ZEBRA: A FLEXIBLE SOLUTION FOR CONTROLLING SCANNING EXPERIMENTS

T. Cobb, Y. Chernousko, I. Uzun, Diamond Light Source Ltd, Oxfordshire, UK

Abstract

This paper presents the ZEBRA product developed at Diamond Light Source. ZEBRA is a stand-alone event handling system with interfaces to multi-standard digital I/O signals (TTL, LVDS, PECL, NIM and Open Collector) and RS422 quadrature incremental encoder signals. Input events can be triggered by input signals, encoder position signals or repetitive time signals, and can be combined using logic gates in an FPGA to generate and output other events. The positions of all 4 encoders can be captured at the time of a given event and made available to the controlling system. All control and status is available through a serial protocol, so there is no dependency on a specific higher level control system. We have found it has applications on virtually all Diamond beamlines, from applications as simple as signal level shifting to, for example, using it for all continuous scanning experiments. The internal functionality is reconfigurable on the fly through the user interface and can be saved to static memory. It provides a flexible solution to interface different third party hardware (detectors and motion controllers) and to configure the required functionality as part of the experiment.

INTRODUCTION

Diamond Light Source[1] is a third-generation 3 GeV synchrotron light source based on a 24-cell double-bend achromatic lattice of 561m circumference. The photon output is optimised for high brightness from undulators and high flux from multi-pole wigglers. The accelerators and first phase of seven photon beamline were constructed from 2002 to 2007; a second phase of fifteen photon beamlines from 2006 to 2012; and a third phase of ten photon beamlines was approved in 2011 with construction due to finish in 2017-8.

In order to efficiently run user experiments it is generally necessary to synchronise a range of equipment. This equipment varies from beamline to beamline, but some examples are listed below:

- Motion controllers like Delta Tau Geobrick LV[2] or Newport XPS[3]
- Area Detectors like Dectris Pilatus[4] or PCO Dimax[5]
- ADC cards like Hytec 8401[6]
- Counter/timer cards like Struck SIS3820[7]
- Multi Channel Analysers like the Canberra Model 556B Acquisition Interface Module[8]
- Interfaces to the machine timing system like Micro-Research Finland Event Receiver cards[9]

Most of the examples above have a range of digital inputs and outputs to allow this synchronisation using trigger or gate signals. For example, a motion controller may be able to produce a pulse every N motor counts that can be used to trigger a detector (see Fig. 1).

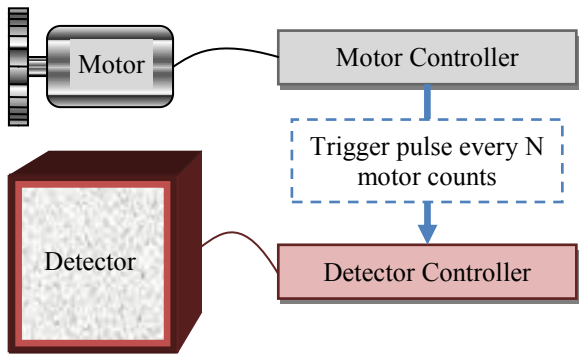

Figure 1: Sample experimental setup.

Unfortunately there are a number of commonly used signal standards, so some level conversion electronics is often necessary to make these connections. In addition to this, some simple logic is often required, like selecting between input signals or stretching pulses to a particular length. In the past, Diamond has created a range of small signal boards that do this job, but the proliferation of such boards makes provision of spares difficult, and increases development time for each new solution, so the decision was made to design a general purpose box to replace these. This box was given the name ZEBRA.

INITIAL REQUIREMENTS

A survey of suggested use cases identified three main categories of requirements: Signal level conversion, simple logic blocks, and position capture.

Signal Level Conversion

It was essential that ZEBRA support all signal standards commonly in use on beamlines, namely:

- TTL (Transistor–transistor logic), 1 kΩ input impedance, line driver output capable of driving a 50 Ω input
- LVDS (Low-voltage differential signalling)
- NIM (Nuclear Instrumentation Module)
- PECL (Positive emitter-coupled logic)
- Open Collector

A comparator input was considered useful, allowing for non-standard voltage signal levels.

ISBN 978-3-95450-139-7

Simple Logic Blocks

A list of logical functions that had already been implemented by other small circuit boards was drawn up and expanded to include other potentially useful blocks:

- Logical AND of up to 4 signals, with the option to invert individual input signals
- Logical OR of up to 4 signals, with the option to invert individual input signals
- Set-reset gate, with options to trigger each input on rising or falling edges
- Pulse divider with programmable divisor
- Pulse generator with delay and pulse width options
- Quadrature generator taking step and direction signals as input

Chaining blocks together was necessary, as was the ability to route the outputs of the blocks to any physical output.

Position Capture

A range of position capture functionality is available in most motion controllers in use on beamlines, with the use cases falling into three main categories:

- Position based capture – where the motion controller outputs pulses when the encoder position reaches a certain value, or regularly spaced series of values
- Time based capture – where the motion controller outputs pulses at regular time intervals, storing the encoder position when this happens
- External trigger capture – where the motion controller stores the encoder position on an external trigger signal

Unfortunately, each controller implements the above functionality in a different way, and the Delta Tau Geobrick commonly used on beamlines only supports the first mode. There was also a requirement to mix modes together, outputting a time based pulse stream within a series of position based gates. The position capture block in ZEBRA was designed to meet all these needs, having an Arm signal, repeating Gates, and a stream of Pulses, each of which could be position based, time based, or externally triggered (see Fig. 2).

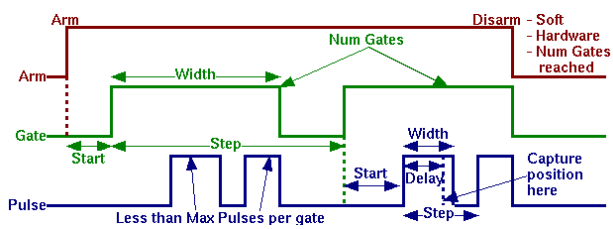

Figure 2: Arm, Gate and Pulse signals of ZEBRA position capture block.

HARDWARE IMPLEMENTATION

ZEBRA consists of a 1 U metal box, with BNC and LEMO connectors on the front for single ended and differential signals, and 15-way D-type connectors on the back for RS422 encoder signals, power and RS232 (see Fig. 3).

Figure 3: ZEBRA physical appearance.

Inside the box, there is a 50 MHz clocked FPGA with some flash and SRAM memory. There is also a significant amount of signal level conversion circuitry for converting input and output signal levels to the CMOS levels used by the FPGA.

FIRMWARE IMPLEMENTATION

Inside the FPGA, there are a number of logic blocks, connected by the system bus. This allows the input of any logic block to be taken from a physical input or the output from any other logic block. Each physical output is also taken from the system bus in the same way (see Fig. 4).

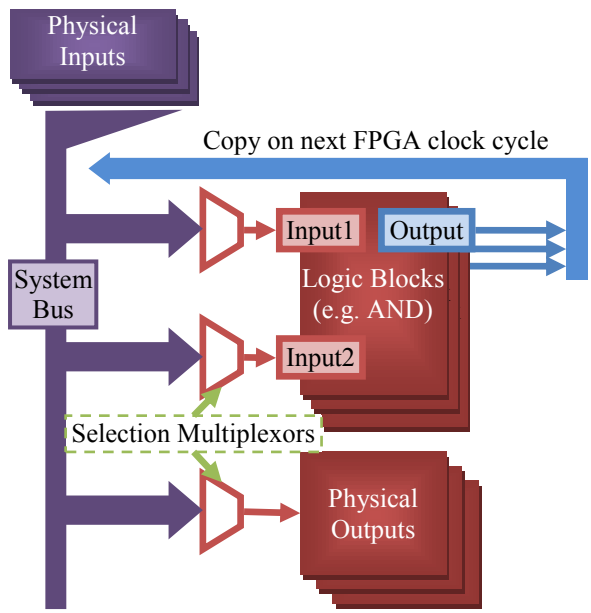

Figure 4: FPGA System Bus.

On each FPGA clock cycle, the system bus values are created from the physical inputs and the last clock cycle's logic block outputs. Each logic block then uses a multiplexor to select a particular input from the bus, and prepares its outputs based on these inputs.

The FPGA also contains a PicoBlaze[10] soft processor which handles the RS232 communications. Using a command and response protocol, it exposes a number of registers over this interface, like the values of the selection multiplexors for each logic block input, or the divisor value in the pulse divider block. It also generates

ISBN 978-3-95450-139-7

unsolicited messages with time and position data whenever position capture is active.

SOFTWARE IMPLEMENTATION

The software consists of an EPICS[11] driver based on the asynPortDriver[12] model, with the driver handling RS232 communications to the FPGA, and exposing a set of parameters to the EPICS database and so to the user. These parameters include all FPGA register, as well as position and time arrays and controls for backing up and restoring from file the register settings. The EPICS database does the conversion to engineering units to make the parameters more presentable to the end user. A set of EDM[13] screens provide the GUI (see Fig. 5).

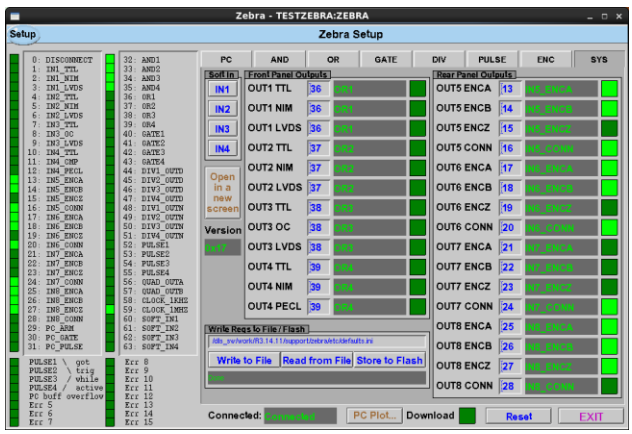

Figure 5: EDM screen showing wiring of output signals.

The left hand side of the screen shows the current status of the system bus, while the right hand side is a tabbed window allowing setting of registers for each group of logic blocks. The currently selected tab shows which elements of the system bus are connected to each output. The EPICS driver is available from Diamond Controls website[14].

USAGE ON BEAMLINES

ZEBRA has found a number of uses on beamlines so far, two of which are detailed below.

Logic blocks

Users on an infrared beamline wish to synchronise an Atomic Force Microscope (AFM) with a fast beam chopper. A Delta Tau Geobrick is used to drive the chopper, requiring an RS422 differential quadrature input to use as a speed reference (OUT5_ENCA and B in Fig. 6). The AFM outputs a 0 - 5 V sine wave that corresponds to its scanning frequency (IN_TTL1 in Fig. 6). ZEBRA uses two pulse generators, two logic gates, and a set-reset gate to debounce the signal, before quadrature encoding it and sending it to the A and B channels of the encoder output.

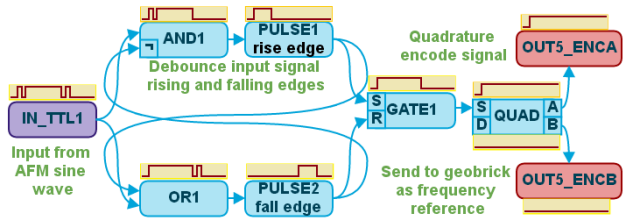

Figure 6: Diagram of chopper logic block connections.

The input signal has some noise on the rising and falling edge, so PULSE1 and PULSE2 extend the rising and falling edges, AND1 and OR1 mask out the noise during these extended period, and GATE1 turns it back into the correct length pulse before QUAD quadrature encodes it for the Geobrick.

Position capture

Users on an imaging beamline wish to do tomography with a fast rotation stage and a PCO Dimax camera. The camera takes a trigger signal and exposes for a fixed amount of time on each trigger signal. To maximise the time the camera is exposing, ZEBRA sends out time based pulses within a position based gate of 0 - 180 °. Figure 7 shows a cropped plot of the position of the stage against the time the trigger pulse was sent to the camera, along with a sinogram generated from the captured data. This data took 9 seconds to capture.

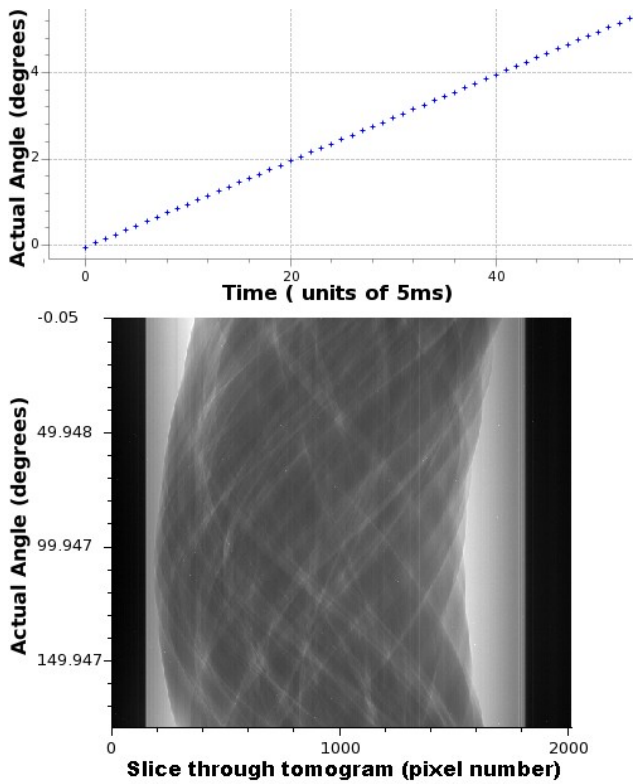

Figure 7: Example of tomography pulses and sinogram.

ISBN 978-3-95450-139-7

The maximum number of pulses per gate is limited to ensure that a predictable number of images are captured, and the capture of the position could be delayed to ensure that it corresponds to the midpoint of the camera exposure time if needed.

LIMITATIONS

ZEBRA is designed around a small FPGA, which is now 99% full. This leaves little room for future improvements, but hopefully the range of logic blocks will satisfy most beamline requirements of it. There are some limitations that may be addressed in a future project:

- Serial line at 115200 baud is limited to streaming about 500 position capture points per second, so acquisitions faster than this lead to a delay at the end while buffered samples are downloaded. A larger FPGA with an ethernet connection would solve this.
- The comparator input threshold can only be changed in hardware. Including some ADC inputs would allow this threshold to be changed remotely.
- ADC inputs would also allow analogue values to be stored as part of position capture.
- Position capture is limited to a repeating pulse with a step fixed in position or time. A larger FPGA would allow arbitrary arrays of position or time points to be used in position capture.

CONCLUSION

Four ZEBRA units have now been rolled out to beamlines, with an additional twelve due to be commissioned this year, and thirteen more in 2014. They have proved especially useful for their position capture interface, facilitating an increase in the use of continuous scanning for faster acquisition. Quantum Detectors Ltd [15] are in the process of manufacturing a second batch for Diamond, and these will also be made available commercially.

ACKNOWLEGEMENT

It is with much sadness that we report that Yuri Chernousko, a developer of ZEBRA and a co-author of this paper passed away on the 31st July 2013.

REFERENCES

[1] R. P. Walker, “Commissioning and Status of The Diamond Storage Ring”, APAC 2007, Indore, India.
[2] THE GEO BRICK LV IMS II; http://www.deltatau.co.uk/geo-brick-lv-ims2.html
[3] XPS Series Motion Controllers; http://www.newport.com/XPS-Series-Motion-Controllers-Universal-High-Perf/300904/1033/info.aspx
[4] PILATUS Hybrid Pixel Detectors; https://www.dectris.com/pilatus_overview.html
[5] High Speed Cameras - pco.dimax S; http://www.pco.de/categories/high-speed-cameras/pcodimax
[6] IP-ADC-8401 8-CHANNEL ADC INDUSTRY PACK; http://www.hytec-electronics.co.uk/DataSheetHtml/8401.html
[7] SIS3820 multi purpose scaler; http://www.struck.de/sis3820.htm
[8] Model 556B AIM - Acquisition Interface Module; http://www.canberra.com/products/radiochemistry_lab/nim-multichannel-analyzer.asp
[9] VME Event Receiver w/o RF (VME-EVR-230); http://www.mrf.fi/index.php/vme-products/76-vme-event-receiver-wo-rf-vme-evr-230
[10] PicoBlaze 8-bit Embedded Microcontroller User Guide; http://www.xilinx.com/support/documentation/ip_documentation/ug129.pdf
[11] Experimental Physics and Industrial Control System; http://www.aps.anl.gov/epics
[12] C++ Base Class for Asyn Port Drivers; http://www.aps.anl.gov/epics/modules/soft/asyn/R4-21/asynPortDriver.html
[13] EDM: an EPICS display manager; http://ics-web.sns.ornl.gov/edm
[14] Diamond Controls - ZEBRA EPICS driver; http://controls.diamond.ac.uk/downloads/support/zebra
[15] Quantum Detectors - Detection systems for research and industry; http://www.quantumdetectors.com

ISBN 978-3-95450-139-7