellanox Technologies, May 2015.

[5] *Supplement to InfiniBand Architecture Specification Volume 1 Release 1.2.1, Annex A16: RDMA over Converged Ethernet (RoCE),* InfiniBand Trade Association, Apr. 2010.

[6] *Supplement to InfiniBand Architecture Specification Volume 1 Release 1.2.1, Annex A17: RoCEv2,* InfiniBand Trade Association, Sep. 2014.

[7] M. S. Birrittella *et al.*, "Intel Omni-Path Architecture Technology Overview", Aug. 2015.

[8] *Linux 4.8,* Linux Kernel Newbies, Oct. 2016; https://kernelnewbies.org/Linux_4.8

[9] *What is NVLink?,* NVIDIA Official Blog, Nov. 2014; https://blogs.nvidia.com/blog/2014/11/14/what-is-nvlink/

[10] A. B. Caldeira, V. Haug, and S. Vetter, *IBM Power System S822LC for High Performance Computing Introduction and Technical Overview,* Oct. 2016

[11] *RDMA copy sample file transfer application,* LNLS SOL Group; https://github.com/lnls-sol/rdmacp/