PROCESSING HIGH-BANDWIDTH BUNCH-BY-BUNCH OBSERVATION DATA FROM THE RF AND TRANSVERSE DAMPER SYSTEMS OF THE LHC

M. Ojeda, P. Baudrenghien, A.C. Butterworth, J. Galindo, W. Hofle,
T.E. Levens, J.C. Molendijk, D. Valuch, CERN, Geneva, Switzerland
F. Vaga, University of Pavia, Pavia, Italy

Abstract

The radiofrequency and transverse damper feedback systems of the Large Hadron Collider digitize beam phase and position measurements at the bunch repetition rate of 40 MHz. Embedded memory buffers allow a few milliseconds of full rate bunch-by-bunch data to be retrieved over the VME bus for diagnostic purposes, but experience during LHC Run I has shown that for beam studies much longer data records are desirable. A new "observation box" diagnostic system is being developed which parasitically captures data streamed directly out of the feedback hardware into a Linux server through an optical fiber link, and permits processing and buffering of full rate data for around one minute. The system will be connected to an LHC-wide trigger network for detection of beam instabilities, which allows efficient capture of signals from the onset of beam instability events. The data will be made available for analysis by client applications through interfaces which are exposed as standard equipment devices within CERN's controls framework. It is also foreseen to perform online Fourier analysis of transverse position data inside the observation box using GPUs with the aim of extracting betatron tune signals.

SYSTEM OVERVIEW

The "Observation Box" (ObsBox) is a custom acquisition system designed to overcome the bandwidth limits of the VME bus to capture diagnostic data. Figure 1 shows an overview of the system.

The transmitters shown in the figure are typically custom-built VME modules in the beam feedback systems with embedded FPGAs that sample different aspects of the accelerator's beam at a high frequency. In many cases, the acquisition of these data streams for diagnostic purposes can be done through the VME bus into the Linux host residing in the crate. However, some analyses of the beam data can really benefit from a substantially higher data bandwidth than the VME bus can provide.

Typically, the transmitters are close to the beam, in the underground caverns. For accessibility and maintenance reasons, the receiver, a Linux server, is installed on the surface, with a fiber link running between the two.

The new transmitters designed for the ObsBox system stream the data out through the fiber links. At the server end, for each fiber, a CERN-designed PCI Express v1 module is used to act as the receiver: the Simple PCI Express Carrier (SPEC) module [1].

The ObsBox server runs a customized Linux kernel intended for hosting real-time software. A tailored Linux kernel device driver is used to control the SPEC modules over the PCIe bus and is able to stream the data to a user-space process through syscalls and a *sysfs* interface.

The user-space process is responsible for reading the driver streamed data regularly and quickly enough and storing it into very large memory buffers, where the data waits to be requested eventually by clients. Clients connect to the ObsBox server over the CERN's technical network, a standard Ethernet network.

The user-space process is also foreseen to perform online data analysis if the network bandwidth is not enough to stream it out to clients. Since at this stage the data is local, using a coprocessor (like a GPU) to perform the analysis is a viable option, which in turn opens the possibility of running other kinds of analysis for which the CPU may not be performant enough.

TRANSMITTER (VME MODULE)

The Longitudinal Position Measurement VME module (LPM) comprises an analogue front-end and a digital signal processing FPGA platform generating digitized information that can be transmitted via fast digital serial links for further processing. The hardware is based on the beam phase measurement module originally developed for the LHC RF Beam Control [2].

Two signals are acquired in the RF front-end: the beam pickup and the 400.8 MHz RF voltage vector sum of eight cavities. These are demodulated into analog I/Q signals and bunch synchronously sampled by four 14-bit AD converters to give I/Q pairs at a data rate of 40.08 MHz. In the FPGA platform, the VME card is equipped with four gigabit links. The bunch synchronous data of the two I/Q signals (at a rate of 40.08 MHz) is "repackaged" into 50 MHz serial data frames, computing 1 Gbps per link. A FIFO synchronization approach has been used to handle the data crossing among clock domains.

The four data streams are sent over a copper to optical converter card towards the ObsBox. At the receiver side, the four optical links are resynchronized again with 40.08 MHz and the revolution frequency clock. Signal generators are also available within the FPGA platform enabling digital calibration and link debugging.

Similiarly, the LHC transverse damper system (ADT) uses two or four pickups per plane per beam to measure beam position at 40.08 Msamples/s, digitized in a VME beam

ISBN 978-3-95450-148-9

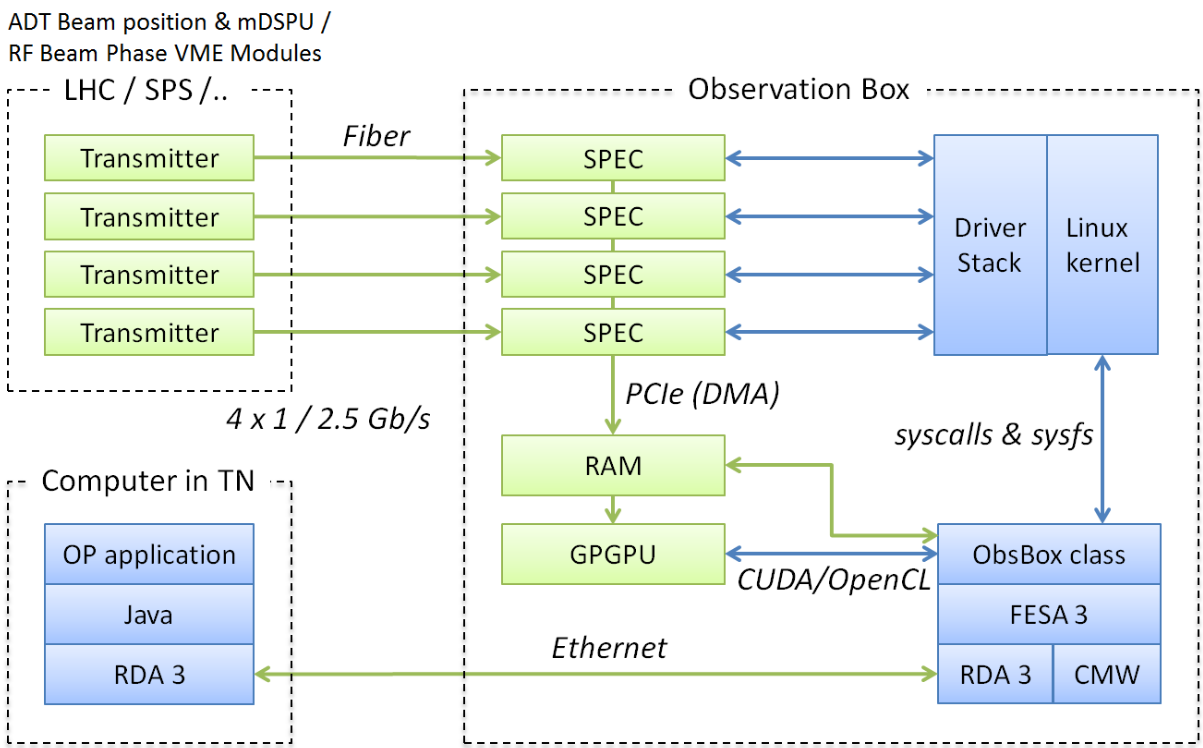

Figure 1: System overview.

position module (BPOS) and streamed to a processing VME module (mDSPU) [3]. Both modules have gigabit links allowing data transfer towards the ObBox using the same format as the LPM..

FIBER LINK

Table 1 describes the data structure transmitted to the receiver over the fiber link in the LHC case.

Since the communication is unidirectional and the receiver has no knowledge on when the transmitter started a block of data (representing one LHC turn), the software in the receiver has to have a way to synchronize with the blocks (frames) being transmitted. In order to do this, each block starts with a fixed constant. On startup or error recovery, the receiver scans the data stream until it recognizes the fixed constant in the stream. It can then acquire the rest of the block and validate its correctness using the CRC. If by chance the fixed constant appeared in the data stream in another position, the CRC will not match, with very high probability. Additionally, other fields can be checked to validate further the block (e.g. checking the block size or the source ID). In case of a CRC-matching failure, the receiver can continue scanning the stream.

The current production system runs each fiber link at 1 Gbps, while it has been designed with a final target speed of 2.5 Gbps, currently in development.

In the deployed LHC use case, the actual payload data rate (after discounting encoding, link synchronization and the data structure overhead), is of approximately 0.6 Gpbs per link. Currently, the production system is running with 4 fiber links per server.

Table 1: Link Data Structure (LHC)

Field	Size [bit]	Description
Fixed constant	32	Arbitrary constant used to identify the beginning of a turn within the data stream
Source ID	32	An identifier of the data being received
Block size	32	Size of the data structure
Turn counter	32	Monotonically increasing, wrapping counter
Tag bits	32	Flags to describe events associated to this turn, e.g. beam instability detected
Reserved	$3 \cdot 32$	Reserved for future use.
Data	$3564 \cdot 16$	Actual payload, which can be arbitrary. In the LHC case, the data is structured in 16-bit words, one per bunch
CRC	32	Used for error detection

ISBN 978-3-95450-148-9

RECEIVER (LINUX SERVER)

Server

The server to host the ObsBox software and receivers has to be carefully considered. There are several requirements that limit the selection of suitable models from manufacturers.

The primary goal of the ObsBox system is to increase the amount of data available to analysis for end users without a priori filtering or decimation, both in terms of data rate (bunch-by-bunch) and in terms of total time buffered (minutes in the new system compared with milliseconds in the old). Given the 0.6 Gbps of actual payload data per channel, and considering 4 channels per server, the server needs to hold a considerable amount of memory. The current servers used are able to hold up to 6 minutes of bunch-by-bunch data for 4 different channels in a single server, requiring 128 GiB of RAM.

It is desirable to minimize the number of servers due to constraints on the available rack space and other infrastructure considerations. Therefore, each server should be able to manage as many fiber links as possible. This limits the number of motherboards (number of PCI Express slots) and CPUs (processing power) that can be selected.

With these requirements in mind, the current system was deployed using SuperMicro's SuperServer 6028U-TR4+ servers (shown in Figure 2), which have 6 available PCIe slots including one x16 graphics card port.

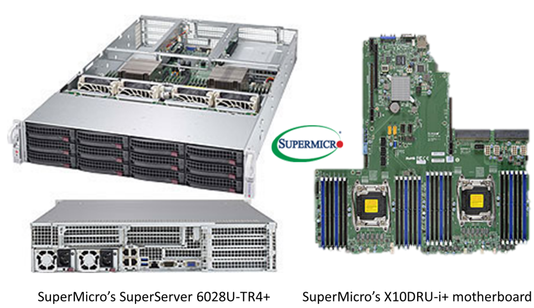

Figure 2: SuperMicro's chassis and motherboard.

SPEC PCIe Module

The module responsible to be the bridge between the optical fiber link and the memory of the Linux host is a codeveloped project between CERN BE-CO and BE-RF groups.

The Simple PCIe FMC carrier module [1] (shown in Figure 3) is a project from the Open Hardware Repository [4], developed by the CERN BE-CO group. It is a simple 4-lane PCIe carrier for a low pin count FPGA Mezzanine Card (VITA 57).

The SPEC's small form-factor pluggable (SFP) receptacle is used to house a fiber optic receiver and a customized FPGA firmware was developed by the CERN RF and Beam Instrumentation groups.

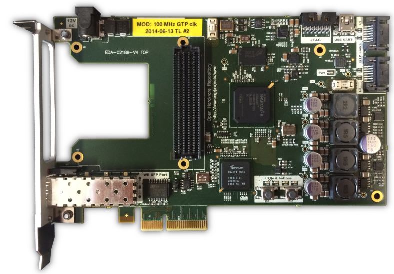

Figure 3: Simple PCIe FMC carrier (SPEC).

Driver

The kernel driver is based on the ZIO framework [5, 6]. ZIO provides a solid framework that simplifies the development of data acquisition drivers. The framework defines the input/output data flows and an user-space interface to access the data. The framework provides buffers, triggers, Linux kernel DMA configuration and it eases the declaration of *sysfs* attributes.

Data access from user-space is completely handled by the framework through char-devices with *mmap* support. Within the ZIO framework, a driver has only to provide support for the hardware access since most of the software logic is done by the framework. In the ObsBox case, the driver is in charge of setting the DMA registers each time some data is ready to be transferred and providing access to the hardware registers for configuration purposes. Everything else is handled by the framework.

User-space Process

The user-space process is the bridge between the clients of the data which will perform the analysis on it and the rest of the system. There are many requirements that the user-space process has to cope with at the same time:

1. Clients may request any recent data at any moment, with different possible filtering criteria. Therefore, the server cannot do any filtering a priori and, thus has to hold all the data until it is considered to have expired. Typically, clients are interested in only a fraction of the streamed data (e.g. selection by bunch, or by time).
2. Depending on the use case, the buffers may be constantly recording new data, or they may start or stop on some events (e.g. beam instability). In addition, the process may have to react to events coming from the CERN timing network.
3. The fact that the driver can only hold a tiny fraction of data in kernel-space before it needs to be released also poses a real-time requirement on the user-space process. The process must regularly and within some deadline read the data from all the channels (devices) from the driver.
4. The process has to expose the data to end users through a typical Ethernet network. The CERN controls framework [7] establishes a standard way of defining, config-

ISBN 978-3-95450-148-9

uring and controlling all the devices in the accelerator. The communication layer is developed within CERN's Controls Middleware [8].
5. While the currently deployed system is not strictly critical for the beam in the accelerator, availability of the diagnostics may be critical for operators in some scenarios. Therefore, the system has to be robust with high availability.

In order to satisfy all these requirements while keeping the software as simple as possible, the user-space process takes advantage of the Front-End Software Architecture (FESA) framework [9], which provides a comprehensive environment for equipment specialists to design, develop, test and deploy real-time control software for front-end computers.

USE CASES

Longitudinal Diagnostics

The initial use case of the ObsBox for longitudinal diagnostics is the observation of the phase of individual bunches. This is essential to study coupled-bunch instability threshold, and to identify such instabilities if they appear. From the bunch phase, one can also compute the power lost by the bunch. At high beam current the LHC suffers from e-cloud build-up, which can be studied and monitored from the bunch phase measurements [10]. Compared to the previous system which was able to acquire only 73 turns worth of data, the precision of both these measurements will be largely improved due to the very large number of turns now available.

Transverse Dampers

For the transverse dampers in SPS and LHC, habitually called "ADT" in the LHC, it is foreseen to use the ObsBox to record and be able to make triggered acquisitions both of processed pick-up signals and the raw beam position signals.

Both VME modules in the ADT system, the BPOS and the mDSPU, have internal memory which is insufficient to store large amounts of data as needed for injection damping analysis, instability or tune diagnostics. Eventually beam pick-up position, feedback signals and headtail oscillation index [11] can be made available. Due to the large amount of data generated by the system, an intelligent trigger system [12] is also foreseen to freeze long records of data, either on demand, at injection, or automatically, in case of instability detected on the beam. Online data processing is also foreseen to extract machine parameters such as the tune [13], with the possibility to add a GPU co-processor. In a first stage, position information from four pick-up signals (BPMCQ7 in point 4 of LHC [3] of both transverse planes) is being recorded by a single server unit. Following experience with this setup a decision will be taken on adding more signals from the set of currently 40 envisaged. With the recently completed upgrade of the SPS damper to the digital VME standard of LHC, it is also foreseen to implement an ObsBox for the transverse plane in the SPS.

CONCLUSION

The new custom acquisition system is already delivering improved resolution in diagnostics for the LHC accelerator compared to the previous system and is on its way to be used in other accelerators and projects as well.

There are two major milestones in the near future for the project. The first one is to achieve the 2.5 Gpbs target speed on each fiber link, which will make the system even more useful for several projects and with less infrastructure requirements (rack space, fiber installations, etc.).

The second milestone is to compute some expensive analysis in the server itself, using a coprocessor (e.g. GPU). This will open a broad new set of possibilities regarding online analysis which may prove useful for the Operations and Physics teams in the department.

REFERENCES

[1] Simple PCIe FMC carrier (SPEC) website: `http://www.ohwr.org/projects/spec`

[2] P. Baudrenghien et al., "The LHC Low Level RF", EPAC'06, Edinburgh, Scotland, June 2006, TUPCH195 (2006), `http://www.JACoW.org`

[3] D. Valuch et al., "LHC transverse feedback", 5th Evian Workshop on LHC beam operation, Evian-les-Bains, France, 2014

[4] Open Hardware Repository (OHWR) website: `http://www.ohwr.org`

[5] A. Rubini et al., "ZIO: The Ultimate Linux Kernel I/O Framework", ICALEPCS'13, San Francisco, October 2013, MOPB026 (2013), `http://www.JACoW.org`

[6] ZIO Linux device drivers framework website: `http://www.ohwr.org/projects/zio`

[7] Z. Zaharieva et al., "Database Foundation for the Configuration Management of the CERN Accelerator Controls Systems", ICALEPCS'11, Grenoble, France, October 2011, MOMAU004 (2011), `http://www.JACoW.org`

[8] M. Arruat et al., "Middleware Trends and Market Leaders 2011", ICALEPCS'11, Grenoble, France, October 2011, FRBHMULT05 (2011), `http://www.JACoW.org`

[9] M. Arruat et al., "Front-End Software Architecture", ICALEPCS'07, Tennessee, USA, October 2007, WOPA04 (2007), `http://www.JACoW.org`

[10] J. F. Esteban Müller et al., "High-accuracy diagnostic tool for electron cloud observation in the LHC based on synchronous phase measurements", PRST-AB, accepted for publication

[11] W. Hofle et al., "Transverse rigid dipole and intra-bunch oscillation detection using the transverse feedback beam position detection scheme in SPS and LHC", IBIC2015, Melbourne, Australia, 2015

[12] T. Wlostowski et al., "Trigger and RF Distribution Using White Rabbit", WEC3O01, these proceedings, ICALEPCS'15, Melbourne, Australia, 2015

[13] F. Dubouchet et al., "Tune measurement from transverse feedback signals in LHC", IBIC2013, Oxford, UK, 2013

ISBN 978-3-95450-148-9