

UFDAC (WP5) Interface Descriptions

Collection of application relevant data by all UFDAC partners

Michael Bussmann (HZDR), Patrick Gessler (XFEL)

March 10, 2016

Preamble

In the following we document the “Definition of data structures and interfaces for software for processing, transfer, and injection of data” in the context of the applications of each participant in WP5.

This is done by putting the data structures and interfaces that are necessary to define in order to transfer knowledge and best practices between WP5 partners in the context of the existing data acquisition hardware and requirements.

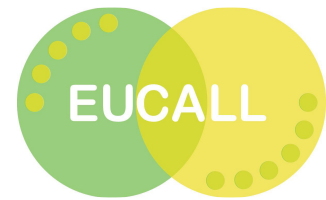
We have thus decided to create a list for each application of each participant that defines not just the data structures that are defined within every application but also additional information like the rate at which data is produced and how it is transported and which hardware is currently used or is foreseen to process the data.

Moreover, data injection at different stages of the analysis chain, data transfer from experiment to data analysis and data processing like filtering, selection, reduction, compression, classification, etc. is detailed in order to have a clear overview of the development status at each partner.

Thus, for each application the following criteria were collected and summarized:

| | |
|-------------------------|--|
| Application: | Short description of the intended scientific application |
| Setup Details: | Details on the scientific setup required to execute the application |
| Sensors/Detectors: | Sensors and detectors used in acquiring data |
| Test requirements: | Any requirements on testing the hardware/software |
| DAQ: | Short review of the data acquisition chain |
| Data-processing el.: | Summary of all hardware used for data processing |
| Interfaces for data...: | Describes the hardware interfaces used for data transfer |
| Protocols: | Describes the corresponding software protocol for data transfer |
| Input data formats: | Describes the data formats coming from the data acquisition |
| Output data formats: | Describes the data formats coming out of the analysis |
| Programming lang.: | Describes the languages in which data acquisition/analysis are written |
| Software: | Describes all software used for data acquisition/analysis |
| Algorithmic details: | Describes special/central features of the algorithms used |
| Additional Res.: | Additional information publically available for reference |
| Aims within UFDAC: | Shortly state the plan for the application within UFDAC |





As becomes clear from the lists below there are two main categories of data, image data and digitizer data as originally identified in the EUCALL proposal.

However, the list reveals a strong diversity when comparing the different applications at the partner facilities in details.

This comes first from the state of development at which each application at an UFDAC partner facility's is in. While some are still in development and some information queried in the list found below is not yet available and will have to be determined, other applications are quite mature and thus more information is already available. Still, even the mature applications face challenges as they are now planned to be scaled for higher data rates and more complex scientific setups.

Bringing these different applications closer together and sharing best practices and common strategies to master the challenges of fast data transfer, injection and analysis is the central aim for UFDAC.

Thus, this report now aims to reveal common structures in the different applications that can be addressed within UFDAC.

Sensors/detectors + DAQ

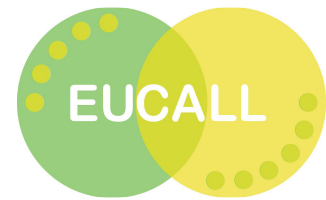
Sensors and detectors are as expected quite diverse and specific for the application. They can be roughly divide in two groups: 1D digitizer data, usually streaming data at a high rate and with multiple channels or multidimensional 'pixel' detectors, usually consisting of a 2D pixel matrix (e.g. MCP, CCDs) and additional information per pixel. These determine the data structure for later analysis. Multi-dimensional image data requires different algorithms for subsequent analysis than 1D data streams. Thus, within UFDAC both groups are treated separately, but will exchange information e.g. on fast Fourier transforms (FFTs).

Data processing element + Programming languages + Software

Here, a great opportunity for common development is clearly visible. For fast analysis, the choice of either FPGA-based or GPU-based solutions or a combination of both is clearly favored by all partners. For the annual EUCALL meeting in 2016 it is thus planned to have a dedicated common session on FPGA/GPU usage for data analysis. Here, we identify the common challenges to be direct memory access (DMA) for fast injection at minimum latency and high bandwidth, data handling for streaming data acquisition and the scalability of the DAQ chain and subsequent analysis to adopt to higher data rates and more data sources/channels.

Consequently, many partners have chosen to use software languages for programming GPUs and FPGAs, e.g. CUDA and VHDL. These choices will influence the development of common software tools and thus will be addressed as one of the next steps within UFDAC. It should be made clear that the choice of programming language or model is not connected to





defining common interfaces, as there is the possibility to have a shared Python/C++/C development.

Interfaces for data processing + Protocols

As in the case of data processing elements, there exists a great common consensus to use solutions based either on Ethernet or Infiniband when transferring data to the data analysis. Within the DAQ chain solutions are more diverse and usually defined by the hardware used. Here, solutions based on MicroTCA, PCIe dominate, yet the variety is big enough that a common standard looks unlikely to emerge within UFDAC. Yet, common solutions to (remote) direct memory access could be envisioned but have yet to be agreed on.

For protocols TCP/IP or UDP dominate the choice for network transfer of data with MPI-based solutions existing on the high-performance computing side of the data analysis. With the exception of XFEL all partners within UFDAC seem to use these standard protocols, but fortunately the XFEL protocol is compatible with these solutions.

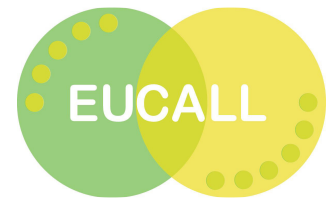
Input data formats + Output data formats

Here, a great opportunity for common development is clearly visible. Although input data formats vary greatly between applications and are often hardware-defined raw data formats that cannot be changed, for fast analysis, the output data is most often in one way or the other HDF5-compliant. HDF5-compliant software is widely available and the format in its parallel form (PHDF5) is a de facto standard for high performance computing file I/O.

Still, internal output data formats vary among the UFDAC applications. It will thus be foreseen to investigate a common meta data format that defines a minimum common set of meta information on image data and digitizer data that then can be used for subsequent data analysis and visualization purposes. Such a meta data format would be defined as an open standard that all participants can agree on and make their special data formats compliant to.

Meta data information on the scientific data such as image size and bit depth for images as well as meta data from the DAQ such as information on integration time, time stamps, etc. will be in the focus of UFDAC. The DAQ meta data will be of use for subsequent data quality management and optimization of data transport, reduction, transfer, storage and compression. For example the dynamic range of a detector and its dark count rate can serve as an information to determine the maximum information content of an image and thus the storage for the image.





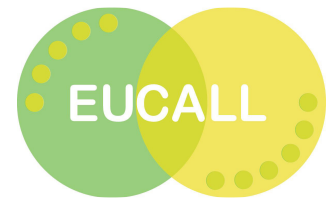
Preliminary conclusion from the definition of data structures and interfaces

For Milestone 5.1 we can now define a set of common working points to address within UFDAC

- All data formats fall into one of the two categories of image data or digitizer data.
- Fast data processing will be done via FPGA- or GPU-based solutions or a mix thereof.
- For this, all applications of all partners are ready to share solutions that are based on C/C++/Python/VHDL.
- For fast data injection, (remote) direct memory access is of great interest.
- Yet, the scalability of the DAQ chain has to be addressed as it is necessary for future data rates to maximize both bandwidth and scalability.
- A starting point for scalable solutions will be data injection and subsequent analysis based on standard network hardware and protocols rather than application-specific hardware and protocols. Solutions favoring TCP/IP, UDP and Infiniband can be shared among the partners
- Yet, PCIe and MicroTCA will be considered when developing solutions for (remote) direct memory access
- (Parallel) HDF5 is the de-facto standard for file input/output used among all partners and can form the basis for defining common meta data formats that describe a minimum set of common scientific data and DAQ data for use by common software solutions.

Accordingly, these working points will be central to the discussion within the WP5 sessions and a common session on GPU/FPGA programming during the 2016 annual EUCALL meeting. It is also foreseen to hold a WP5 workshop in the fourth quarter of 2016 to foster the exchange between the partners.

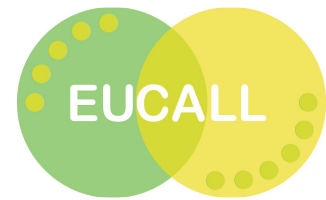




Partner: ELI-ALPS

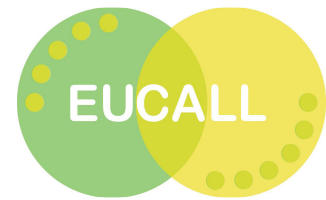
| Item | Description | Detail |
|-------------------|---|---|
| Institution | ELI | Extreme Light Infrastructure |
| Application | Processing Digitizer data Spectroscopy MCP String-arrays | |
| Setup Details | DAQ at secondary sources and end-stations. Data from experimental areas will be send over Ethernet connection to HPC front-end for on-line data analysis. Off-line data will be available for remote analysis user interface. | At the HPC front-end data should be analyzed and data reduction should be performed by GPUs and or FPGAs before storing in the permanent storage. |
| Sensors/Detectors | MCP's, photo-diodes, ... | Framework of procurement |
| Test requirements | | Appropriate radiation protection |
| DAQ | Spectroscopy | 3MP, 16 bit, 10 fs |
| | MCP | 1000 fs, 16 bit Single frame: 7.63 MB Data rate: 5.96 Gb/s Data on-line: 10 min Data off-line: 10 days Operation mode: 2 hours / day, 10 min sampling, then replace target |
| | Multi spectral imaging | 1 MP, 16 bpp, 100 fs Data rate: 1.5 Gb/s Data online: 10 min Data offline: 30 min Operation mode: 2 hours / day, 10 min sampling, then replace target, |
| | CCD | 1 MP, 16 bpp, 1 Khz Frame size: 2 MB Data rate: 2 Gb/s Data online: 5 hours Data off-line: 90 days Operation mode: continuous, 10 hours / day |





| | | |
|--|---|---|
| | String-array | 4 MP, 16 bpp, 50 Hz Frame size: 8 Mb Data rate: 3 Gb/s Data online: 30 min Data offline: 100 days operation mode: 8 hours / day, 30 min continuous sampling, than replacing target |
| Data processing element (FPGA, GPU, DSP, CPU) | CPU GPU FPGA | NVIDIA GPU compute capability |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | <i>From experiment:</i> 1-40 Gb Ethernet <i>HPC:</i> Infiniband QDR/FDR | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | Standard UDP, TCP/IP Message Passing Interface (MPI) RDMA over Ethernet | |
| Input data formats (CSV, HDF5, ...) | HDF5 Binary (proprietary) | |
| Output data formats (CSV, HDF5, ...) | HDF5 CSV (under consideration ROOT) | Different strategies are being evaluated depending on its application |
| Programming Languages (VHDL, C, C++, CUDA, ...) | C, C++, CUDA. VHDL. | |
| Software including versions (Frameworks, Toolbox, ...) | LAPACK, ScaLAPACK Boost (ROOT) | |
| Algorithmic Details | FFT, ... | |
| Additional Resources (web repositories, links, documentation, ...) | Not available | |
| Aims within UFDAC | Finding effective methods for streaming, processing, reducing, and storing ultra fast data streams. | |





Partner: ELI-NP

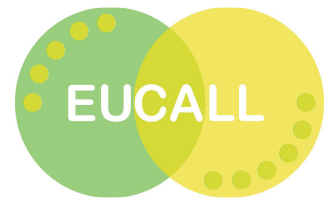
| Item | Description | Detail |
|---|--|--|
| Institution | ELI-NP (IFIN-HH) | Extreme Light Infrastructure – Nuclear Physics / ‘Horia Hulubei’ National Institute for R&D in Physics and Nuclear Engineering |
| Application | Up to 10Hz targets positioning system by image processing | HPL driven experiments with high repetition rate on solid targets require target repositioning and alignment in the focal spot of the laser within 100ms. Image processing is used as a method to provide automatic robot – based positioning of targets and characterization. |
| Setup Details | CCD camera, Ethernet connection, GPU/CPU unit, online processing | |
| Sensors/Detectors | CCD, min 2Mpixels, min 12bit | |
| DAQ | | |
| | | |
| Typical duration of experiment | 4 weeks | 300 -100k shots/day |
| Test requirements | Test bench | |
| Image/Digitizer properties (size, resolution, data format, encoding, ...) | min 2Mpix, 20-30FPS, Tiff/HDF5 | |



| | | |
|--|-------------------------------|---------------------------------------|
| Data bandwidth (in, out, intermediate, ...) | 100MB/s to GPU for processing | |
| Latencies | Not yet determined | |
| Data processing element (FPGA, GPU, DSP, CPU) | GPU, CPU | NVIDIA GPU, series not yet determined |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | Ethernet | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | TCP/IP | |
| Input data formats (CSV, HDF5, ...) | | |
| Output data formats (CSV, HDF5, ...) | HDF5 | |
| Programming Languages (VHDL, C, C++, CUDA, ...) | C, C++, CUDA | |
| Software including versions (Frameworks, Toolbox, ...) | TANGO, Matlab | |
| Algorithmic Details | | |
| Additional Resources (web repositories, links, | | |

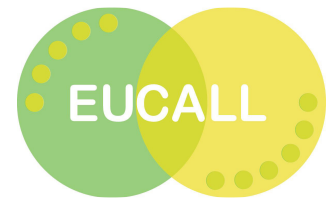
| | | |
|--|---|--|
| documentation, ...) | | |
| Aims within UFDAC | Target alignment/characterization for optimal acceleration | |
| Item | Description | Detail |
| Institution | ELI-NP (IFIN-HH) | Extreme Light Infrastructure – Nuclear Physics / ‘Horia Hulubei’ National Institute for R&D in Physics and Nuclear Engineering |
| Application | In-situ gamma spectroscopy | HPL driven nuclear physics experiments require unconventional active detection techniques due to the harsh environment (EMP) |
| Setup Details | Gated LaBr ₃ detector, digitizer, optical link, PC | Waveform acquisition shortly after the laser pulse followed by onboard signal processing for delayed gamma events. |
| Sensors/Detectors | Gated LaBr ₃ detector | |
| DAQ | digitizer | (manufacturer CAEN or NI) |
| | | |
| | | |
| Typical duration of experiment | 2 weeks | |
| Test requirements | test setup | Detection, DAQ, gate generator, radioactive gamma source |
| Image/Digitizer properties (size, resolution, data format, | ≥12 bit, ≥500 MS/s | |

| | | |
|---|--------------------------------|--|
| encoding, ...) | | |
| Data bandwidth (in, out, intermediate, ...) | To be determined | |
| Latencies | | |
| Data processing element (FPGA, GPU, DSP, CPU) | FPGA, CPU | |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | Optical link | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | PCIe | |
| Input data formats (CSV, HDF5, ...) | binary | |
| Output data formats (CSV, HDF5, ...) | ROOT | |
| Programming Languages (VHDL, C, C++, CUDA, ...) | C, C++, VHDL | |
| Software including versions (Frameworks, Toolbox, ...) | ROOT, LabVIEW FPGA, Quartus | |
| Algorithmic Details | | |
| Additional Resources (web | | |



| | |
|--|------------------------|
| repositories, links, documentation, ...) | |
| Aims within UFDAC | Online data processing |





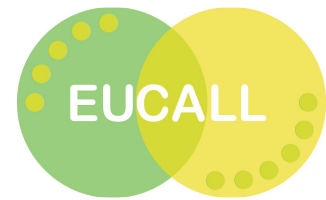
Partner: ESRF

| Item | Description | Detail |
|---------------------------------------|--|--|
| Institution | ESRF | European Synchrotron Radiation Facility |
| Application | X-Ray detectors DAQ | High bandwidth and low latency data transfer using zero-copy RDMA over routed network assuring data integrity. |
| Setup Details | DAQ for X-rays detectors at synchrotron Beamlines. Connection to High Performance Computing system by cables (Ethernet, PCIe, IB, ..) with some online processing functions. Remote analysis user interface. | Streaming of detector data via multiple channels to several memory destinations on many HPC systems. |
| Sensors/Detectors | Target detectors: multiple | Proof of concept likely based on Smartpix pixel detector (ESRF) |
| DAQ | Not Determined | |
| | | |
| Typical duration of experiment | Not Determined | |
| Test requirements | FPGA development (7 family Xilinx FPGA), board, CPU, RoCE NIC, PCIe cables | |
| Image/Digitizer | Not determined | |

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 654220

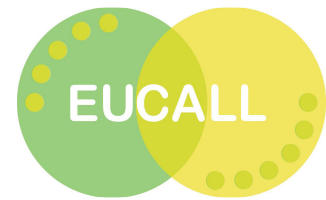


| | | |
|---|--|---|
| properties (size, resolution, data format, encoding, ...) | | |
| Data bandwidth (in, out, intermediate, ...) | Not determined | Depend on the link to be selected in the frame of this study. |
| Latencies | Minimum. Not determined | Depend on the selected link and the system architecture |
| Data processing element (FPGA, GPU, DSP, CPU) | FPGA, CPU | |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | 10 Gbps Ethernet, PCIe, Infiniband | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | RoCE, TCP/IP, PCIe, IB | The protocol is not yet determined, we might need to implement some for comparison purposes. |
| Input data formats (CSV, HDF5, ...) | Binary | |
| Output data formats (CSV, HDF5, ...) | Binary | |
| Programming Languages (VHDL, C, C++, CUDA, ...) | VHDL/Verilog – C – Python – HLS – TCL | Scripts can be done under unix using CSH |
| Software including versions | Vivado 2015 and likely Matlab | |



| | | |
|---|---|--|
| (Frameworks, Toolbox, ...) | Simulink | |
| Algorithmic Details | | |
| Additional Resources (web repositories, links, documentation, ...) | www.molex.com www.mellanox.com www.ieeexplore.com http://intranet.esrf.fr/ www.onestepsystems.com | |
| Aims within UFDAC | Study and prototyping of a generic DAQ platform for High Data rate X-rays detectors | |



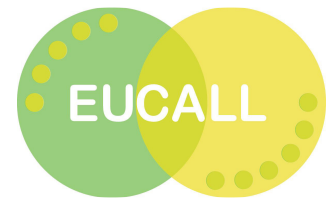


Partner: HZDR

| Item | Description | Detail |
|---|---|---|
| Institution | HZDR | Institute of Radiation Physics, FWK |
| Application | Small-Angle X-Ray Scattering | Method to study structure properties of matter with intense x-ray beams. Specially interest in hot dense matter features in the nanometer range. |
| Setup Details | Pixel detectors. Read out via Ethernet. Laptop or workstation for data acquisition. | |
| Sensors/Detectors | Indirect CCD or CMOS. Direct CMOS | Indirect: PIXIS XF 2048B with GADOX scintillator. Direct: Timepix detector. |
| DAQ | Laptop (Switch to DAQ server in the future for multiple detector acquisition) | Proprietary software (LightField, Sophy). |
| Typical duration of experiment | 2 days | 24h / day measurement time |
| Test requirements | Intense X-Ray beam | |
| Image/Digitizer properties (size, resolution, data format, encoding, ...) | Indirect: 2048 x 2048 pixel image in .spe 3.0 format. Direct: 512x512 pixel image in tiff 32 bit format or ASCII file. | Files can contain multiple frames with specific information (detector parameters, exposure time, other metadata) of each frame contained in a footer. |
| Data bandwidth (in, out, | In: 100 MB / s | about 10 MB per scattering image, assuming 10 Hz repetition |

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 654220

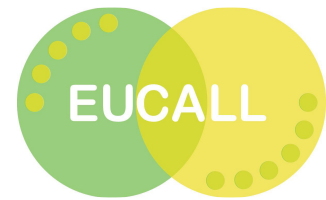




| | | |
|---|--|---|
| intermediate, ...) | Intermediate: 100 MB / s Out: Unknown | rate Flexible user-selectable readout, region of interest and binning |
| Latencies | Unknown | |
| Data processing element (FPGA, GPU, DSP, CPU) | CPU | GPU support planned |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | USB 2.0 USB 3.0 10 Gbps Ethernet | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | | |
| Input data formats (CSV, HDF5, ...) | Binary (proprietary) HDF5 | |
| Output data formats (CSV, HDF5, ...) | Tiff, ascii | |
| Programming Languages (VHDL, C, C++, CUDA, ...) | Python C++/C | |
| Software including versions (Frameworks, Toolbox, ...) | LightField 4.10 Sophy 1.3.5 Python 2.7 ROOT 5.34 | KARABO 1.4.x EPICS 3.14.12.5 |
| Algorithmic Details | | |
| Additional Resources (web) | http://www.princetoninstruments.com/products/PIXIS-XF https://medipix.web.cern.ch/medipix/pages/medipix2/timepix.ph | |

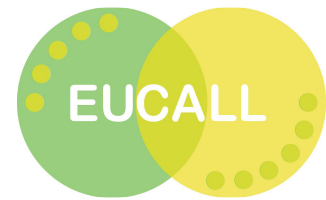


| repositories, links, documentation, ...) | p | |
|---|--|---|
| Aims within UFDAC | General structures for imaging. Generalize data distribution for usage in e.g. imaging | |
| Item | Description | Detail |
| Institution | HZDR | Institute of Radiation Physics, FWK |
| Application | Fast Digital Signal Processing Currently implemented as a testbed application | Testbed Application: Positron Annihilation Spectroscopy |
| Setup Details | DAQ at Accelerator Laboratory Ethernet connection to HPC system with online data analysis Remote analysis user interface | Streaming of waveform data for fitting via GPUs in HPC cluster. Data reduction by ~1000, depending on number of fitting parameters. Storage of fitting parameters to disk |
| Sensors/Detectors | High purity Germanium Semiconductor Detectors | Company: ORTEC |
| DAQ | Spectroscopy Amplifier + Pulse Shaper + Signal to Noise | NIM Amplifier: Canberra 2020 |
| | Keysight Technologies Digitizer | 14 bit, 440 Msps |
| | A/D Wandler | DC440 |
| Typical duration of experiment | 4 days | 24h / day measurement time |



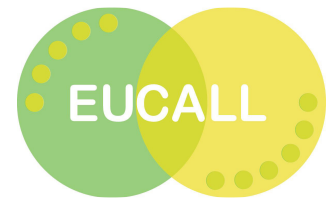
| | | |
|--|---|--|
| Test requirements | Na 22 source | Appropriate radiation protection |
| Image/Digitizer properties (size, resolution, data format, encoding, ...) | 2000 points per event 1D data 4kb/event 8kEvents/s 10 TB/experiment | Waveform data, fixed sampling |
| Data bandwidth (in, out, intermediate, ...) | In: 100MiB/s per Crate Intermediate: 40MiB/s per GPU Out: 0.3 MiB/s for UI/storage | |
| Latencies | Not yet determined | |
| Data processing element (FPGA, GPU, DSP, CPU) | CPU GPU | NVIDIA GPU compute capability > 3.5 (K20 and up) |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | Compact PCI PCIe 10 Mbps Ethernet Infiniband QDR/FDR | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | Message Passing Interface (MPI) ZeroMQ (UDP, TCP/IP, TCP/IP over Infiniband) | Abstract interface via GrayBat |
| Input data formats (CSV, HDF5, ...) | Binary (proprietary) | |
| Output data formats (CSV, HDF5, ...) | ROOT, CSV | Interface to Go4 |
| Programming Languages | C++14 CUDA | C++ template metaprogramming |





| | | |
|--|--|--|
| (VHDL, C, C++, CUDA, ...) | | |
| Software including versions (Frameworks, Toolbox, ...) | cracen graybat (DevBranch 10.03.) Root Boost > 1.56.0 | Application name is 'cracen' 'graybat' is used to abstract communication |
| Algorithmic Details | Levenberg-Marquardt for Fitting | |
| Additional Resources (web repositories, links, documentation, ...) | https://github.com/ComputationalRadiationPhysics/cracen https://github.com/ComputationalRadiationPhysics/graybat | |
| Aims within UFDAC | Generalize data distribution for usage in e.g. imaging | |



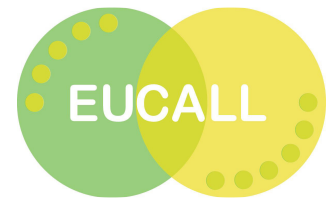


Partner: PSI

| Item | Description | Detail |
|---|--|--|
| Institution | Paul-Scherrer-Institut | Macromolecules and Bioimaging SLS Detectors Group |
| Application | Charge Integrating Detectors | Algorithm to convert dynamic gain switching charge integrating data into single photon counts |
| Setup Details | Jungfrau detector in an X-Ray box or at SLS beamlines | Acquiring data with detector and processing on the FPGA. Afterwards sending to Backend system |
| Sensors/Detectors | Jungfrau, AGPID, Gotthard | |
| DAQ | In house developed multi module detector system | ASIC+Frontend Board(s) |
| | Backend Server(s) | Up to 2x 10GbE connection per detector module to a server |
| | Control PC | Linux Server |
| Typical duration of experiment | Several hours to days | Depending on the experiment data taking can last several days at the beamline |
| Test requirements | Beamline access for pilot experiments | Development and early testing can be done with an X-Ray tube in house |
| Image/Digitizer properties (size, resolution, data format, encoding, ...) | Jungfrau: 16M pixel detector (500k Pixel modules) 75µm x 75µm pixel size 2kHz frame rate 14 bit ADC + 2 bit gain per | 2 MSB encode the gain switching state (G0, G1 and G2) A per gain per pixel offset correction has to be applied to the 14 bit raw ADC value. |

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 654220

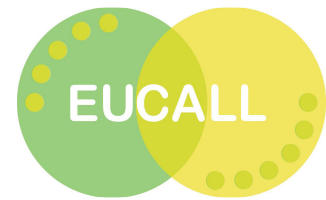




| | | |
|--|--|--|
| | pixel 2D raw data | |
| Data bandwidth (in, out, intermediate, ...) | In: 2.2 GByte/s per module Out: User defined Intermediate: 6.6 GByte/s (Data+Conversion factors) needed to feed the processing unit | Data rate on final storage depends on frame rate requested by user |
| Latencies | FPGA negligible CPU/GPU: Not yet determined (depends on buffer size) | Maximum latency not more than 2 seconds for online displays |
| Data processing element (FPGA, GPU, DSP, CPU) | FPGA, GPU or CPU | Any combination (e.g. FPGA+GPU) is possible. |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | 2x 10GbE – FPGA Optionally converged to 40GbE PCIe-GPU | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | UDP-FPGA to Backend PCIe-GPU | |
| Input data formats (CSV, HDF5, ...) | Raw charge integrated pixel data | 14 bit ADC + 2 bit gain per pixel 512 row x 1024 columns row wise |
| Output data formats (CSV, HDF5, ...) | Photon counted pixel data. I.e. number of photon per pixel in the user selected time frame | In a HDF5 data container |



| | | |
|---|--|--|
| Programming Languages (VHDL, C, C++, CUDA, ...) | VHDL (FPGA) CUDA (GPU) C,C++, Assembler (CPU) | |
| Software including versions (Frameworks, Toolbox, ...) | Altera Quartus Prime Xilinx ISE/Vivado CUDA CERN Root | Altera/Xilinx for FPGA programming CUDA for GPU CERN Root for data evaluation |
| Algorithmic Details | Remove previous measured pixel value offset per gain Apply a per gain scaling factor | A calibration curve with 3 linear sections (one per gain) is needed to recover the number of photons |
| Additional Resources (web repositories, links, documentation, ...) | https://www.psi.ch/detectors/jungfrau https://www.psi.ch/detectors/projects#Charge_Integrating_Read_Out | |
| Aims within UFDAC | Developing high speed detectors with the advantages of both worlds (pixel counting and dynamic gain charge sharing) | |




Partner: XFEL

| Item | Description | Detail |
|--------------------------|--|---|
| Institution | XFEL | European XFEL GmbH |
| Application | Pulse analyzer of digitizer data | Analyze each received pulse that is output of detector and depends on algorithm (Peak detection, Energy of Pulse, zero suppression and ...)generates data |
| Setup Details | Micro TCA crate installation in the LAB | There are eight digitizers in Micro TCA crate that receive data, process it and send processed data in pre-defined packets to software. |
| Sensors/Detectors | MCP, APD, photo diode detectors | Detects photons or electrons and generates pulses based on received particles and these pulses are fed to digitizers for processing |
| DAQ | Micro TCA crates + NAT MCH +Vadatech CPU | Because of remote control capability of Micro TCA system this technology has been chosen. In each crate there is one CPU for running software on it to process and store received data from digitizers, one MCH to control digitizers and provide remote control capability and eight digitizers(from SP-Device company) to receive pulses and do on-line processing |
| | SP-Device Digitizers | 12 bit, 2 GSPS |

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 654220



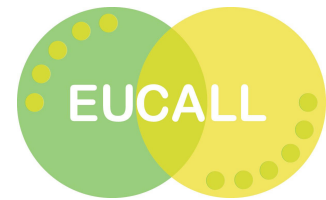
| | | |
|--|---|---|
| Typical duration of experiment | 1 week | 24h / day measurement time |
| Test requirements | Precise signal generator | Agilent M8190A is used in LAB to generate trigger signal with pulses |
| Image/Digitizer properties (size, resolution, data format, encoding, ...) | Digitizer with 12 Bit resolution and 2/4/8 GSPS sampling rate. 100 Mbyte/s output data rate | Peak values, Energy of Pulse, zero suppression. These three different algorithms work on FPGA to get peak time information and peak height of each pulse, Energy calculator integrates samples of each pulse and send this value to software, Zero suppression sends only pulse samples which number of samples per pulse can be configured by user |
| Data bandwidth (in, out, intermediate, ...) | 12 Gbyte/s input bandwidth(four Channel per digitizer) 100 Mbyte/s Output | Four channels per digitizer works with 2GSPS and 12 bit resolution. After processing data encapsulated in packets and depends on Algorithm |
| Latencies | Online processing | |
| Data processing element (FPGA, GPU, DSP, CPU) | FPGA, Xilinx virtex6 | A Xilinx lx240t is used in SP-Device AD412 digitizers and all algorithms has been implemented in this FPGA |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | PCIe interface | PCIe Gen I with 4 lanes is used in digitizers to communicate and transfer data |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | Specialized Protocol | One protocol is defined for communication between software |


| | | |
|---|---|---|
| | | and firmware and for each algorithm there is dedicated packet to use bandwidth efficiently |
| Input data formats (CSV, HDF5, ...) | Specialized Protocol in stream | Software receives processed data in predefined packets and after extraction saves data in HDF5 |
| Output data formats (CSV, HDF5, ...) | HDF5 | Software saved data in HDF5 format |
| Programming Languages (VHDL, C, C++, CUDA, ...) | VHDL, C++ | VHDL is used to design Firmware for FPGA in Xilinx ISE12.4 and C++ is used to develop software |
| Software including versions (Frameworks, Toolbox, ...) | Xilinx ISE 12.4 Questa Sim(for simulation) SP-Device DevKit | Because of SP-Device DevKit Xilinx ISE12.4 is used for firmware development. Mentor graphics Questa sim is also utilized for simulation. SP-Device Devkit is an IP Core that provides all IO communication for digitizer (PCIe, ADC configuration and ...) |
| Algorithmic Details | | |
| Additional Resources (web repositories, links, documentation, ...) |  <p>Acquisition Data Interchange Format https://docs.xfel.eu/alfresco/d/a/workspace/SpacesStore/9fc815ad-c923-45eb-a547-e64b99aaa727/Acquisition%20Data%20Interchange%20Format.docx</p> | |

| | | |
|--------------------------|---|--|
| Aims within UFDAC | Exchange of experience and possibly spreading the developed interfaces and protocols if interest exists | |
| Item | Description | Detail |
| Institute | XFEL | European XFEL GmbH |
| Application | Low Latency Protocol | Communication interface developed to be the high-speed link to distribute beam information with minimal latency, allowing information to be usable on acquisition systems before data arrives. |
| Setup Details | High-speed link communication between FPGA based data acquisition devices. | Detectors, diagnostics or decision devices. |
| | Timing System | Device providing a common clock reference, synchronization and timing information across all devices |
| Sensors/Detectors | 1D/2D detectors VETO System Diagnostic System | Rejection of acquired data Beam based feedback system to improve the beam parameters based on diagnostic or detector systems |
| DAQ | FPGA with compatible interfaces | Compatible interfaces and transmission mediums (connectors, cables) need to be provided |
| | Timing System | Common clock and trigger reference to all systems together |

| | | |
|--|---------------------------------|---|
| | | with timing information can be provided from an external source (Timing System) or generated internally in the FPGA Board that contains the protocol. |
| Typical duration of experiment | 7 days / Week | 24h / day measurement time |
| Test requirements | FPGA with compatible interfaces | FPGA Boards to simulate transmitter and receiver devices |
| | Signal Generator | Trigger and clock reference for all systems. Timing information could be generated internally on the FPGA board acting as the Transmitter. |
| Image/Digitizer properties (size, resolution, data format, encoding, ...) | 1D/2D Detector data | |
| Data bandwidth (in, out, intermediate, ...) | In/Out: >1.25 Gbps | Data bandwidth typical application is 3.215 Gbps, with 16 bit I/O. I/O can be configured during operation and bandwidth is limited by the Interface (10 Gbps is achievable using SFP+ interfaces) |
| Latencies | > 22 clock reference cycles | Information related to transmission data (16 clock cycles) plus data, whose process depends on data word size (6 clock cycles for 16 bit data) |
| Data processing | FPGA | GTP/GTX Transceivers |

| | | |
|---|------------------------|--|
| element (FPGA, GPU, DSP, CPU) | | |
| Interfaces for data transfer (10Gbps Ethernet, PCIe, ...) | SFP/SFP+/Ethernet | |
| Protocols (TCP/IP, UDP/IP, PCIe, ...) | Custom XFEL protocol | Configurable protocol that allows data type and size to change during operation. Data information is sent in advance to optimize the receiver's logic to the data type and size to be transmitted. |
| Input data formats (CSV, HDF5, ...) | User defined | Information concerning the data to be transmitted is send in advance of the actual data. Parameters sent include data type, data word size, IDs (beam and transmitter). This information can be reconfigured during operation. |
| Output data formats (CSV, HDF5, ...) | User defined | Read detail section of Input data formats. |
| Programming Languages (VHDL, C, C++, CUDA, ...) | HDL | VHDL GTP/GTX core manager provided by FPGA vendor HDL Protocol |
| Software including versions (Frameworks, Toolbox, ...) | Xilinx ISE Simulink | XFEL High Level FPGA Framework |
| Algorithmic Details | Not applicable | |



| | |
|---|---|
| Additional Resources (web repositories, links, documentation, ...) | XFEL EuCALL Public Folder https://docs.xfel.eu/share/page/guest-access?nodeRef=workspace://SpacesStore/d7f3dfec-01c4-42b5-a0d0-f290e1c2f80d  |
| Aims within UFDAC | Generalize timing information and beam data distribution for usage in e.g. imaging, diagnostics |

