OVERVIEW OF APPLICATIONS AND SYNERGIES OF A GENERIC FPGA-BASED BEAM DIAGNOSTICS ELECTRONICS PLATFORM AT SwissFEL

W. Koprek, B. Keil, G. Marinkovic, PSI, Villigen, Switzerland

Abstract

For SwissFEL electron beam diagnostics we combine application-specific detectors and front-end electronics with a common solution for digitization, interfacing and FPGA-based digital signal processing. Many key components and standards we use were initially developed by PSI for the European XFEL BPM system, but are equally suited for a broad range of SwissFEL diagnostics systems with little or no modifications. Examples are the FPGA signal processing hardware and firmware/software, ADC and DAC boards, interface boards or peak detection front-end electronics. By following a modular generic hardware and firmware/software design approach, we can cover a larger number of different monitor types with moderate development effort. Applications of our generic platform include BPMs, bunch length monitors, beam arrival time monitors, beam loss monitors. This paper gives an overview of the design, present and future applications of our generic platform, discussing the synergies and differences of the required hardware, firmware and embedded software solutions.

OVERVIEW

In the typical diagnostic system the sensor signals are connected to front-end electronic cards which do analogue signal processing and conditioning. The analogue signals are digitized by analogue to digital converters (ADC) cards and then processed digitally in Field-programmable Gate Array (FPGA) firmware and embedded Central Processing Units (CPU). The processed data is read by control systems over communication interfaces.

Most of the diagnostic systems have this structure therefore one can distinguish in this scheme common and dedicated components. The components may be either hardware modules like an ADC card, can be a firmware component implemented in Very high speed integrated circuit Description Language (VHDL), or software running in embedded processors. The common components - the light grey boxes in Fig. 1 – are present in most applications. The dark grey boxes are specific firmware/software components and are usually different for every application. Having in mind the similarity of various applications we focused on development of common components which can be used by the application developers.

The SwissFEL is a double bunch machine with 28 ns bunch space and 100 Hz repetition rate [1]. In this mode of operation some functions are time critical and have to

be implemented in VHDL, but most of the functions can be processed by embedded CPUs between two machine pulses within 10 ms.
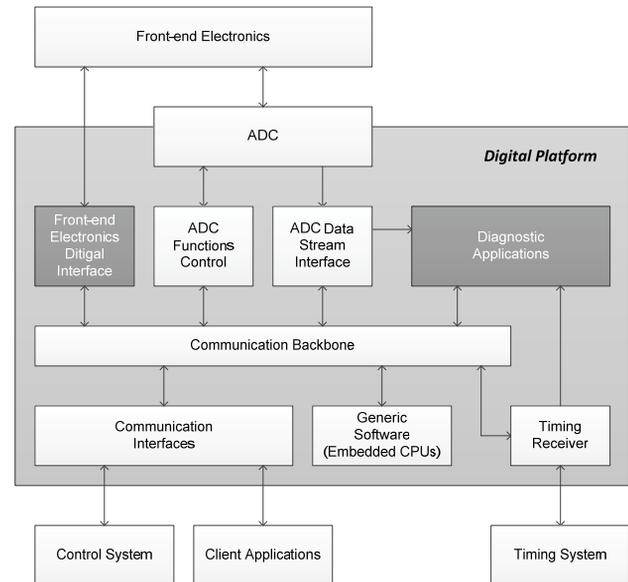

Figure 1: Block diagram of the electronics platform.

The communication backbone is built up of several bus instances and bridges between them and provides access to all components in the system which have bus interface. Communication interfaces block is a set of bridges from communication backbone to other protocols which allow access by control system or various client applications.

Synchronization of the diagnostic systems with the machine is achieved by an embedded timing receiver. This component decodes serial data stream distributed by the timing system over fibre optic links and generates synchronously local triggers and clocks.

HARDWARE PLATFORM

In order to build various systems with common components the system must be modular. Therefore our typical system consists of several hardware cards of various types like carriers, mezzanines or rear transition modules. The hardware platform is based on VERSA Module Eurocard bus (VMEbus) and has been already presented in other publications [2] and the following list gives only brief description:

- General Purpose Analog Carrier (GPAC) is a digital VME card which contains three Virtex-5 FPGA chips, three Spartan3 FPGA chips, and two 500 pin connectors for mezzanine cards. This board is used in every application for digital signal processing - see top left picture in Fig. 2.

- ADC12FL, ADC16HL – these are two mezzanine cards, one with eight 12-bit ADCs and 500 MSPS sampling rate, and the second with six 16-bit ADCs and 160 MSPS sampling rate. These two ADC cards cover all diagnostic applications - see the top right picture in Fig. 2.
- Radio Frequency Front-end Electronics (RFFE) is a set of various VME size cards where the sensor cables are connected. The cards output processed analogue signals which are connected to ADCs. Each RFFE has digital interfaces which are used to control RFFE functions from GPAC board. The specific RFFE boards are described in diagnostic applications section.
- Rear transition modules (RTM) are boards which are plugged on the rear side of the crate and have two functions. One is to provide digital interfaces between GPAC and RFFEs. The second is to extend functionality of the front board e.g. the COM RTM contains additional interfaces for GPAC.
- Modular BPM Unit (MBU) is a crate initially designed for BPM systems but it can be used as well for other diagnostic systems. This crate has no VMEbus and is meant for modular stand-alone systems - see the bottom picture in Fig. 2.
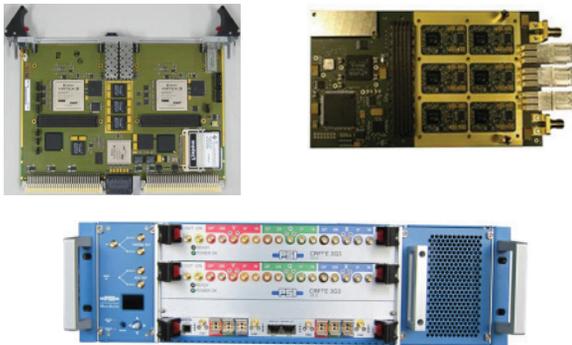

Figure 2: Hardware components.

Configuration Options

Based on the above described hardware components one can build a diagnostic system in two major configurations. One configuration is built with a VME crate and the other with an MBU.

VME Based Configuration The VME crate is able to host several digital cards with VMEbus interface and allows building complete system including the control system as presented in Fig. 3. In the VME crate one can install front-end electronics communicating with GPAC over special transition cards (GPAC/RFFE RTM) deploying user pins from the VME P2 connector. In the same crate the CPU with control software and timing module is installed. The CPU communicates with GPAC over VMEbus. Depending on number of available slots in the VME crate it is possible

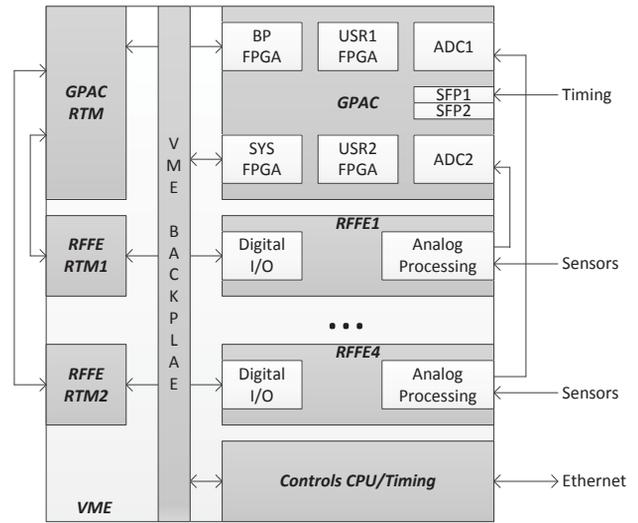to have several GPAC boards with corresponding RFFE cards in the same crate.

Figure 3: VME based hardware configuration.

MBU Based Configuration In this configuration the control system is not included. The MBU contains only a single GPAC and up to four RFFEs as illustrated in Fig. 4. The MBU contains customized backplane without VME bus since there is only one card with VMEbus interface. The rear transition modules from the VME configuration are replaced by direct connections in the MBU backplane. A dedicated communication RTM board (COM RTM) extends the GPAC interfaces implemented in the VME connectors. The SFPs of GPAC and COM RTM provide physical interfaces for communication protocols between MBU and control systems placed somewhere else in the machine. This solution is preferred for small systems physically distributed along the machine such as BPMs. Many MBUs can be connected to a single Controls CPU by means of Ethernet or fibre optic links. The number of MBUs connected to a single CPU depends on its computation power and it can be easily balanced among many CPUs by reconnecting the MBU.
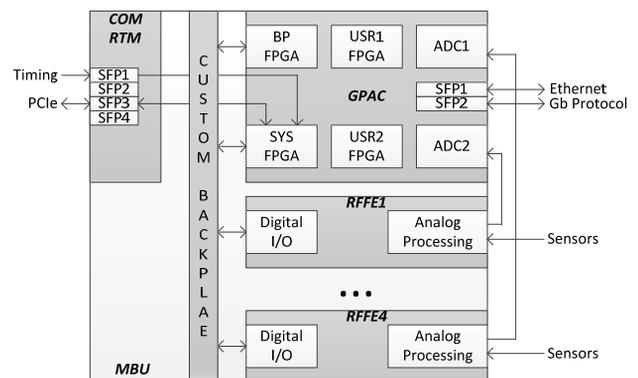

Figure 4: Hardware installation in an MBU.

GENERIC FIRMWARE AND SOFTWARE

Besides the common hardware components there is also a firmware and software platform which is common for many diagnostics applications and can be used as a base in the diagnostic projects. The following sections describe common firmware and software solutions which are used in various diagnostic applications.

Communication Backbone and Interfaces

The GPAC board has Xilinx Virtex-5 and Spartan3 FPGA chips. Virtex-5 has built-in CPU PowerPC 440, and Spartan3 has synthesized CPU called uBlaze. Both processor types use Processor Local Bus (PLB) for communication. The base firmware configuration of GPAC has three Virtex-5 chips with PowerPC, and two Spartan3 with uBlaze, and the user applications are distributed over those FPGA chips and CPUs. Therefore it was necessary to build a kind of infrastructure which allows communication between all CPU subsystems and the control system. Fig. 5 presents the block diagram of the communication infrastructure.
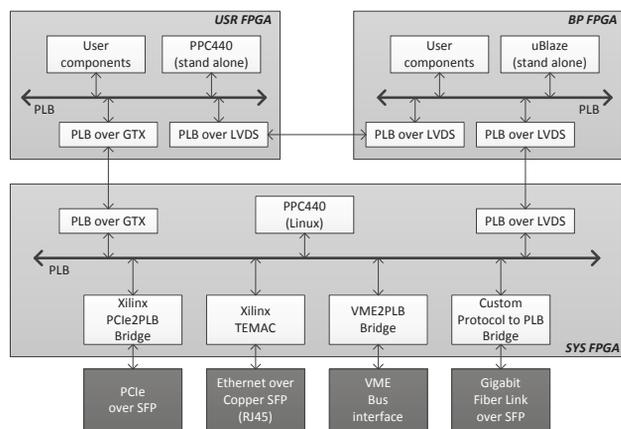

Figure 5: Communication components.

The PLBs are connected to each other by PLB to PLB bridges. The PLB bridge maps part of the PLB address space in remote FPGA to address space of the local PLB based system. The bridge is transparent which means that the PLB Master reads an address on local PLB and the bridge forwards the read transaction to remote PLB, the remote PLB master executes the transaction and sends back the data and the local PLB slave completes the transaction. From PLB Master point of view the difference between local and remote transactions is only in latency. The PLB bridge is implemented in two versions: PLB over GTX and PLB over LVDS. They differ in physical layer. Two Virtex-5 chips are connected by RocketIO (a gigabit serial link called GTX in Virtex-5) and can run with baud rate up to 5 Gbps. Spartan3 has no RocketIO, therefore the link is built on LVDS lines and is scalable depending on the number of available LVDS lines between two FPGA chips.

The communication with external systems is determined by available communication interfaces and

protocols in those systems. There are four interfaces currently implemented in GPAC: VME slave, PCIe endpoint, 1Gb Ethernet, and PLB over GTX bridge to another system with PLB. The VME, PCIe, and PLB over GTX have PLB master functionality which means that the external system can directly access memory locations in GPAC. The Ethernet interface is used to communicate from client applications by means of TCP/IP protocols with applications running on Linux system in GPAC.

Timing Receiver for SwissFEL

The timing system for SwissFEL consists of a single event generator (EVG) and several event receivers (EVR) connected by fibre optic links [3]. The timing system distributes several events, event clock, and machine status information such as beam on/off, pulse number, charge, etc. The hardware of the timing system is based on the VME standard. The EVR is usually installed in the VME crate next to the controls CPU. But in case of hardware configuration with MBU there is no place for EVR card. Therefore the GPAC board contains a kind of embedded event receiver implemented in FPGA. The block diagram of the receiver is presented in Fig. 6.
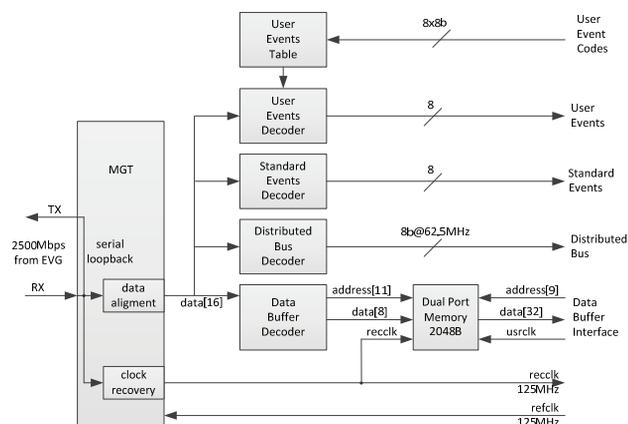

Figure 6: SwissFEL embedded timing receiver.

The fibre link from the timing system network is connected to one of the SFPs on the COM RTM and the data stream is decoded in the FPGA. The firmware allows generation of up to eight triggers from user defined events and it contains a data buffer where the machine status data is received and kept for user applications. The standard events decoder and distributed bus decoder are also implemented according to the specification of the timing system and can be used by the diagnostic applications.

Generic Software

The generic software is implemented in PowerPC in SYS FPGA. The PowerPC is running PetaLinux – this is a Xilinx version of Linux for embedded systems – which was adapted for the GPAC hardware. The Linux system is an open platform for any kind of generic software as well as for user extensions. In PetaLinux environment one can write a new program, cross-compile, and upload it to

GPAC and run it without disturbing the running system. Among various kind of generic software installed on Linux these are the two most important ones for the application developers: firmware maintenance software and web server.

Firmware Maintenance Software   The maintenance software gives possibility to run on Linux several programs to test the system status, read and write any address in the system, and perform remote update of the firmware. The secure copy protocol (SCP) server running on GPAC gives access to file system on a compact flash (CF) card with file system mounted to Linux where configuration files are stored. It is possible to update remotely configuration of all FPGA chips on the GPAC board as well as the Linux image. The firmware update strategy was implemented to keep the system always running and available remotely. It is possible to have several configuration versions and the boot firmware checks if the recently uploaded firmware works and makes the GPAC accessible from outside. If not, the booting firmware loads so-called golden image which works always and gives possibility to upload another, working version of the user firmware.

Web Server The quick way of getting access to the common firmware running on GPAC can be achieved by a web browser. A web server was implemented on the GPAC board. It starts automatically and it has a set of web pages which allows operation of the ADC scope application. The web server is based on open source GoAhead web server. The web server attaches to every requested web page a special library written in JavaScript called WebMAP. This library is responsible for continuous communication of the webpage on the client side with the web server. This library continuously retrieves data from the web server and updates the local web page. An example of such web page is presented in Fig. 7. The screenshot presents a set of control fields with timing and triggers parameters as well as waveforms from four ADC channels.

Figure 7: Web page for ADC scope.

The WebMAP library allows building new webpages for user applications by creating new Hyper Text Markup Language (HTML) files without changing the web server. Data embedding in the HTML is done by special use of

HTML tags property *title*. This property contains hardware address, data type, and other parameters used by the WebMAP library to access and format displayed data from GPAC. After the web page is loaded by the web browser, the WebMAP library automatically finds tags with defined *title* properties, parses the *title* strings, and performs periodic data update of their content. The new or modified HTML files only have to be uploaded to GPAC CF card by file transfer protocol and can be applied without disturbing the running system.

ADC Based Oscilloscope

Each of the diagnostic applications has analogue signals connected to one of the two types of ADC mezzanine cards. The ADC data stream can be processed directly by VHDL firmware or can be stored in local memory and used for post processing either in embedded CPU or in control system. The ADC based oscilloscope records ADC data and provides interfaces to user applications as presented in Fig. 8.

The parallel data lines coming from ADC have to be aligned in the FPGA to a single clock in order to process them synchronously. Then the data is buffered in FIFOs which keep past samples in order to see what happened before trigger conditions occur. Then the buffered samples are gated by the trigger control component. The trigger can work in auto mode, single mode, or with external trigger. The memory can store up to 4 k samples per channel. This is dual port memory which allows separation of the ADC clock domain from the local clock.
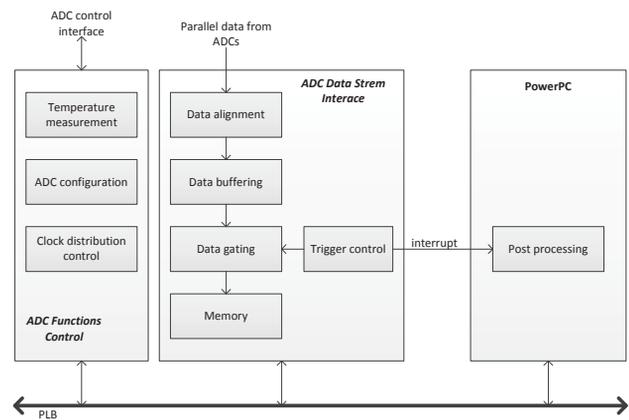

Figure 8: ADC Scope firmware block diagram.

The ADC Functions Control component is used to interface various components on the ADC mezzanines. The ADC cards have configurable clock distribution circuit where one can select local or external clock, set individual clock delays and division factors. The ADC12FL card can also synthesize in phased-lock loop (PLL) different sampling frequencies. The component also provides configuration of the ADC settings. The post processing in PowerPC is a piece of software which calculates basic parameters of the recorded wave forms such as minimum, maximum, mean value, standard deviation, etc.

DIGITAL PLATFORM APPLICATIONS

The above described digital platform is a base for application specific extensions. If the diagnostic application deploys specific front-end electronics, the developer has to prepare in firmware RFFE specific control components. The firmware library of the digital platform already provides basic VHDL components for typically implemented interfaces such as I^2C, UART, SPI, and 1-Wire. In next step the developer implements application specific ADC data processing. Depending on the latency requirements this can be done either in VHDL or in software in PowerPC. The following applications are based on the GPAC digital platform.

Beam Position Monitors The BPMs for SwissFEL have dedicated RFFEs for two types of cavity pickups for linac and undulators [5]. The data is sampled with 16-bit ADCs. The ADC data is processed directly in FPGA and is supported by pulse to pulse feedbacks implemented in PowerPC [6]. The digital platform was also successfully used for implementation of BPM systems for European XFEL and FLASH accelerator at DESY. The Cavity BPM systems have been running in FLASH since one year [7] and the first Button BPM systems were tested at FLASH and now they are running in XFEL [8].

Beam Loss Monitors The BLMs use a kind of generic RFFE called PAC. The PAC has several connectors for mezzanine cards with analogue circuits for signal conditioning. The BLMs use 12-bit ADCs and the data processing is also done in VHDL and is supported by PowerPC software [9].

Bunch Compression Monitors The BCMs are used in bunch compressors to measure the quality of bunch compression [10]. It deploys specific RFFE for peak detection. The data is sampled by 12-bit ADCs. Due to relaxed condition for latency, the whole processing will be implemented in software in PowerPC. In this case the system will use the external PLB over GTX bridge to communicate with BPM systems in order to normalize the compression measurement.

Bunch Arrival Monitors The bunch arrival monitors have dedicated front-end electronics [11]. Currently it only uses the GPAC platform for sampling signals with 12-bit ADC and the post processing will be implemented in future.

CONCLUSION

The FPGA based digital platform besides common hardware has common firmware and software platform for all diagnostics applications. Deploying of the common firmware/software as a base for specific diagnostics applications reduces significantly the development effort. All the elements have been already tested in some applications. Nevertheless there are still computation resources available for further extensions if necessary.

The next version of GPAC board based on seven series of Xilinx FPGA chips is under development. The new GPAC board will be backward compatible with other hardware components. It will increase the life time and maintainability of existing and future applications by another 15 year.

REFERENCES

[1] Home page of SwissFEL project: http://www.psi.ch/swissfel/

[2] B. Keil et al., "A Generic BPM Electronics Platform for European XFEL, SwissFEL and SLS", IBIC'12, Tsukuba, Japan (2012).

[3] B. Kalantari, "SwissFEL Timing System", presentation given at MRF Workshop: http://www.mrf.fi/index.php/timing-workshop Prague (2014).

[4] Embedded Web Server GoAhead: https://embedthis.com/goahead/

[5] M. Stadler et al., "Development of the SwissFEL Undulator BPM System", IBIC'14, Monterey, USA (2014).

[6] B. Keil et al., "Design of the SwissFEL BPM System", IBIC'13, Oxford, UK (2013).

[7] M. Stadler et al., "Beam Test Results of Undulator Cavity BPM Elecdtronics for the European XFEL", IBIC'12, Tsukuba, Japan (2012).

[8] D.M. Treyer et al., "Design and Beam Test Results of Button BPMs for the European XFEL", IBIC'13, Oxford, UK (2013).

[9] C. Ozkan Loch et. al. "System Integration of SwissFEL Beam Loss Monitors", MOPB051, these proceedings, IBIC'15, Melbourne, Australia (2015).

[10] F. Frei et al., "Development of Electron Bunch Compression Monitors for SwissFEL", IBIC'13, Oxford, UK (2013).

[11] V. Arsov et al., "First Results from the Bunch Arrival-Time Monitor at the SwissFEL Test Injector", IBIC'13, Oxford, UK (2013).