PandABox: A MULTIPURPOSE PLATFORM ADAPTED FOR MULTITECHNIQUE SCANNING AND FEEDBACK

Yves-Marie Abiven†, Jérome Bisou, Guillaume Renaud, Frédéric Ta
Synchrotron SOLEIL, Gif-sur-Yvette, France

Isa Servan Uzun, Michael Abbott, Andrew Cousin, Tom Cobb
Diamond Light Source, Oxfordshire, United Kingdom

Abstract

Synchrotron SOLEIL and Diamond Light Source are two third generation light sources located respectively in France and the UK. In 2015, both facilities started the collaboration project "PandA" to overcome technical limitations of SPIETBOX at SOLEIL and Zebra at Diamond as well as to manage obsolescence of the products. The collaboration enables both institutes to share the technical leadership on hardware, firmware and software developments. The initial objective is to achieve multi-channel encoder processing to synchronize motion systems and data acquisition during experiments addressing simultaneous and multi-technique scanning. However, its design based on Xilinx Zynq SoC is thought to be powerful and modular in terms of firmware as well as for hardware. This flexibility permits envisaging derivative applications and interfacing to different third party hardware. This paper details the organization of this collaboration, status of the ongoing project in terms of hardware and firmware capabilities and the results of the first tests at both sites.

PROJECT STATUS

The first step on the project confirmed that the organisation of the project was efficient, sharing leadership between institutes. After few months agreeing specifications in term of hardware and firmware, the last 10 months have been intense, focusing on the design. On one side SOLEIL designed the 3 main boards which are integrated in a 19" rack. On the other side Diamond developed the firmware and software to interface PandABox to EPICS or TANGO control system, up to being able to configure the system and make acquisition. At each step of the design both institutes worked together towards reviewing and validating designs. The First prototype presented in Figure 1 was delivered in August. The prototype hardware was tested successfully along with initial firmware and software releases at both sites during September, and only a few hardware modifications were identified. Update of the electronic and mechanic designs were completed in October and second prototypes were ordered with expected delivery in January.

On the firmware side, Diamond delivered first version of the firmware allowing both institutes to start hardware tests: Processor, FPGA, SFP, FMC and all inputs and outputs. Communication via a TCP server permits to remotely work with the platform: Reading on-board sensors, setting and acquiring encoder position or generating triggers on TTL outputs. All standard configurations required for first application are available. Today the functional tests of the firmware are ongoing and already promising.

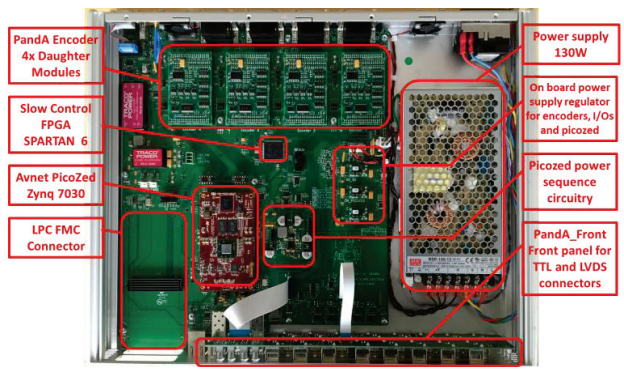

Figure 1: PandABox inside view.

HARDWARE ARCHITECTURE

The hardware architecture is designed to meet the requirements for simultaneous and multi-technique scanning with support for a wide range of encoder protocols. The architecture presented in Figure 2 was developed to be modular and flexible in order to be open for a maximum numbers of applications. The platform is built with:

I/O interfaces

- Multi-Channel TTL and LVDS I/Os for synchronous
- triggering and clocking.4-Channels of encoder interface I/Os, supporting Incremental, Absolute encoder protocol (AquadB, SSI, EnDat and BiSS). Integrated on the platform as a mezzanine boards allowing this functionality to be updated.
- A fully compliant Low-Pin Count FMC slot for interfacing to analog and digital off-the-shelf boards or custom I/O modules.
- JTAG allowing programming and debugging for Zynq and Spartan-6 during prototyping.
- Serial port, giving a terminal for linux administration.

Communication interfaces

- 3-Channels of SFP Gigabit Transceiver interface for UDP triggering, Timing System, Diamond Communication Controller or custom high-speed serial connectivity.
- A Gigabit Ethernet connectivity for control systems integration and high-speed data acquisition.

† yves-marie.abiven@synchrotron-soleil.fr

ISBN 978-3-95450-189-2

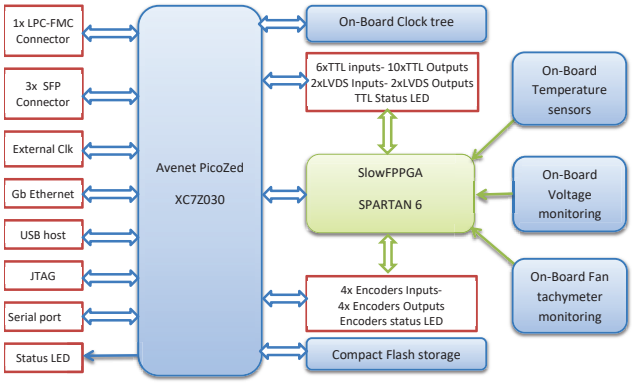

Figure 2: PandABox architecture.

Control and data processing

- A Slow Control FPGA to control the configurations of the I/O encoder signals specific to encoder type, TTL inputs impedance and some status LEDs on front panel. It also interfaces to on-board temperature, voltage and fan-speed sensors for monitoring purposes.
- The main processor, SoC Zynq 7030, based on an off-the-shelve product AVNET PicoZed [1] for tighter system integration. It is the heart of the system, managing data acquisition, data processing and data exchange with control system.

Additionnal functionalities

- USB Host, permitting to connect USB stick to upgrade the firmware is available in the rack

Hardware Design

PandABox is develop in a 1U 19" rack which integrates four boards: a carrier board, a front panel,4x encoder daughter boards and the PicoZed SoC mezzanine.

PandA Front and back panels It was designed giving access to all triggering capabilities (TTL, LVDS I/Os), communication interfaces (SFP, Gbe) and an optional FMC as show on Figure 3. The connection to the carrier board is done through 2x FFC cables. The encoders I/O as well as the auxiliary functionalities, such as USB, JTAG and serial port are implemented on the carrier board and accessible from the back-panel, as shown on Figure 4.

Figure 3: PandABox Front Panel.

Figure 4: PandABox Back Panel.

- PandA Encoder Daughter Board provides interfacing to industry standard encoders for the platform. The rack can be equipped with four boards.
- PicoZed System-On-Module is the backbone of the system. It embeds a Dual-core ARM Cortex-A9 and an Artix 7 FPGA. On one side the ARM processor runs a TCP server on an embedded Linux OS which provides the control system. On the FPGA side, encoders' positions are processed for triggering and data acquisition as detailed in Figure 7.

PandA carrier boards Inside the rack, PandA carrier board is an 8-layers PCB designed to connect all the boards together. It embeds all the power supply circuitry to regulate the voltages required for Picozed and I/Os up to 6 different voltages from 1V to 24V and powered by an external 12V. An on-board ±15V is available for FMC boards requiring this external power supply for DAC or ADC based applications.

The Slow Control FPGA has been centrally positioned on this board to facilitate routing of the functions it provides:

- Five temperature sensors distributed on the carrier board and monitored via an I2C bus as well as the separate voltage monitoring.
- PandA_Encoder daughter boards, through the control lines which configured the logic circuitry for AquadB or serial communication for the type of encoder connected to a dedicated channel. It allows Incremental, SSI, BISS or Endat encoder protocols.
- The configuration of the PandA front panel, High-Z or 50-Ohm impedance for the TTL Inputs, status LEDs. All the control is done through shift register connected in I2C.

The four encoder interfaces have been designed to allow modularity of this function. The stacked 15-way D-type connectors used for RS422 encoders are mounted on the carrier and connected to the encoder daughter board connectors. Connector's gender modification is possible since the adaptation is done on the daughter board. This approach permits each institute to maintain its pinout. Concerning the daughter boards, today it is designed with buffers to interface the different encoder types, but can be re-designed for new custom functionalities (ie. serial DAC, ADC interfacing).

A special care has been taken for routing the differential pairs and clocks, industrial standard fully-compliant Low Pin Count (LPC) FPGA Mezzanine Connector (FMC), the Gb/s Ethernet and the MGTs. These include pair matching and length tuning to the Picozed connectors. Concerning the FMC module, a specific space behind the connectors has been kept free for use of ELF specific board [2].

At the heart of the system architecture is the PicoZed System-On-Module. The PicoZed module contains Xilinx Zynq-7030 All Programmable (AP) SoC and the common functions required to support the most SoC designs, including memory, configuration, Ethernet, and clocks. It

ISBN 978-3-95450-189-2

provides easy access to over 100 user I/O pins through three I/O connectors on the rear of the module.

Prototype Validation

The first prototype is available at both institutes and fully tested. After correcting few design and manufacturing errors, electrical tests were successfully carried out. The Picozed quickly boots up and the boot sequence can be monitored through a terminal on the UART. For the slow control FPGA, during initial tests, the FPGA was configured from on-board PROM. Now, it is programmable via the Zynq MIOs, taking 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/O on the front end as well as the encoder inputs and outputs on the rear panel are fully operational.

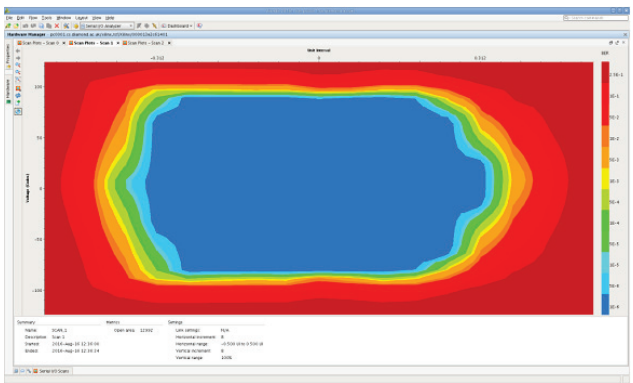

Figure 5: Eye Scan eye using with IBERT core. Line rate of 6.25Gbps with 550m fibre loopback patch.

Concerning SFP connectors, the Gigabits transceiver were tested with loopback using IBERT (Integrated Bit Error Ratio Tester) from Xilinx. This IP has been specifically implemented for testing purposes. SFP and GTX are validated at 6.25Gbps & 2.5Gbps using GTX-CLK0@125MHz with up to 550m fibre, giving acceptable eye scan result as shown on Figure 5.

A similar testing for the FMC connected were carried out using a dedicated firmware functional block. The test validates all FMC I/O, GTX connectivity and clocking pins.

FIRMWARE ARCHITECTURE

Figure 6 below depicts the system architecture for PandABox firmware and software. Linux OS running on the ARM processor manages the following peripherals:

- QSPI flash contains persistent configuration.
- SD card contains rest of Linux file system.
- Network interface to system.
- Programs FPGA, manages software interface.

The firmware is structured into numerous Functional Blocks (FBs) on the Programmable Logic (PL) part of the Zynq device, with each block configured via a number of registers. Processing System (PS) on the same Zynq device is used to realise required software architecture includes control system interfaces as shown in Figure 7.

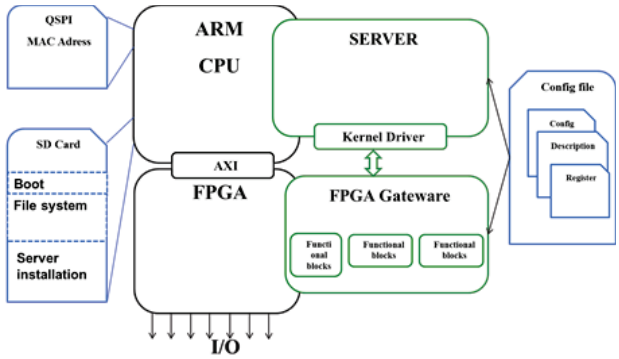

Figure 6: PandaBox overall system architecture for firmware and software.

Flexibility of the architecture is achieved through a set of Config files which describe each Functional Block's input and output ports, configuration registers and descriptions. This approach allows to design and compile a custom set of Functional Blocks into the firmware with access from the TCP Server.

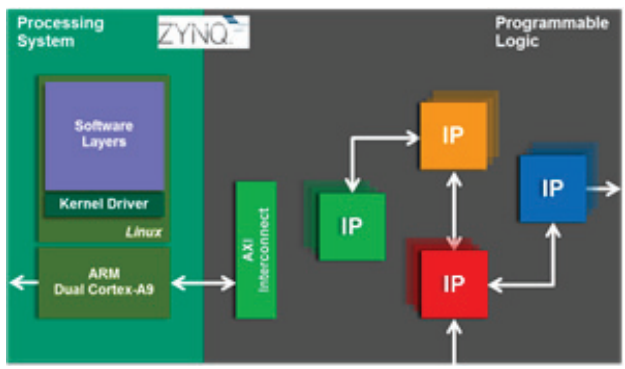

Figure 7: PandA firmware flexible architecture overview.

Functional Blocks

The list of functional blocks implemented 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 defined:

- Encoder Input/Output blocks supporting multiple encoder protocols.
- Quadrature Input/Output blocks to enable of any discrete signal on the system for quadrature operations.

ISBN 978-3-95450-189-2

- 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 and Capture

Four (4x) independent Position Compare (PCOMP) blocks are implemented to generate a stream of trigger pulses, which can be position based, time based, analog input based or externally triggered. A new feature included to the position compare functionality is the ability to use LUT-based arbitrary arrays of position or time points instead of regularly spaced series of values.

This facilitates:

- Position based capture – where output pulses are generated when the encoder position reaches a certain value, or series of values.
- Time based capture – where output pulses are generated at regular time intervals, storing the encoder positions as well.
- Analog input based capture – where output pulses are generated when analog input position reaches a certain value, or series of values.
- External trigger capture – where the motion controller stores the encoder position on an external trigger signal.

By cascading multiple position capture blocks, it is possible to mix modes together, 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.

Functional Blocks' Connectivity

To achieve connectivity between all FBs, as well as the physical inputs and outputs, two internal buses are defined. The System Bus is a 128-bit wide register is composed of concatenating all physical I/Os, and registered discrete outputs from logic and position-based blocks. Similarly, all 32-bit outputs of the position based blocks are concatenated together to create the 1024-bit wide (32x 32-bit) Position Bus.

By connecting the input of any logic block to the system bus via a 128x1-bit multiplexer, it becomes possible to cascade multiple blocks to achieve the desired functionality.

Each physical output is also taken from the system bus in the same way (see Fig. 8). Position Bus feeds all the 32-bit position inputs of all position compare blocks using 32x32-bit multiplexer so that position compare processing can be done on encoder, timer or analog input values.

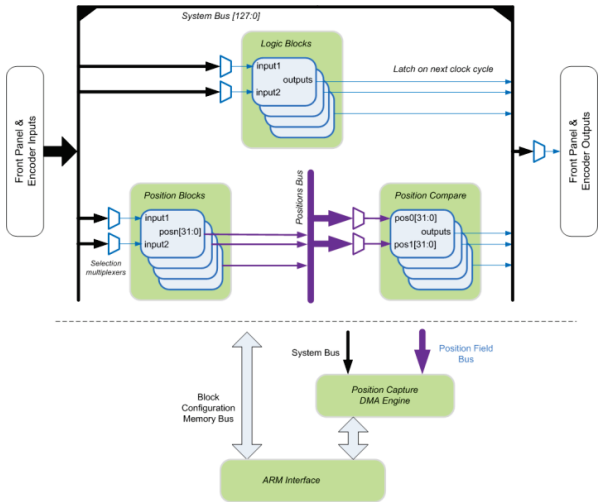

Figure 8: System and Position Bus enables full connectivity among FBs during runtime.

SOFTWARE ARCHITECTURE

Figure 9 depicts standalone software architecture, built on top of firmware specific kernel device driver, running on Zynq PS.

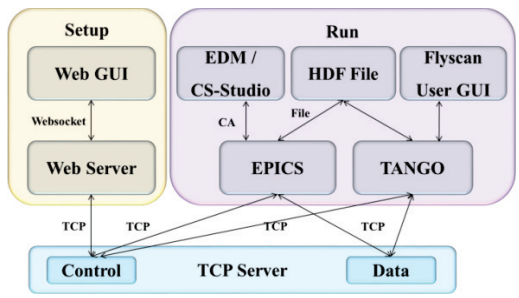

Figure 9: PandABox software architecture.

Kernel Driver

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

PandA TCP Socket Server

The Panda socket server provides a bridge between the register interface to the FPGA firmware controlling the hardware and other users software. The interface provided by this server is designed to be simple and robust. The structure of PandaBox firmware is directly reflected in the functional interface provided by this server: most commands read or write specific fields.

ISBN 978-3-95450-189-2

The socket server publishes two socket end points, one for configuration control, the other for streamed data capture. 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.

Webserver and Web GUI

Building on top of the TCP server, a webserver currently under development provides configuration level access to a PandABox via a web GUI. It is written in Python using the pymalcolm [3] framework, and exposes all blocks and their fields by introspecting what is available on the TCP server. It serves up a React [4] based user interface which communicates to the webserver using a JSON over websocket protocol. A key part of this user interface is being able to visualise and "rewire" connections between blocks, as well as setting parameters.

Once a PandABox is configured, the runtime control portion normally consists of "Arming" and then listening for data on the TCP server's data port with EPICS or TANGO software.

Figure 10: Web GUI interface showing an example application design with FBs.

EPICS and Tango

For EPICS and TANGO interface; it is envisaged to implement fixed block setup for common use cases and areaDetector driver or device server connected to DATA port to capture data and write to HDF file.

FIRST TESTS AND NEXT STEPS

First Tests

Integration of the prototype units have been under progress at both SOLEIL and DIAMOND to exercise its robustness. At Diamond, a complete hardware test setup was established in the lab including GeoBrick motion controller, X, Y and Z-stages (motors and encoders) and a PandABox under control of Mapping software framework [5]. The real-world application for Beamline I08, as shown in Figure 10, is being tested in terms of logic functionality and data capture. At SOLEIL, PandABox was successfully tested on the SOLEIL Nanoscopium beamline, replacing the SPIETBOX classically used in the Flyscan [6] architecture. A 1000x1000 continuous 2D scan was performed in a bidirectional (raster 'zigzag') mode as shown in Figure 11.

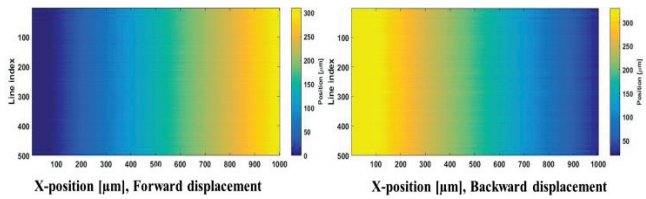

Figure 11: Plot of the 1000x1000 scan, ploting back and forth displacement on a single line, showing repetability in both direction.

Future Firmware FB Upgrades

PandA FPGA firmware architecture enables adding new FBs into the design a straight forward process. Following future FBs are defined to be added to PandA capabilities:

- A 4 Input/4 Output Channel 24V IO FMC module is currently under development, to be used to interface GeoBrick motion controller.
- Micro-Research Event Receiver FB is going to be integrated to recover timing events and clocks distributed across the event network.
- Simple, reliable, UDP/IP FB is to be implemented for triggering external instruments.

Next steps will be the deployment and commissioning of PandABox on beamlines for mapping and flyscan applications.

REFERENCES

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

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

[3] "Github: pymalcolm", `https://github.com/dls-controls/pymalcolm`.

[4] "React: A JavaScript library for building user interfaces", `https://facebook.github.io/react/`.

[5] R. D. Walton, "Mapping Developments at Diamond", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, paper THHB3O01.

[6] N. Leclercq at al., "Flyscan: a Fast, Simultaneous and Multitechnique Data Acquisition Platform for the SOLEIL Beamlines", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, paper WEPGF056.

ISBN 978-3-95450-189-2