

# A 48 Channel Pulse Shape Digitizer with DSP

J.-P. Martin, *University of Montreal, Canada; Senior Member, IEEE*  
and P.-A. Amaudruz, *TRIUMF, Canada; Member, IEEE*

**Abstract-** A 48 channel, 40-65 MS/sec., 10 bit pulse shape digitizer card has been designed in the VME-6U form factor. The design uses 6 octal Flash Analog to Digital Converter (FADC) chips (ADS5121 or ADS5122) from Texas Instruments. The FADCs are read out by 6 Altera Cyclone FPGAs. A 7<sup>th</sup> FPGA is used to collect and merge the event fragments. The present firmware includes trigger latency buffers, waveform segment buffers, real-time digital filtering, time stamp generation, amplitude evaluation, event formatting and buffering, plus a simple VME A24D32 interface. The design also includes a source synchronous bi-directional serial LVDS link for the interconnection of the modules to a large system. The first version of the module is designed for an R&D test readout system connected to a few hundred drift chamber cathode pads or TPC cathode/anode pads. The final version is intended for the readout of the KOPIO preradiator cathode drift chamber pads (~75,000 channels). This version will only have the LVDS interconnect. The present design using off-the-shelf commercial components is an interesting alternative to an ASIC approach for intermediate size readout system. It occupies about 3 times the area of an equivalent ASIC system, and has comparable power dissipation. In its present form, the development version of the module can be useful as a general purpose waveform digitizer and DSP with its high density (48 channels per single width VME module) at a relatively low cost per channel.

## I. INTRODUCTION

The digitizer card, named “VF48”, has been designed as a validation model for the readout system of the KOPIO preradiator [1], as well as the basic building block for data acquisition systems for the development and testing of the drift chambers prototypes. In this particular application, the physical pitch of the readout channels is compatible with a non-ASIC design, with the readout cards mounted directly on the detector. ASIC and non-ASIC solutions are both possible candidates for the digital part of the system. At the level of 75,000 data channels the preliminary design study showed that the non-ASIC solution is still cost effective and meets the power dissipation constraint. As a bonus, the system can be developed more rapidly, and the flexibility of the commercial FPGAs allows for field upgrading.

The Low Voltage Differential Signaling (LVDS) serial link will also be used to validate the full readout architecture. This includes clock and trigger distribution, parameter loading and event collection. The serial architecture drastically reduces the cabling requirement, which is not a negligible budget item. In the final system, some of the links will be over copper and other over optical fiber.

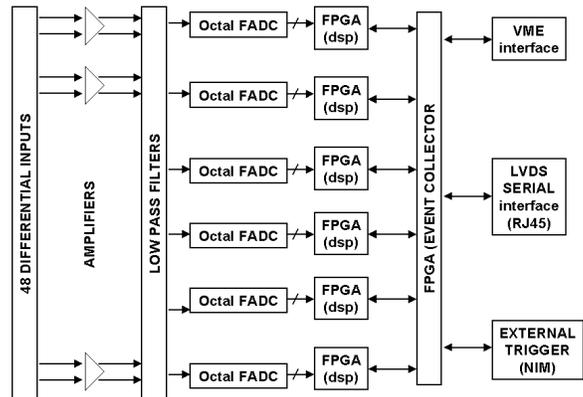


Fig. 1. Simplified block diagram of the VF48 card

The point-to-point communication protocol at every level will remain the same, regardless of the physical medium. To take advantage of this serial link, a VME 6U collector card has also been developed to readout up to 12 VF48 cards over copper LVDS links. This collector card can provide the clock distribution and the level zero (L0) trigger decision. It also has an input for an external trigger. The trigger logic can also use the hit information coming from the front end FPGAs. The physical interconnection between the VF48 cards and the collector card is done with standard Category 6 (CAT6) patch cables equipped with RJ45 connectors.

The VME port at the card level is a valuable tool in the development process. First, it is very convenient for the testing of the individual cards. Second, it simplifies the setup of a scaled down version of the final readout system early in the R&D phase of the KOPIO drift chamber development. Third, it makes the VF48 much more general, so it can be used for other purposes.

## II. PHYSICAL DESCRIPTION

A simplified block diagram of the VF48 card can be seen in Fig. 1. The 48 differential inputs are located on a front panel 100 pin header connector compatible with standard small pitch flat cables. The input pins feed 48 differential amplifiers that adapt the dynamic range of the preamplifiers to the dynamic range of the ADCs. A DC offset of either polarity can also be injected in order to obtain a larger dynamic range for unipolar pulses. An anti-aliasing filter is

inserted between the amplifier outputs and the ADC linear inputs in order to optimize the signal to noise ratio.

Six octal ADS5121 ADC chips are required for the analog to digital conversion. The 80 bits from the 8 digitized data samples of each ADC chip are collected by an Altera Cyclone EP1C6 or EP1C12 front end FPGA according to the size of the buffers required, and the complexity of the digital system processing. A 7<sup>th</sup> FPGA collects the information from the 6 front-end devices. It is interconnected to the front end FPGAs with serial LVDS links, similar to the external links, except that the wires are differential printed circuit board traces rather than patch cables. This collector chip also takes care of the VME interface logic as well as the external LVDS link. The VME interface supports the basic A24D32 transfers, and is compatible with most commercial VME interfaces or embedded processors.

A front panel coaxial connector is available for an external trigger. Each card is also equipped with a crystal clock. For a system that uses more than one card, the on board crystal clock can be disabled while running from an external master clock. The master clock can be taken from either the LVDS external link or from the VME SERCLK bus line. A synchronization signal can also be derived from the LVDS link, or the VME SERDAT bus line. These signals are synchronous with the master clock and may be used to synchronize the time stamp logic in each card.

The external LVDS link is accessible from an RJ45 connector on the front panel and can be connected to a custom distribution card. This option may be used with multi-crate systems. The distribution cards may be daisy chained to form a distribution and data collection tree. The physical medium for a link is a standard Category 6 (CAT6) patch cable. As the links are source-synchronous, with the reference clock and the data bits transmitted over separate pairs within the patch cable, the pair to pair timing skew is a concern. The deserializer logic allows for a maximum skew of 2 nanoseconds in either direction. The CAT6 specifications tolerate a difference of propagation time of 10% for two cables of the same length. We expect the pair to pair skew within one cable to be less than that. Even in the worst case of the CAT6 specifications, a 3 meter cable will have a skew of less than 1.5 nanoseconds. When CAT7 cable becomes available, this will be reduced to 600 picoseconds.

### III. FRONT END PROCESSING LOGIC

The six front end processing FPGAs are programmed with identical firmware. Each FPGA processes the 8 channels of digitized waveform samples coming out of one ADS5121. For the KOPIO application, the master clock and the ADC sampling rate are 25 MHz. Another option is a 20 MHz master clock, and a 40 MHz ADC sampling rate. With

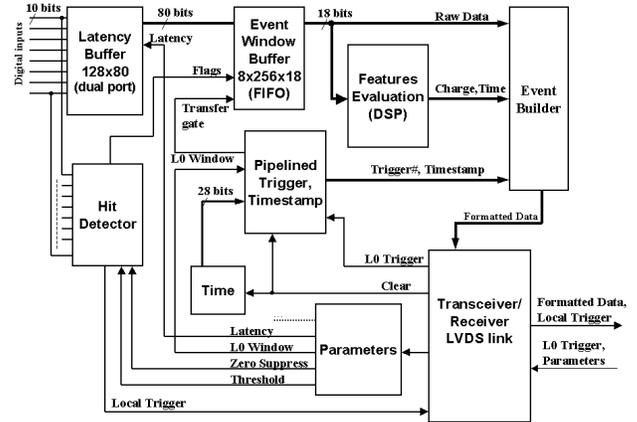


Fig. 2. Simplified block diagram of the front end processing logic

the ADS5122, the sampling rate can be extended to a maximum of 65 MHz. The front end processing logic includes various buffers, digital filters, hit detectors, trigger control, event fragment building and formatting, time stamp logic, “feature extraction” logic, and serial link control. In the baseline design, the “features” extracted in the analysis of the pulse shapes are the charge and time associated with the signal. Fig. 2 shows a simplified block diagram of the front end processing logic. The details of the various elements for one of the 6 front-end processing logic groups are described below.

#### A. Time stamp logic

A 28 bit counter clocked at the master clock frequency generates the local coarse time stamp reference. The depth of the counter has been chosen in the KOPIO application to fit into a single 32 bit event word, yet be large enough not to overflow during a beam spill. In this application, all the time stamp counters are reset synchronously at the leading edge of the beam spill.

#### B. Hit detector logic

The ADC data bits for each channel are processed with a digital filter that performs a function equivalent to an analog clipped delay line, giving a baseline suppressed pulse. A digital comparator generates a hit pulse when the resulting baseline suppressed signal crosses a specified threshold. This information is relayed to the master trigger logic responsible for the level zero trigger decision (L0). The hit information is also used in the “zero suppress” mode of operation. In this mode, only the pulse shapes on the channels where a hit was detected are taken into account when an L0 trigger is generated. If the “zero suppress” mode is not activated, all the channels are processed at the occurrence of an L0 trigger, regardless of their content.

### C. Local trigger logic

When an L0 trigger is received from the master trigger decision logic, a 20 bit trigger counter is incremented by one, and the value of the trigger counter as well as the time stamp is pushed into a FIFO. At the same time, a “transfer” gate is generated to accept a specified number of ADC data samples or “window” for further processing.

### D. Trigger Latency buffer

The trigger latency buffer is a dual port memory configured as a circular list. It has 128 memory locations and is 80 bit wide (8 x 10 bits). The input port is connected to the ADC data stream. The memory is continuously overwritten at the ADC sampling rate. The output port reads back the data at the present write address, minus a specified latency parameter. The trigger latency buffer acts as a digital delay line for the ADC signals. The latency parameter (or delay) can be adjusted so that the proper segment of the pulse shapes is contained within the “transfer gate” when an L0 trigger is received.

### E. Window buffer

The window buffer contains eight 256 X 18 bits FIFOs (or 1024 x 18 for the deep buffer option). The data inputs are connected to the delayed data signals from the trigger latency buffer. The FIFO “write” control lines are connected to the “transfer gate” from the local trigger logic. At every L0 trigger, a pulse shape “window” is pushed into the FIFOs. The window buffer is clocked at the same rate as the ADC sampling rate, storing the eight data pulse shape segments in parallel. An “almost full” flag is generated when the space left in the FIFO becomes less than what is required to store a complete “window”. The “almost full” information is relayed to the master trigger logic to prevent more L0 triggers from being generated until enough room becomes available again. In this manner, when the throughput limitation causes some dead-time, the events that are read out remain consistent.

The outputs ports of the eight FIFOs are connected to an 8 to 1 bi-directional multiplexer. The multiplexer selects one of the eight FIFOs, and connects its data outputs as well as all the FIFO read control lines and flags to a common port. The FIFO selection is controlled by the event builder logic. While the 8 FIFOs are written into simultaneously, they are read out sequentially, one L0 window at a time. This throughput limitation matches the communication link to the event collector, and simplifies the formatting of the event.

### F. Digital signal processing (Feature extraction)

When the raw data and the various flags are extracted from the window buffer for event assembly, a copy of this information also feeds the input of the feature extraction

logic. This logic block can then process each pulse shape “window” one channel at the time. In the present implementation, the data flow is divided into two streams, one for the charge evaluation, and one for the calculation of a time vernier with a time granularity of 1/16 of the master clock. In each of the two paths, the pulse shape data is processed with a digital filter. These are conventional time invariant filters presently using 8 samples from a digital pipeline [2][3][4]. Each point of the resulting filtered pulse shape consists of the weighted sum of these eight samples. The values of the individual weights are adjusted according to the resulting filtered pulse shape that is required. In our application, the sum of the weight factors is set to zero in order to provide baseline cancellation without the use of a pedestal parameter. Zero is a valid value for any of the weight factors. In some cases, the same filter function is satisfactory for both the charge path and the timing path, so only one digital filter needs to be used. This operation is done in real time, yielding one filtered datum at every clock cycle.

The charge evaluation simply finds the maximum value of the filtered pulse shape, once the signal has been detected to be above a specified threshold.

The time evaluator uses an algorithm similar to an analog constant fraction discriminator. It measures the crossover point of the differentiated signal. This is achieved with a linear interpolation of the two points before and after the zero crossover. This interpolation is done in units of 1/16 of a clock interval. A counter records the number of samples that occurred until the crossover was detected. This is done in units of one clock interval. Both values are then combined to produce a time evaluation relative to the beginning of the data “window”. The absolute time corresponding to the beginning of the window is obtained from the event time stamp.

Both the charge and time information are pushed into a FIFO in order to achieve multi-hit capability within the same data window.

### G. Event builder

The event builder has multiple input ports: the raw data, flags from the “window buffer”, the outputs of the “features” FIFO, and the outputs of the local trigger FIFOs - containing time-stamp and trigger number information. These inputs are scanned under the control of the event builder state machine during the process of formatting the event fragment.

In the idle state, the event builder waits for trigger information to be available at the output of the trigger FIFO. When this situation is detected, the 24 bit trigger number and header flags are pushed into the first event long word. The time stamp is pushed in the second word, and an acknowledge is sent to the trigger FIFO to update its pointers.

The data readout loop is then initiated, starting with channel zero. The channel number is pushed into the event

list, and the 10 bit raw data words are packed two by two into successive event long words. The start and end of window flags bits are also added in the first and last long word of the raw data for the current channel (In the “suppress raw data” mode of operation, the raw data is read out in the same manner in order to clear the window buffer, but the data is not written into the event list). During this process, the feature extraction logic processes the raw data stream, and the results are available at the end of the window data corresponding to a L0 trigger. When the event builder detects the last raw data word of the window, it moves to the feature readout state, and pushes the information from the features FIFO into the event list until the FIFO is empty.

The channel number is then incremented by one and the data readout loop is executed again until the last channel has been processed. A trailer word is then inserted, and the state machine comes back to its idle state, ready to process the next trigger.

Handshake logic suspends this process when the event fragment collector stage is not ready to accept more event list elements.

#### *H. Serial link logic*

Each of the six front-end processing FPGAs communicates with the event fragment collector through a bi-directional full duplex serial link. The link operates at 200 Mbits/sec. In the transmitting direction, the link serializes the information 8 bits at the time, synchronous with the master clock. This eight bit field is partitioned into two groups: four information data bits and four bits for trigger, synchronization, parity and handshake flags. The effective data transfer rate over each of the six serial links is 100 Mbits/sec. The data field may either be event builder data, parameter data, or possibly local trigger pattern data. Since the trigger bit can be inserted at any master clock cycle in both directions, the trigger decision latency can have a fixed value. The serial link state machine controls the steering of the various information elements.

The master clock is distributed over a separate differential link.

#### *I. Parameters*

The serial link protocol allows for transmission of parameter data in both directions. The addressing scheme allows the parameter to be written at any level of the data tree. Within the card, this may be either in any one or all of the front end FPGAs, or in the collector FPGA. Each parameter is addressed by a destination identifier, plus a parameter identifier. The maximum size of a parameter is 64 bits. At power up, the parameters are loaded with default values hard-coded in the firmware.

The event collector merges the event fragments coming from the six front-end serial links. The link logic is identical with that of the front-end. For each of the six serial links, the information is de-serialized into 2 x 4 bit streams. The event data bits are then reformatted into long words, exactly as they came out of the front end event builder, and pushed into one of the six link event fragment FIFOs. This collection process has a peak data throughput of 600 Mbits/sec.

#### *A. Event fragments collection*

The event fragment collector logic operates in a manner similar to that of the front-end event builder. It scans the six FIFOs link and extracts the data information one channel at a time per L0 event, concatenating the information in the output buffer. In a large system of VF48 cards read out with external LVDS links, the same process can be repeated at every level of the event merging tree. In the case of a single card, the card’s output buffer is the final destination. This buffer is directly accessible in FIFO mode through a VME read command.

#### *B. Parameter dispatching*

The parameter loading and read-back mechanism is accessible directly from VME through two separate sets of registers for read and write. Each set is composed of a 32 bit address register and a 64 bit parameter data register. These registers are present in each VF48 card to support a single card system. The address register defines the path and the parameter identifier as well broadcasting over multiple branches is allowed. The read-back of a parameter is done in two steps: first, a dummy write function is initiated with the target address being written into the parameter address register together with a read flag bit. At the target destination, the parameter logic sends back its parameter value, together with its address and the acknowledge flag. This information is stored in the second set of data registers. Second, a VME read command is initiated to read-back the parameter information from this second set of data register. In the read-back mode, broadcast to multiple targets is not supported.

#### *C. Master trigger logic*

In the internal trigger mode, the global L0 trigger decision is usually performed at a single point, at the top level of the LVDS link hierarchy. However, the basic global trigger logic is present in each card, and may be activated as required. For the present application, the trigger logic either selects the logical OR of the trigger requests coming from the front-end, or an external trigger port. It also starts a dead-time timer to prevent multiple triggering during the “transfer gate” period of the front end. The front end can nevertheless record multiple hits within a trigger. The raw

## IV. EVENT COLLECTOR

data stream is inherently multi-hit, and the feature extraction logic has multi-hit capability.

## V. RESULTS

The first few prototype cards are at their very early stage of testing, only functionality results are available at this moment. The serial links have proven to be very robust. For the external links over CAT6 patch cables, the tests have only been made so far with relatively short lengths of cables (3 meters) with zero error rate. All the baseline functionality of the card has been tested in a single card configuration. A preliminary test of the signal noise without inputs has also been performed. The RMS noise measured on one channel at random is less than one third of the least significant bit, or  $\sim 70$ dB. The dynamic specifications for the ADC itself mention a signal to noise ratio (SINAD) of 59 dB for a full-scale sine wave at 20 MHz.

## VI. CONCLUSION

The VF48 cards are interesting devices for the instrumentation of intermediate size detectors, when the development of an ASIC is not an absolute necessity. They provide a reasonably high channel density, and do not require any custom component. With the VME interface, they can equip detector test systems of a few hundred channels.

With the serial link architecture, they can readout efficiently up to 100,000 channels. With the bandwidth filter removed, and with an adequate timing sequencer, the modules can also be used for the readout of multiplexed preamplifiers.

## ACKNOWLEDGMENT

The authors would like to thank Altera for providing FPGA Cyclone samples and Miles Constable for the English revision of the work.

## REFERENCES

- [1] Yu. G. Kudenko "KOPIO Experiment at BNL" Kaon Decay Workshop for Young Physicists, Tsukuba, Japan, 14-16 February 2110. Published in "Tsukuba 2001, Kaon Decay Physics" pp. 169-189
- [2] M. Rost and W. Weihs "20 MHz hardware digital filter for signals of the ZEUS forward tracking detector" Nuclear Instruments and Methods in Physics Research Section A, Vol. 345, Issue 2, June 1994, pp. 324-328
- [3] G. Bertuccio, et al. "An Optimum Digital Signal Processing for Radiation Spectroscopy" Nucl. Instr. And Meth, Vol A353, 1994.
- [4] Steve W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing", Second Edition, California Technical Publishing, San Diego, California, <http://www.dspguide.com/pdfbook.htm>