ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

PANDABOX: A MULTIPURPOSE PLATFORM FOR MULTI-TECHNIQUE SCANNING AND FEEDBACK APPLICATIONS

S. Zhang*, Y.M. Abiven, J. Bisou, G. Renaud, G. Thibaux, F. Ta,
S. Minolli, F. Langlois, Synchrotron SOLEIL, Gif-sur-Yvette, France
M. Abbott, T. Cobb, C.J. Turner, Diamond Light Source, Oxfordshire, UK
I.S. Uzun, Rutherford Appleton Laboratory, Didcot, UK

Abstract

PandABox is a development project resulting from a collaboration between Synchrotron SOLEIL and Diamond Light Source started in October 2015. The initial objective driving the project was to provide multi-channel encoder processing for synchronizing data acquisitions with motion systems in experimental continuous scans. The resulting system is a multi-purpose platform well adapted for multi-technique scanning and feedback applications. This flexible and modular platform embeds an industrial electronics board with a powerful Xilinx Zynq 7030 SoC (Avnet PicoZed), FMC slot, SFP module, TTL and LDVS I/Os and removable encoder peripheral modules. In the same manner, the firmware and software framework has been developed in a modular way to be easily configurable and adaptable. The whole system is open and extensible from the hardware level up to integration with control systems like TANGO or EPICS. This paper details the hardware capabilities, platform performance, framework adaptability, and the project status at both sites.

PROJECT INTRODUCTION

PandABox (Position and Acquisition Box) was initiated to overcome obsolescence problems and technical limitations in Synchrotron SOLEIL and Diamond Light Source concerning their in-house designed products SPIETBOX (SOLEIL) and ZEBRA (Diamond); two products that have been integral to beamline operations to synchronize various equipments and for position capture.

Figure 1: PandABox, a collaboration project between Diamond and SOLEIL.

The collaboration [1] proved efficient in sharing leadership and experience between both synchrotrons in hardware, firmware, and software developments. Along the development process, the two institutes have worked closely to review and validate the design at each step: 2 prototypes have been designed and improved within the scope of the project and the PandABox is now validated for production. The firmware and software have been improved and are now stable for EPICS or TANGO control system interfacing. The project is also available on the Open Hardware Repository (OHWR) to share with other institutes.

The PandABox project [2] had four repositories were created: PandABox-gw, PandABox-hw, PandABox-sw, and PandABox-tst (giving acces to the official releases of the firmware, TCP server, webserver and hardware source files). All documentation is accessible such as Hardware user guides and FPGA development Kits for PandABox firmware. Although the initial deliverables of the project are complete, both institutes still do collaborative work to improve the platform based on their own experience with the PandABox.

HARDWARE DESIGNS

PandABox is packaged in a 19" 1U rack which integrates several PCB boards: one detachable front panel board, one carrier board holding 4x encoder daughter modules and 1x PicoZed SOM (System-On-Module) module. The connection between the front panel and the carrier board is made through 2x FFC cables. On the carrier board, an FMC (FPGA Mezzanine Connector) slot is using an LPC (Low Pin Count) connector that can optionally receive an FMC mezzanine card to extend the hardware functionalities and I/O capabilities. [3]

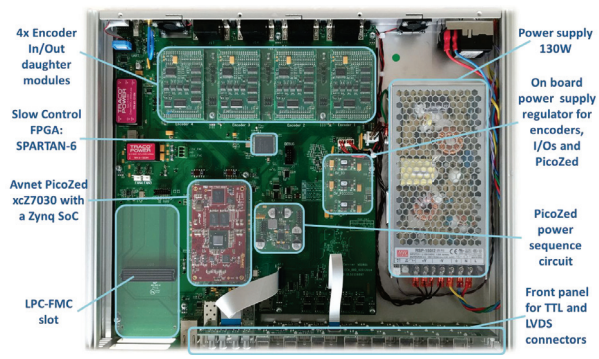

Figure 2: PandABox internal hardware top view.

The hardware was designed to meet the requirements for simultaneous and multi-technique scanning applications with support for a wide range of encoder protocols. The architecture presented in Fig. 3 was developed to be modular and flexible in order to enable the maximum number of applications.

* szhang@synchrotron-soleil.fr

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

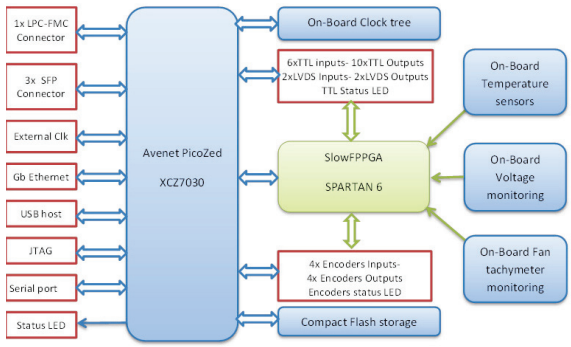

Figure 3: PandABox hardware global architecture.

PandA Carrier Board

The PandA carrier board is fixed at the bottom of the rack, providing it with a variety of connection interfaces for the I/O boards and their modules. This 8-layers PCB carrier board is powered by an external commercial AC/DC 12V/130W power supply module and it integrates an electronic circuit to provide regulated power supply for: the PicoZed, the encoders boards, the slow FPGA, the I/Os, etc. Up to 8 different DC voltage supply lines (1-24V range) are provided on the carrier board of which a regulated ±15V power supply is available for the FMC mezzanine card if required.

The encoder interface is designed in a modular way: 4x dual stacked 15-way D-type connectors for 4-channel encoder inputs and outputs are mounted on the carrier board and have connections with the encoder daughter modules. The connector genders and pinouts can be modified by adapting the encoder daughter modules. For this reason the encoder module has been developed to adapt to SOLEIL and Diamond requirements so that both institutes can share the same hardware and maintain their own encoder connectivity standards. Each daughter module is designed with buffers to interface different encoder types and currently supports quadrature incremental encoders and absolute encoders with SSI, EnDat or BiSS-C communication protocols.

Special care has been taken to the PCB design of sensitive signals such as differential pairs and clocks, fully-compliant LPC-FMC, Gbps Ethernet and MGTs (Multi-Gigabit Transceivers). This step includes pair matching and length tuning techniques up to the PicoZed socket. The LPC-FMC slot is placed on the left side of the carrier board. Behind the FMC connector there remains space for the use of D-TACQ ELF products which fit the same connector and front panel footprint as standard FMC modules but are physically larger to allow for a bigger payload. [4]

Control and Data Processing Elements

PandABox slow controls, realized by a Xilinx Spartan-6 FPGA in the central of the carrier board, are tasked with:

- Temperature monitoring with 5x temperature sensors distributed on the carrier board, via I2C bus from sensors to the FPGA.
- On-board power supplies voltage monitoring.
- Fan-speed monitoring with fan tachymeter sensors.
- Configurations of the I/O encoder signals specific to encoder type (Incremental, SSI, BiSS-C or EnDat).
- Front panel configurations through shift register signals: select High-Z or 50-Ohm impedance for the TTL Inputs, control two among four status LEDs (two other LEDs are controlled by PicoZed).

Data acquisition, data processing, and data exchange with control systems are managed by the PicoZed SOM module [5]. It's the backbone of the system and is based on the Xilinx Zynq-7030 AP (All Programmable) SoC (System-On-Chip). The main processor, Zynq-7030 SoC, is the heart of the system; it embeds a Dual-core ARM Cortex-A9 processor and Artix-7 FPGA based programmable logic. As shown in Fig. 4, one side the ARM processor runs a TCP server on an embedded Linux OS which provides system control and data capture for clients. On the PL (Programmable Logic) side, FPGA firmware provides Functional Blocks like encoder position processing for triggering and data acquisition (section **FIRMWARE AND SOFTWARE**).

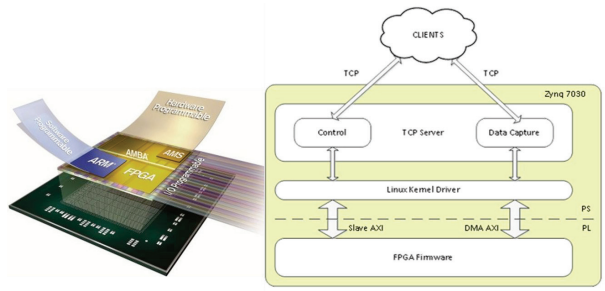

Figure 4: Zynq SoC firmware architecture.

I/O and Communication Interfaces

Various connectors are installed on the front and rear panels to interface different signals (encoder, trigger, timing...) and to communicate with platform users, control systems or other electronic platforms, see Fig. 5.

Figure 5: PandABox front and rear panels (SOLEIL version)

The connectors and their supported I/Os are listed as below:

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

- Multi-Channel TTL and LVDS I/Os for synchronous triggering and clocking signals with high timing resolution via BNC, LEMO and SMA connectors.
- 8x 15-way D-type connectors for 4-Channel encoder inputs and 4-Channel outputs.
- 1x FMC slot with LPC (Low-Pin Count) connector for receiving an optional analog or digital off-the-shelf or custom FMC Mezzanine card (e.g. ADC, DAC, TDC cards etc.).
- 1x Gigabit Ethernet connector for control system integration and high-speed data acquisition.
- 3x SFP (Small Form-factor Pluggable) sockets providing modular interfaces that can easily adapt to various fiber optics and copper networking standards by connecting compact and hot-pluggable SFP transceiver module to one socket.
- 1x JTAG interface for Zynq and Spartan-6 FPGA programming and debugging during developments.
- 1x RS-232 port providing a serial console terminal to access the embedded Linux system.
- 1x USB host allowing to connect USB memory for firmware upgrades.

Prototype Validation

The first prototype has been fully tested at both institutes at the end of 2016. After a few design and manufacturing corrections, electrical tests were successfully carried out. The Zynq SoC boots up quickly and the startup sequence can be monitored through the RS-232 UART terminal. The slow control FPGA was programmed from an on-board PROM during initial tests. It's now programmable via Zynq SoC MIOs which takes about 11 seconds. Once running and configured on the network, communication with the platform is functional over the TCP server. All digital TTL and LVDS I/Os on the front panel as well as the encoder inputs and outputs on the rear panel are fully operational. The SFP ports were tested with an SFP transceiver module and a loopback firmware using a Xilinx customizable IP core IBERT (Integrated Bit Error Ratio Tester) designed for evaluating and monitoring the 7 series FPGA GTX transceivers. Each SFP port together with a Zynq GTX, has been validated at 6.25Gbps and 2.5Gbps using GTX-CLK0@125MHz with up to 550m optical fiber cables, giving acceptable eye scan results as shown in Fig. 6.

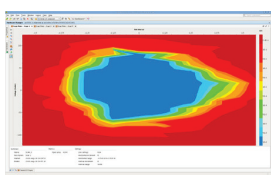

Figure 6: IBERT eye scan result.

A similar test for the FMC socket has been done with a dedicated testing-firmware and an FMC loopback mezzanine card. The test allowed to validate all the FMC I/O, GTX connectivity and clocking pins.

However, while debugging an internally developed firmware for the FlyScan software synchronization over UDP frames through an SFP port (section **PANDABOX APPLICATIONS AT SOLEIL**), overheating was observed on the Zynq SoC component of a commercial-grade PicoZed module which sometimes caused the SoC OS to reboot. Measured with a thermal camera as shown in Fig. 7, the temperature difference between two PicoZed grades with the same firmware is significant. For this reason, industrial-grade PicoZed modules will be mostly employed for SOLEIL operations.

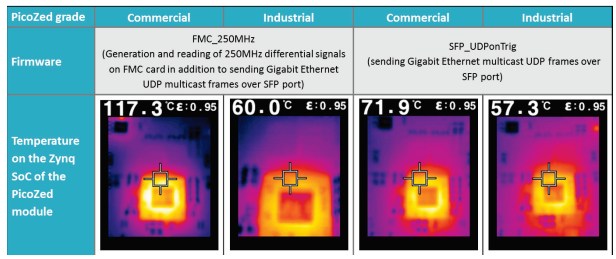

Figure 7: PicoZed overheating.

FIRMWARE AND SOFTWARE

The PandABlocks framework is an open-source firmware and software stack developed by Diamond that powers the PandABox. [6]

The firmware is structured into numerous Functional Blocks (FBs) on the PL part of the Zynq device, with each block configured via a number of registers. They can be wired together at run-time according to application-specific requirements and an FB is controlled and interrogated through the associated registers. Status reporting and high speed data acquisition is handled by the Processing System (PS) on the same Zynq device and exposed via a TCP server with a protocol suitable for integration into control systems like EPICS or TANGO.

FPGA Functional Blocks

The list of logical functional blocks includes:

- Function generators to implement any 5-input Boolean function.
- Set/Reset Gate blocks with configurable inputs, and an option to force its output independently.
- 32-bit Pulse divider with programmable divisor and two pulse outputs (divided and non-divided).
- Pulse Generator blocks to produce configurable width and delayed pulses triggered by its input and also able to handle pulse trains.
- Sequencer blocks to perform automatic execution of sequenced frames to produce timing signals.

Similarly, a wide range of position based blocks are implemented:

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

- Encoder Input/Output blocks supporting multiple encoder protocols.
- Quadrature Input/Output blocks to enable of any discrete signal on the system for quadrature operations.
- Counter/Timer blocks to implement multi-channel 32-bit up/down counters synchronous to system clock.
- Analog input/outputs up to 8-channels through FMC module connectivity.
- Position Compare (PCOMP) to generate a stream of trigger pulses based on: position, time, analog input, or external triggers. It can use LUT-based arbitrary arrays of position or time points instead of regularly spaced series of values. It's possible to mix modes together by cascading multiple PCOMP, such as outputting a time-based pulse-stream within a series of position-based gates to support continuous scanning experiments.

A central Position Capture (PCAP) block is responsible for capturing user-defined fields across the design including timestamps, encoder values, analog input values, counters and system status on trigger events.

As shown in Fig. 8, wiring of FBs is achieved via a common Bit-bus and Position-bus routed internally across the design. A 128-bit bit-bus is used to feed Functional Block's bit-type inputs while a position-bus is used to feed the position-type inputs. Configuration and monitoring of FBs are achieved via registers mapped to processor memory address space.

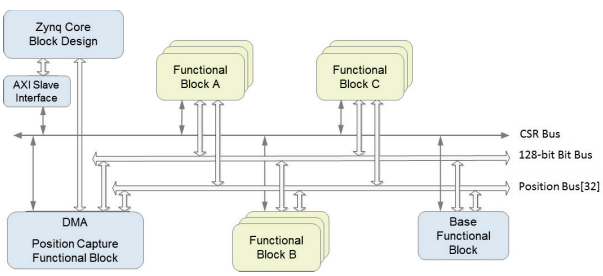

Figure 8: PandABLocks framework architecture.

New FBs (Functional Blocks in Fig. 8) can be added by implementing VHDL modules for user-defined configurable functionalities. There can be multiple instances of these FBs [6]. Each Functional Block has:

- Discrete bit-type input and output ports,
- 32-bits wide position-type input and output ports,
- Configuration and status registers.

A simple block diagram of a generic FB is shown in Fig. 9:

At SOLEIL some FBs are developed or are under development to feed application requirements (see section **PANDABOX APPLICATIONS AT SOLEIL**), with the help of the firmware development kit provided by Diamond.

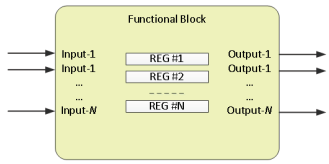

Figure 9: A simple block diagram of a generic Functional Block.

Software Architecture

Fig. 10 depicts the system architecture for the PandABox firmware and software. A Linux OS running on the ARM processor is managing the following peripherals:

- QSPI flash containing persistent configurations (e.g. MAC address).
- SD card containing the remaining Linux file system.
- Network system interfacing.
- The FPGA programs, and software interfacing.

Flexibility of the architecture is achieved through a set of configuration files which describes each Functional Block's input and output ports, configuration registers, and descriptions. This approach allows the design and compilation of a custom set of Functional Blocks into the firmware with access from the TCP Server without rebuilding the TCP server and webserver.

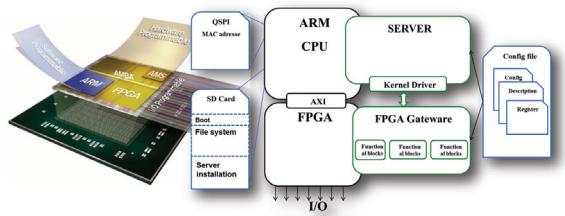

Figure 10: Flexible PandABox firmware architecture.

The TCP socket server, acting as a bridge between the FPGA firmware registers interface and remote clients, publishes two socket end-points: one for configuration control, and the other for streamed data capture (see Fig. 4). The configuration control socket accepts simple ASCII commands and returns all data in readable ASCII format. The data capture socket supports no commands and simply streams captured data in a lightly structured binary format.

Wrapped by the TCP Server, the kernel driver provides control and status register access to system blocks, and interfaces with the DMA engines for synchronous read/write data transfers. The kernel driver provides low-level interfacing between the firmware FBs and upper software layers through the TCP server. Its firmware specific implementation includes:

- A generic register address space for direct access to each firmware FBs' configuration and status registers.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

- A Read-DMA engine to receive position capture from the PCAP block in real-time.
- A Write-DMA engine for writing user-defined tables to the PCOMP blocks for position compare operation.

Fig. 11 depicts the standalone software architecture for the PandABox project. It is our intention that PandABox will be integrated either with an EPICS IOC, a web-server interface, or with a separate TANGO Device Server.

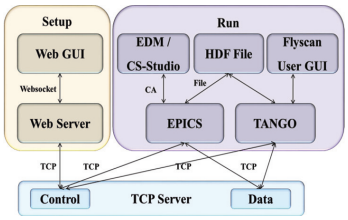

Figure 11: PandABox software architecture.

Building on top of the TCP server, a web-server was developed via a web GUI, also included in the framework, to visualize and change the wiring of the Functional Blocks as well as setting their parameters. It proves very useful during SOLEIL new FBs development. Shown in Fig. 12 is a low-level configuration for FlyScan application with a SOLEIL developed block SFP, whose I/O ports and parameters are accesible through this graphic interface.

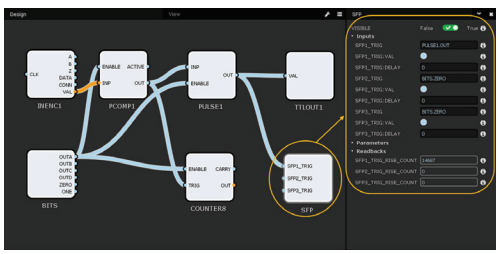

Figure 12: PandABox web GUI interface.

TANGO SOFTWARE AT SOLEIL

3 types of TANGO Device Servers (DServers) are currently being created: hardware *PandABox* DServer for box configuration and monitoring, generic *PandABox-DataViewer* DServer for customized interfaces, and specific DServers for dedicated applications. All these DServers will communicate with the platform through a PandABox library as shown in Fig. 13. It implements the communication protocol for TCP/IP clients and provides functionalities for higher level clients, which is currently under development at SOLEIL.

Hardware DServer: PandABox

The PandABox TANGO DServer provides basic controller functions:

- Read controller temperature

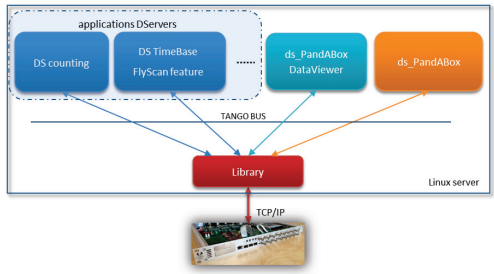

Figure 13: TANGO software architecture at SOLEIL.

- Get firmware version
- Load configuration file
- Get and save PandABox configuration
- Get all connected clients' information
- Send low level ASCII-format command

This DServer has been developed, as shown in Fig. 14

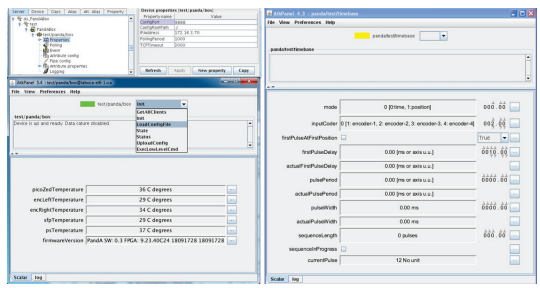

Figure 14: TANGO DServers PandABox (left) and PandABoxTimeBase (right) interface.

Application DServers

The application DServers allow to carry out functions control for different applications (e.g. FlyScan timebase, pulse counting, analog acquisitions etc.). We are currently focused on the development of the DServer PandABoxTimeBase used for continuous scans (FlyScan) applications, which is the initial motivation of the PandABox project. It aims to control the generation of hardware signals for FlyScan synchronization (managed by the last generation product SPIETBOX). The software is being implemented and tested with the PandABox (see Fig. 14). In addition, another DServer PandABoxUDPTimeBase will be developed to generate FlyScan software synchronization notifications. The study of this function is ongoing. Other DServer applications will be defined in the future according to the application requirements.

Generic DServer: PandABoxDataViewer

PandABoxDataViewer, a generic DServer, dynamically creates attributes and commands to structure a customized interface. It is very useful for a new application prototyping or expert diagnosis due to its generic nature and flexibility.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

The DServer interface can be generated by defining the expected commands, attributes and state/status in the DServer property, by specifying their low-level commands/registers name, read/write access rights, data types etc. This DServer is currently under specification and will be developed in the near future.

PANDABOX APPLICATIONS AT SOLEIL

Several applications were classified as part of the initial motivations for the PandABox project whereas some upgrade applications were identified during project development.

FlyScan Synchronization

The first application of PandABox is to generate synchronization signals for data acquisitions during experimental continuous scans. Different to the classic step-by-step scans, correlated data are simultaneously collected, spatially or temporally, from different sensors along the continuous trajectory of one (or more) actuator(s). This approach has been more and more adopted in Synchrotrons which can significantly optimize scanning experiments by potentially saving hours of beam-time and giving access to process-dynamics (e.g. thermal annealing, continuous mechanical deformation, etc.). The continuous scanning or mapping is implemented with FlyScan project [7] at SOLEIL and Malcolm framework [8] at Dimaond.

At SOLEIL, the distribution of FlyScan synchronization signals are in the form of hardware (e.g. TTL) or software (UDP) notifications. The software notification is often used to synchronize less critical data acquisitions during a scan, especially for equipments that don't possess any hardware trigger input. The hardware and software notification signals are currently generated by two in-house cards: SPIETBOX and SPIController, which are going to be replaced by the PandABox.

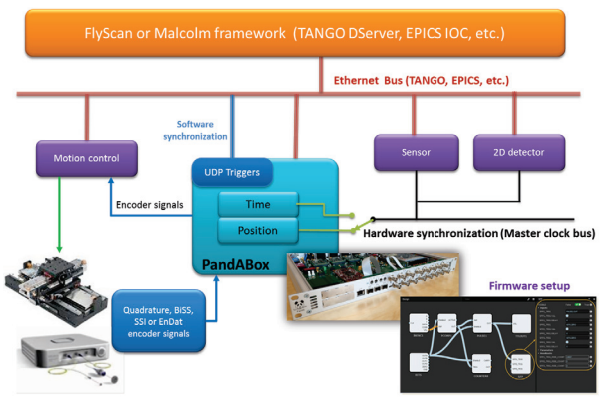

Figure 15: Continuous scanning synchronization with PandABox.

To perform the hardware triggering with PandABox, one example configuration has been made to demonstrate the triggers generation in different conditions. Once the TANGO software is developed, continued tests could be launched to integrate the PandABox into a real FlyScan environment.

With SPIController, the software notification is multicasted over Ethernet UDP frames. In the same way, a new PandABox Functional Block has been added at SOLEIL to send multicast UDP frames out over an SFP port and an SFP-RJ45 converter module in the event of hardware triggers. The firmware development needs to be finalized with FlyScan notification protocol adaptations, after which the TANGO software needs to be specified and developed.

Hardware Encoder Processing

The multi-channel encoder processing is part of the initially identified PandABox applications. By parameterizing the Functional Blocks and cascading them into a certain configuration, we can easily make a standalone running encoder processing machine that could provide position calculations in addition or subtraction of 2 encoder inputs, multiple encoder averaging, encoder protocol conversions, etc. The resulting encoder information could be:

- Transferred to acquisition-systems through the TCP server
- Used by other internal processes (e.g. FlyScan trigger generation)
- Outputted to other external systems (e.g. motion controller, another PandABox)

Fast Beam Attenuation Controls Upgrade

This particular application is to upgrade and improve the existing Real-time control system for the fast beam-attenuation with an XPAD 2D detector.

Current Control System To overcome the detector maximum count-rate limitation, which is one of the main limitations of the state-of-the-art counting hybrid pixel detectors, a fast beam-attenuation system was developed at SOLEIL. It has been carried out at the SixS (Surfaces Interfaces X-ray Scattering) beamline of SOLEIL. This system is fully autonomous that allows for a dynamic change of the beam-attenuation as a function of the photon flux received by an XPAD S140 photon counting detector. [9]

In addition to the detector the system comprises of three main elements: the XPAD 2D detector intensity analyzer, the control board, and two sets of attenuating foils (see Fig. 16).

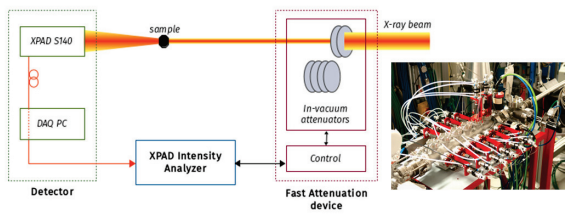

Figure 16: Fast Attenuation system on SixS beamline.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

The control board connects directly to the attenuators hardware to command the attenuation changing actions. It is based on a CompactPCI board (TCP631 motherboard from TEWS Technologies with custom mezzanine board) with the core element being a Xilinx Spartan-3 FPGA.

The attenuators constitutes of a series of metallic foils, each being twice the thickness of the previous one, each being actuated with binary logic that moves the filter in and out of the beam pathway. This series of binary logic permits to cover a large attenuation range with a small number of single metallic foils.

The detector intensity analyzer performs a cyclic estimation of the photon flux from the image acquisitions of the XPAD 2D detector. The estimated count rate value is then compared with two selected thresholds that define the desired count rate range. The system has been implemented in a processing-unit based on a commercial FPGA development platform (Cyclone V GX Starter Kit from Terasic Technologies Inc). This board is connected to the detector with a single optical fiber through a PCI Express board.

The Fast Attenuation device is controlled via TANGO Controls software that provides access to configuration parameters such as: thresholds, attenuator set selections, and manual/automatic operation. [9]

Upgrade Controls with PandABox The upgrade solution is to combine the detector intensity analyzer function and the control board function into one system in a PandABox platform (see Fig. 17).

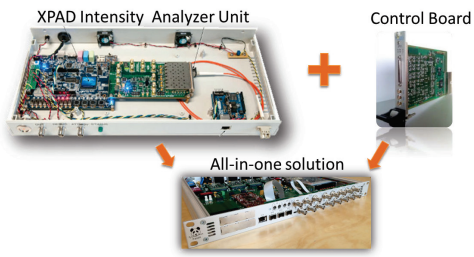

Figure 17: Fast beam-attenuation system upgrade with PandABox.

The motivation of this solution is to benefit from 3 main advantages:

- Technical support from the ECA group: operation and maintenance services (e.g. on-call service).
- All-in-one solution: combining two hardware modules into one makes the control system more compact.
- Performance improvements with the PandABox onboard FPGA/SoC provides high processing capabilities.

The first step is to migrate the detector intensity analyzer firmware code provided by the detector group from an Altera FPGA to the PandA Xilinx target; a firmware migration with inherent difficulties with IP and syntax incompatibilities due to the different FPGAs and programming tools. The firmware transfer is however almost completed and tests with the detector on the SixS beamline is currently in progress. Fig. 18 shows ILA (Integrated Logic Analyzer) captured detector data frames.

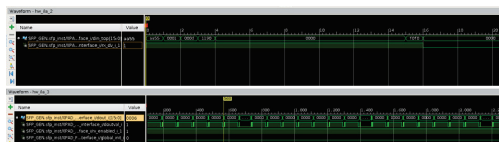

Figure 18: ILA captured detector data frames with PandABox.

The second step will be to improve the analyzer by modifying its calculation algorithm; something that will be done in close collaboration with the SOLEIL detector group.

The third step will be to migrate the control board functions to PandABox, of which the final step will be to fully integrate the full solution.

The TANGO Dserver PandABoxDataViewer is supposed to control the system and provide access to the configuration parameters during application commissioning.

SUMMARY AND NEXT STEPS

The initial objectives have been achieved and it has been confirmed that the resulting system is well adapted as a multi-purpose platform, allowing us to address a large range of applications such as multi-technique scans and feedback applications.

The following developments are to be implemented at SOLEIL:

- Finalizing Tango DServers (PandABoxTimeBase, PandABoxDataViewer).
- Finalizing the UDP notification FB development for FlyScan software triggering and developing the PandABoxUDPTimeBase TANGO DServer.
- Finalizing the fast beam-attenuation analyzer migration and starting the actuator control upgrade.

Next steps for SOLEIL will be the deployment and commissioning of the PandABox on beamlines for FlyScan applications, the fast beam-attenuation application, and call for tenders for PandABox supplies. Diamond is currently providing feedback about the experience using PandABox for beamline applications; improvements to have a more easily configurable system are currently being discussed. The PandABox is now available on the OHWR, to share with other institutes, which is expected to help improve the platform and give new ideas of PandABox-applications in research areas.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUAPL05

REFERENCES

[1] I.S. Uzun, at al., PandA Motion Project - A Collaboration Between SOLEIL and Diamond to Upgrade Their 'Position and Acquisition' Processing Platform, Proc. of ICALEPCS'2015, MOPGF098, Melbourne (Australia).

[2] PandABox project on OHWR, https://www.ohwr.org/projects/pandabox/wiki/wiki.

[3] YM. Abiven, at al., PandABox: a multipurpose platform adapted for multi-technique scanning and feedback, Proc. of PCaPAC'2016, THHWPLIO01, Campinas (Brazil).

[4] D-TACQ Solutions, http://www.d-tacq.com.

[5] Avnet, Picozed Product Brief, http://zedboard.org.

[6] Chris.J.Turner, at al., PandABlocks Open FPGA Framework and Web Stack, Proc. of ICALEPCS'2017, THPHA068, Barcelona (Spain).

[7] N. Leclercq, at al., Flyscan: a Fast, Simultaneous and Multi-technique Data Acquisition Platform for the SOLEIL Beamlines, Proc. of ICALEPCS'2015, Melbourne (Australia).

[8] T. Cobb, at al., Malcolm: A Middlelayer Framework for Generic Continuous Scanning, Proc. of ICALEPCS'2017, TUPHA159, Barcelona (Spain).

[9] A. Dawiec, at al., Real-time control of the beam attenuation with XPAD hybrid pixel detector, 2016 JINST 11 P12018.