

A NEW FAST DATA LOGGER AND VIEWER AT DIAMOND: THE FA ARCHIVER

M.G. Abbott, G. Rehm, I.S. Uzun, Diamond Light Source, Oxfordshire, UK

Abstract

At Diamond Light Source, position data from 174 Electron Beam Position Monitors (BPMs) and a number of X-Ray BPMs is distributed over the Fast Acquisition communications network at an update rate of 10 kHz; the total aggregate data rate is around 15 MB/s. The data logger described here (the FA Archiver) captures this entire data stream to disk in real time, re-broadcasts selected subsets of the live stream on demand to interested clients, and allows rapid access to any part of the saved data. The archive is saved into a rolling buffer allowing retrieval of detailed beam position data from any time in the last four days. A simple socket-based interface to the FA Archiver allows easy access to both stored and live data from a variety of clients. Clients include a graphical viewer for visualising the motion or spectrum of a single BPM in real time, a command line tool for retrieving any part of the stored data by time of day, and Matlab scripts for exploring the dataset, helped by the storage of decimated minimum, maximum, mean, and standard deviation data.

INTRODUCTION

Diamond uses Electron Beam Position Monitors for electron beam diagnostics and orbit position feedback. Each BPM button block is connected to a Libera [1] beam position processor, comprising RF processing for each of four pickup buttons, with ADCs and an FPGA which reduces the data to streams at a variety of data rates, and an embedded processor providing an EPICS interface [2].

In the Libera FPGA the sampled button intensities are converted to X, Y positions and reduced by decimation and filtering to machine revolution frequency (534 kHz) and then to the “Fast Acquisition” (FA) data rate of 10 kHz and the “Slow Acquisition” rate of 10 Hz for EPICS clients. The FA data is broadcast through the Diamond Communication Controller (CC) network [3, 4] which makes this data stream simultaneously available to all connected machines. The CC network allows machines to be freely added as receivers or transmitters with little or no impact on the operation of the network; the FA Archiver is one such machine.

The FA archiver receives CC updates at the FA data rate of 10 kHz, each update consisting of position information for all connected BPMs together with data from a handful of other data sources. This data is stored to disk in a rolling multi-day buffer and made available to client applications throughout Diamond.

The archiver runs on a dedicated server and writes to local disks; at Diamond we use a 6 Terabyte archive (two dedicated 3 TB disks as a 6 TB RAID 0 volume) giving us just under 4½ days of archive of 182 CC ids. The archiver acts as a socket server supporting a simple command language; commands are provided for subscribing to any subset of the live data stream, and for retrieving any part of the archived dataset.

Live data is available from the FA archiver both at the full 10 kHz data rate and decimated by filtering down to 1 kHz. Archived data is available at the full data rate or decimated binned by factors of 128 or 16384 (by default) to provide an overview of the stored data.

FA ARCHIVER AND CC NETWORK

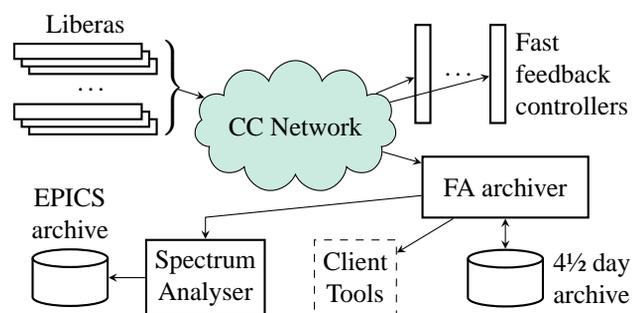


Figure 1: FA Archiver and CC network in context.

The Communication Controller network was originally constructed to make the 10 kHz FA data from each Libera BPM available for the operation of the fast feedback controllers which maintain position stability of the electron beam. The network synchronously distributes beam position X and Y coordinates every 100 μ s with a propagation delay of around 50 μ s.

The design of the CC network with a highly redundant topology broadcasting by storing and forwarding makes it easy to add new nodes to the network, both as contributors and as passive listeners. Other sources contributing to the network are a handful of X-ray BPMs and power pick-ups from the RF cavities. The FA archiver acts as a passive listener and makes the FA data freely available, see Fig. 1.

The archiver stores data at the full data rate, referred to here as the “FA” rate, and at two decimated rates, “D” data decimated by a factor of 128, and “DD” “double decimated” data decimated by a factor of 16384.

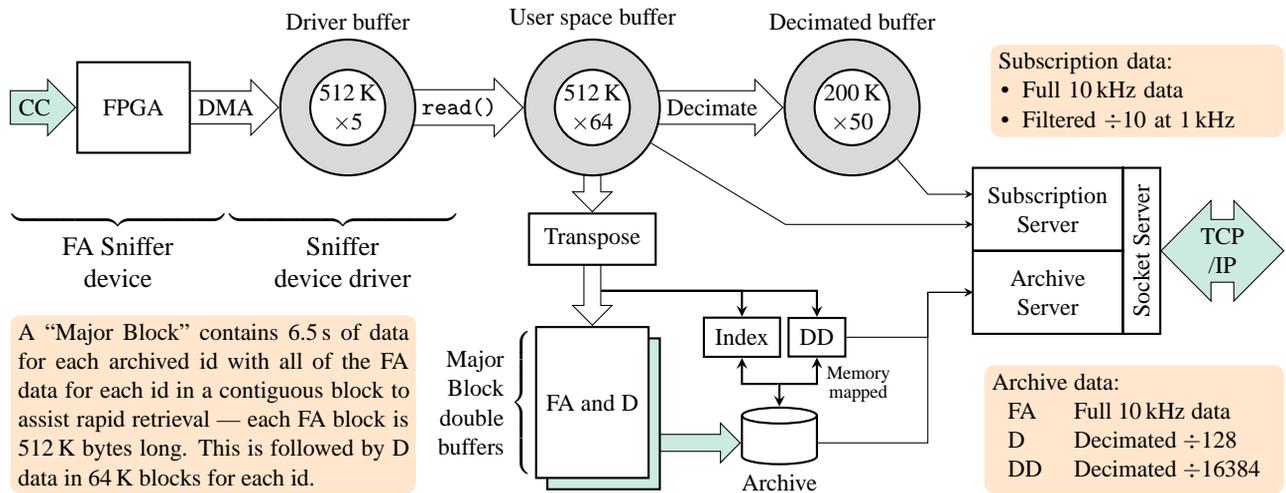


Figure 2: Architecture of FA archiver showing device, driver and archiver process writing to disk and serving clients.

ARCHIVER ARCHITECTURE

The FA archiver consists of the following components:

FA Sniffer. This is an implementation of the Communication Controller on a Virtex-5 FPGA PCI Express (PCIe) board. The sniffer captures communication frames and transfers them by DMA to memory.

Sniffer device driver. The device driver communicates with the sniffer card and makes the data stream available as a Linux device node, `/dev/fa_sniffer0`.

Archiver process. The archiver process reads frames from the device driver, saves them to disk after processing to optimise subsequent reading, and provides a socket server for client access to both the live data stream and a $\div 10$ decimated stream, see Fig. 2.

Client tools. A variety of client tools are able to connect directly to the archiver server, including a data visualisation tool (`fa-viewer`) for viewing the live data, a general purpose command line tool (`fa-capture`) for reading from the archive, and a Matlab script (`fa_zoomer`) for exploring the archive.

FA Sniffer

The FA Sniffer is so called because it passively “sniffs” FA data from the communication controller network. The FPGA design integrates the Diamond Communication Controller [3] into a bus master PCI Express architecture using the commercially available Xilinx Virtex-5 FPGA ML555 development platform [5]. The design, see Fig. 3, consists of target logic, DMA initiator logic, status/control registers and the Virtex-5 endpoint core for PCI Express. Target logic is responsible for capturing single double-word memory write and memory read PCIe Transaction Layer Packets (TLPs) for control and status register access. The DMA initiator logic generates memory write TLPs to transfer 2 K byte frame data from the communication

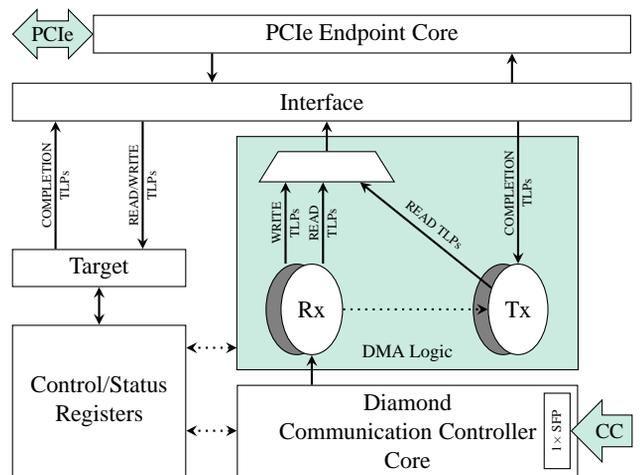


Figure 3: Architecture of the FA sniffer FPGA.

controller core to the host’s system memory. The complete design occupies 30 % of bit slices and 20 % of block RAMs available on the FPGA.

The communication controller core captures a complete frame from the CC network on every communication controller cycle, with a new cycle starting every 100 μ s. The frame transferred to memory by the sniffer is 2048 bytes consisting of 256 samples, each a pair of four-byte numbers corresponding to measured X and Y positions for FA ids 1–255 together with a 32-bit timestamp counter repeated in position 0 as “id 0”.

The sniffer card maintains a queue of up to two DMA targets, address and frame count, one being written to, the next to be loaded when the first target is filled, at which point an interrupt is raised to the host machine. At this point the device driver is responsible for setting up a fresh buffered DMA target, otherwise the sniffer will halt.

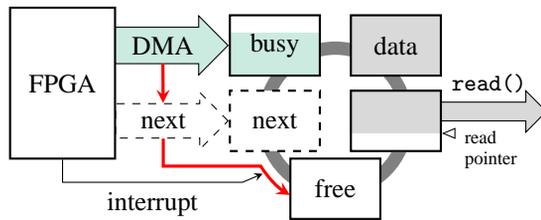


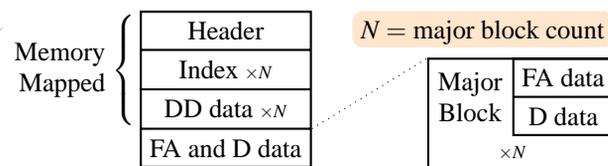
Figure 4: Architecture of the sniffer device driver.

Sniffer Device Driver

The sniffer device driver, see Fig. 4, maintains a circular pool of buffers for the FA sniffer to write to, by default five buffers of 512 K bytes each, enough storage for 40 ms of data per buffer. When the sniffer is running, two buffers are allocated to the sniffer and the remaining buffers contain data for the user space program to read or are free for the next DMA transfer. On each interrupt the pool cycles round by one buffer unless there is no free buffer, in which case the sniffer is allowed to halt.

Data capture is also interrupted during CC network synchronisation or if a communication error occurs. Any interruption is signalled by the driver as “end of file” and recorded by the archiver as a gap in the archive.

The Archive on Disk



Per major block, repeated for each archived FA id:

FA data: x, y $\times 65536$
 D/DD data: $\bar{x}, \bar{y}, [x], [y], [x], [y], \sigma_x, \sigma_y$ $\times 512/4$

Figure 5: Layout of archive store on disk.

The FA data stream from the sniffer is uniform, with constant sized updates received at an unchanging interval. The only variation in this structure is the occasional presence of gaps in the data stream where synchronisation of the communication network has been performed.

This uniform data format allows the archive to be stored as a simple fixed-format file, see Fig. 5. For performance it is better for the archive to be stored directly on the underlying block device, as filesystem block management otherwise adds a significant overhead for very large files.

Early experiments with the archive revealed that the optimum block size for transfers to and from disk is around 512 K bytes and so the layout of the archive on disk is optimised for reading by placing the data for a single FA identifier into blocks of 512 K bytes or 65536 samples each, or around 6½ seconds of data.

At the same time, to help with generating an overview of the entire archive, data is binned into 128 and 16384 sample bins in which the mean, minimum, maximum, and standard deviation values are computed. The decimated $\div 128$ data is stored together with the FA data as a “major block”, the $\div 16384$ data is stored separately in a memory mapped area.

These major blocks are indexed by timestamps recorded in the index which is searched by binary search when an archive read request is processed.

A fixed size (4096 byte) header defines all of the operating parameters of the archiver (block sizes, decimation factors, etc.) together with the list of FA ids archived, and also records the current active block.

Data Processing

Two major blocks are maintained in memory, one being prepared as frames are received from the FA sniffer, the other being written to disk. FA sniffer frames are transposed (in blocks of 256 frames) into the major block layout.

To ensure that reads from the archive do not delay writes there is a limit on the number of reads that can be requested simultaneously, and there is also an interlock to prevent a read being initiated while writing to disk is in progress.

Filtered Decimation

For some applications of the archiver live FA data feed at the full 10 kHz data rate is too much; for example most of the environmental disturbances on the beam have their main effects in the bottom few hundred Hertz. In particular, the full orbit Spectrum Analyser (described below) only covers frequencies up to 320 Hz.

Thus the archiver decimates the entire FA data stream by a factor of 10 using a compensated CIC filter [6, 7], and this data stream is available as an alternative data source for subscribers. The spectrum of the decimated data is flat to ± 0.25 dB with better than 100 dB alias rejection up to 350 Hz.

APPLICATIONS

Archive Retrieval. A command line tool `fa-capture` provides full access from the command line to the dataset stored on the FA archiver, allowing any part of the archive to be retrieved and saved in either raw or Matlab format. Start and end times or start and sample count can be specified in a variety of formats.

Archive Exploration in Matlab. A Matlab tool called the `fa_zoomer` allows the archive to be explored interactively. An interactive window displays the current selection and allows any part of the selection to be zoomed and retrieved in more detail from the archive. The initial window uses “double decimated” data to show an overview of either the last 24 hours or the entire available archive (4½ days), and the decimation is reduced as zooming is refined.

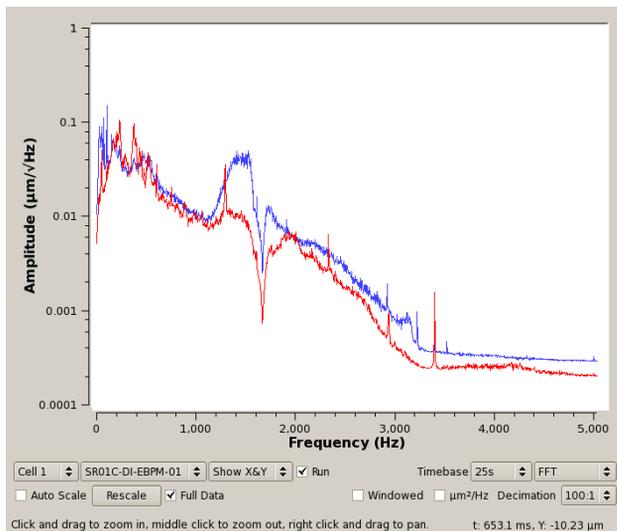


Figure 6: FA viewer application showing beam motion power spectrum of selected BPM over the last 25 seconds.

Live Visualisation. Figure 6 shows the *fa-viewer* tool, a Python Qt application which visualises the motion of the beam at any location. The user can choose to observe any available FA identifier which is then displayed as either a live history display (for up to 60 seconds of history) or a variety of spectral displays: FFT on log y or log xy scales, or an integrated power spectrum. Graphing is done using PyQwt [8] which is both fast and flexible, making it straightforward to develop a large variety of visualisations.

Audio Playback. As the beam disturbance frequencies as processed by the FA archiver are comfortably audible it was natural to try playing back the beam position as sounds. This turns out to work very well, and is occasionally quite instructive. Both topup and injection have very distinctive sounds, but what is more interesting is the variety of very odd sounds that appear to be superimposed on the beam by disturbances in the RF system.

Long Term Spectral Analysis. A dedicated server subscribes to the 1kHz complete live orbit data and computes averaged power spectra in 1 Hz bins averaged every five minutes covering the spectral range from 1 to 320 Hz. Complete waveforms for all frequencies and all 174 BPMs are generated as EPICS PVs, both as power density spectra for each BPM and orbit disturbance for each frequency, and a limited selection of these waveforms is archived in the long term EPICS archive.

This allows a detailed examination of changes in the spectral behaviour of the machine over extended periods; it is easy to look at the spectrum for a week's operation of the machine and identify long term changes in disturbance.

Other Users. The sources for the archiver and driver can be downloaded from the Diamond web site [9], and the FA archiver has been used at Soleil since May 2011.

CONCLUSIONS

The motivation for the FA Archiver has been to gain access to longer periods of synchronously recorded position data from all channels (the whole orbit) at FA data rate. Previously our setup had only been able to capture ten seconds of this data using buffers inside the fast feedback processors. Having access to records of up to several days of this data facilitates the following:

- Identification and detailed investigation of any orbit events that happened in the recent past. To this end, the maximum/minimum/standard deviation calculations on the $\div 16384$ decimated data are invaluable.
- Correlation of orbit events with other measurements by adding X-ray BPMs and RF cavity voltage readings to the FA network [10].
- Modal analysis of orbit motion over longer periods. This is useful for the optimisation of the fast orbit feedback and for the identification of BPMs or correctors with subtle malfunctions.

Together with the long term spectral analysis all these methods allow monitoring and improvement of the reliability and performance of the fast orbit feedback system and ultimately the beam stability.

REFERENCES

- [1] Instrumentation Technologies, <http://www.i-tech.si>.
- [2] M.G. Abbott, G. Rehm, I.S. Uzun, "The Diamond Light Source Control System Interface to the Libera Electron Beam Position Monitors", ICALEPCS 2009.
- [3] I.S. Uzun, R. Bartolini, G. Rehm, J.A. Dobbing, M.T. Heron, J. Rowland, "Initial Design of the Fast Orbit Feedback System for Diamond Light Source", ICALEPCS 2005
- [4] M.G. Abbott, J.A. Dobbing, M.T. Heron, G. Rehm, J. Rowland, I.S. Uzun, S. Duncan, "Diamond Light Source Electron Beam Position Feedback: Design, Realization and Performance", ICALEPCS 2009.
- [5] Virtex-5 LXT ML555 FPGA Development Kit for PCI Express, <http://www.xilinx.com/products/boards-and-kits/HW-V5-ML555-G.htm>
- [6] E.B. Hogenauer, "An economical class of digital filters for decimation and interpolation", IEEE Transactions on Acoustics, Speech and Signal Processing, pp. 155–162, April 1981
- [7] Altera Corporation, "Understanding CIC Compensation Filters", AN-455-1.0, www.altera.com/literature/an/an455.pdf
- [8] <http://pyqwt.sourceforge.net/>
- [9] <http://controls.diamond.ac.uk/downloads/other/fa-archiver>
- [10] G. Rehm, M.G. Abbott, C. Bloomer, I. Uzun, "Synchronous Measurement of Stability of Electron Beam, X-Ray Beam, Ground and Cavity Voltage", DIPAC'11