An FPGA-based Data Acquisition System for Directional Dark Matter Detection

Chen Yang, Jiayi Sheng, Aravind Sridhar, Martin C. Herbordt
Department of Electrical & Computer Engineering
Boston University, Boston, MA 02215
Email: {cyang90,jysheng,aravinds,herbordt}@bu.edu

Catherine Nicoloff, James B. R. Battat
Department of Physics
Wellesley College, Wellesley, MA 02481
Email: {cnicolof,jbattat}@wellesley.edu

***Abstract*—Directional dark matter detection seeks to reconstruct the angular distribution of dark matter particles traveling through the laboratory. A directional detector with high spatial resolution has the potential to increase the sensitivity per unit volume by over two orders of magnitude, but requires the development of a high-channel-count, high-speed readout system. This paper describes an FPGA-based digital back-end system to handle a 16 Gbps data stream from 10^3 independent detector channels sampled at 1 MHz. Results of an implementation of this system are presented, along with plans for future development.**

***Keywords*-Directional Dark Matter Detection; FPGA; Time Projection Chamber; DAQ; Real-time.**

I. Introduction

Astrophysical observations reveal that dark matter accounts for $\sim 80\%$ of the matter in the Universe [1]. There is a world-wide program underway to detect the evidence of an interaction between a dark matter particle and a target nucleus in the laboratory. For Weakly Interacting Massive Particles (WIMPs – a favored dark matter candidate [2]), the interaction is an elastic collision that creates a recoiling nucleus. Directional dark matter detection is a sub-field of WIMP detection that aims not only to detect the occurence of such an event, but also to reconstruct the geometry of the nuclear recoil track [3]. The angular distribution of recoil tracks provides a smoking-gun signature of WIMP dark matter [4].

The implementation of a directional dark matter detector is challenging [5]. A promising, mature technology is the low-pressure-gas Time Projection Chamber (TPC) [6], in which a nuclear recoil generates a track of ionization that is drifted to a readout plane using a uniform electric field. To reconstruct 3D tracks in a TPC, one needs fine spatial granularity ($\sim 200\,\mu$m) over large areas ($1\,\mathrm{m}^2$). A challenge, then, is how to read out the charge signal from such a detector that has $\sim 10^4$ independent channels. In this work, we introduce an ASIC- and FPGA-based charge readout system for a prototype directional dark matter detector. Beyond dark matter detection, this work is of broad interest in experimental particle physics because it provides high spatial resolution in a large-volume detector.

One of us (JBRB) is a member of the Directional Recoil Identification From Tracks (DRIFT) experiment, in operation since 2000 [7]. DRIFT uses a TPC filled with negative-ion gas (a NITPC), in which the track diffusion is reduced to the thermal limit [8], facilitating the scale-up to larger-volume detectors. Also, the slow drift speed of negative ions (~ 10 cm/ms) relaxes the sampling frequency requirement at the readout plane (in a TPC, the 3D track is reconstructed by building up sequential samplings of the 2D readout plane, x-y). For example, 1 MHz sampling corresponds to $100\,\mu$m granularity along the drift direction (z). DRIFT currently has the leading limits on dark matter interactions from a directional detector [7]. It was recently shown that a NITPC filled with SF_6 read out with high spatial resolution would increase the sensitivity of DRIFT by two orders of magnitude per unit volume [9]. Our work aims to develop the desired high-resolution readout, dubbed **Mi**cromegas with **N**egative **I**ons for **D**irectional **D**ark matter **D**etection, or **MiNI-3D**.

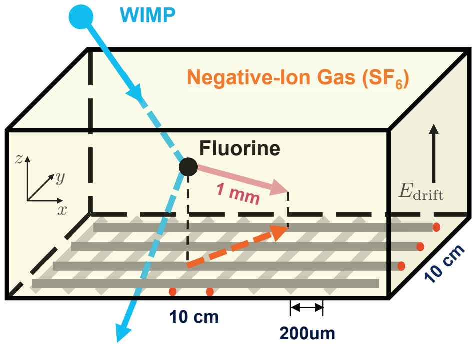

Figure 1. Schematic of a WIMP-induced fluorine recoil in a TPC with strip readout. The WIMP (blue line) collides with a fluorine nucleus (black dot) to create a fluorine recoil (pink arrow). The resulting track of charge drifts to the x-y readout plane (orthogonal, electrically isolated strips), forming a 2D projection of the track (orange dashed arrow). The track position and geometry are found by spatial coincidence between the strips. In this example, strips marked with red ovals at their ends will receive a significant charge signal.

At 10^3–10^4 channels, the data processing requirements of MiNI-3D fall in an intermediate regime: too substantial for commercially available equipment, yet not large enough to justify dedicated ASIC development. Fortunately, the goal of scaling liquid argon (LAr) neutrino detectors to channel counts $> 10^5$ has stimulated the production of ASICs for

978-1-5386-3472-1/17/$31.00 ©2017 IEEE

charge integration and shaping, as well as digitization and multiplexing. By fortuitous coincidence, the requirements for charge detection in liquid argon match the needs of a gas-based NITPC. As a result, the transimpedance gains and shaping timescales, as well as the digitization rate of custom ASICs developed by Brookhaven National Laboratory (BNL) for LAr detectors [10], are fully compatible with our needs. Indeed, MiNI-3D does not require any ASIC modification, or even any circuit board design/layout. All boards are either commercial-off-the-shelf (COTS) or fully specified by BNL. This greatly reduces the data acquisition system (DAQ) development time and cost, and allows us to benefit from ongoing upgrades (by BNL) to the ASIC design. The BNL analog and digitizer ASICs are packaged into a so-called Front-End (FE) system, which we couple to our NITPC for a dark matter search.

As a prototype for MiNI-3D, we will use a TPC with a $10 \times 10\,\mathrm{cm}^2$ micromegas gas amplification device with 10^3 x-y strips [11], read out with the BNL FE. The details of the MiNI-3D detector are presented in Section III, but the key idea is that the 16 Gbps data generation rate from the FE requires us to develop a digital Back-End (BE) system to manage and trigger on the fast data stream, recording only interesting events to disk. This paper describes the FPGA-based implementation of the BE, as well as the FE-BE interface.

ASICs and FPGAs have been used in dark matter detection before. For example, the NEWAGE directional dark matter experiment uses an amplifier-shaper-discriminator (ASD) ASIC for data collection and trigger application which can be used in a micro pixel chamber [12]. But that chip records time-over-threshold (TOT) across four channels instead of recording the entire waveform. The MIMAC directional dark matter experiment created a custom ASIC to sample and trigger on 1024 channels at a rate of 50 MHz, again, recording only TOT, not the full waveform [13]. More recently, the LUX non-directional dark matter experiment has developed an FPGA-based trigger system to monitor the signal on 122 photomultiplier tubes (PMT). The FPGA implements digital filtering and an event trigger based on the analog sum of eight PMT signals, and waveforms are recorded by a COTS waveform digitizer [14]. That system has 10× fewer detector channels, and does not implement channel-by-channel triggering.

II. Detection scheme and physics requirements

A. Time Projection Chamber with Micromegas readout

The $10 \times 10 \times 2.5\,\mathrm{cm}^3$ detection chamber is filled SF_6, a negative ion gas. Fig. 1 shows the TPC-based detection scheme. A WIMP may enter the chamber and interact with the fluorine nucleus of a SF_6 molecule, generating a fluorine nucleus recoil. As the nucleus comes to rest it generates a track of ionization electrons. The highly electronegative SF_6 gas will readily attach the free electrons, forming SF_6^-.

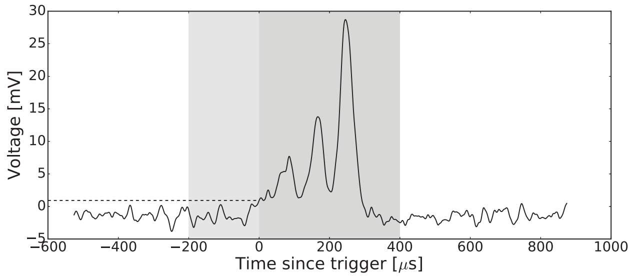

Figure 2. Sample waveform from a single detector strip. Each peak is caused by a different charge carrier species with different drift velocity. The time between peaks can be used to determine the z coordinate of an event in the detector. The area under all peaks gives the total ionization falling on that strip. Dashed line shows a voltage threshold trigger level, and gray shading shows pre- and post-trigger ranges (here, we show only $200\,\mu$s pre-trigger and $400\,\mu$s post-trigger).

This track of SF_6^- negative ions then drifts under an applied electric field ($\approx 500\,\mathrm{V/cm}$) toward the micromegas readout plane. In the micromegas amplification gap ($256\,\mu$m), a strong electric field ($\sim 20\,\mathrm{kV/cm}$) strips the excess electron from the SF_6^- ion and then generates an electron avalanche (exponential amplification). This charge signal is sensed by an array of orthogonal strips with a $200\,\mu$m pitch, with 500 strips in the x and 500 strips in the y directions.

B. Properties of the charge signal

From the charge signal, we can reconstruct the recoil energy and track the geometry (length and direction) and 3D location of the event in the detector. The recoil energy is determined from the total deposited charge in the detector (assuming that the recoil quenching factor is known [15]). Signal coincidence between the x and y strips gives spatial information in 2D (both the absolute x and y coordinates of the interaction and the length of the track Δx and Δy). The third component Δz of the recoil track is obtained by compiling many such measurements in rapid succession (1 MHz sampling).

Locating the interaction vertex in the detector (a.k.a. full-volume fiducialization) is critical in dark matter detection because many background populations are caused by radioactive impurities in the detector materials. Those backgrounds interact near the boundary of the active volume, so, by measuring the positions of events, these backgrounds can be tagged and cut during the analysis. The x-y position of an event is determined as explained above. Negative ion gases like SF_6 enable the measurement of the z coordinate (along the drift direction) through the multi-peak structure in the charge signal (see Fig. 2). Each peak corresponds to a different species of negative ion (*e.g.* SF_6^- and SF_5^-), with different drift speeds [9], [16]. The difference in arrival time between the species at the readout plane is therefore directly proportional to the total drift distance z. Accordingly, the DAQ must record all charge peaks.

In sum, the design requirements for the charge readout are then 1 MHz simultaneous sampling of 10^3 channels, with 12-bit resolution to provide enough dynamic range to capture the tallest and shortest peaks in the charge signal, and with a long enough record (20 ms) to capture all negative ion peaks.

C. *Event trigger*

The event trigger is a critical piece of the DAQ. Ideally, the event trigger would sense the total ionization of a recoil, not just a threshold voltage on a single channel. This requires that the trigger examine the integral of the waveform and incorporate the signal from several adjacent detector channels (*i.e.*, a multi-channel trigger). Furthermore, because a trigger may not be satisfied until a sizable fraction of the event has transpired, the DAQ must preserve pre-trigger information (for *all* channels). Our design preserves 5 ms of pre-trigger and 15 ms of post-trigger data. Because a 1 mm-long recoil would span $\sim$ 10channels, the DAQ need only transfer data from a small subset of the channels to disk (*e.g.*, the channels involved in the trigger, plus a few neighboring channels).

During a WIMP search, the expected event rate is set by the backgrounds, with 1 Hz typical for the 1 m^3 DRIFT-IId detector [7]. During calibration runs, however, the event rate can be much higher (1 kHz) and the triggering and readout scheme should handle this rate. Furthermore, the event trigger should be flexible and adjustable in the field to respond to evolving needs and detector performance changes. The use of an FPGA in the BE provides the technological resources and flexibility demanded by the above requirements.

III. Data acquisition system

The detector system consists of three major parts (see Fig. 3): the TPC, including the micromegas with x-y strip readout as described in Section II, the Front-End (FE) electronics for signal conditioning and digitization, and a single FPGA-based Back-End (BE) that handles the real-time data stream, applies trigger conditions, and saves interesting events to a computer for off-line analysis.

A. *Front-End: signal conditioning and digitization*

A set of eight FE boards made by BNL measures the charge signal on each of the 10^3 micromegas strips. Each FE board provides analog signal conditioning and digitization, and outputs a single serialized stream of the digital data. The detector charge signal on each channel is first conditioned by a custom analog ASIC (LARASIC4, CMOS 0.18 μm technology with charge preamplifier, and 5th-order shaping amplifier with selectable peaking time of 0.5, 1, 2 or 3 μs and overall gain of 4.7, 7.8, 14, or 25 mV/fC), and then digitized by a second custom ADC ASIC (16-channels, 12-bit, up to 2 MHz simultaneous sampling). One FE board contains 8 analog and 8 ADC ASICs and a single Cyclone IV EP4CGX50DF27C7N FPGA that handles the digital stream from the digitizer ASICs. The FPGA serializes the digital data for transmission on a single transceiver line. The result is a digital stream of data from up to 128 detector channels sampled simultaneously at up to 2 MHz. Our detector has 1,000 readout channels, and so we assign a single FE board to 125 adjacent channels (4 FE boards for x and 4 FE boards for y), and select a 1 MHz sampling rate. Each FE board operates independently, sending its data stream to a single back-end at a data generation rate of 2 Gbps.

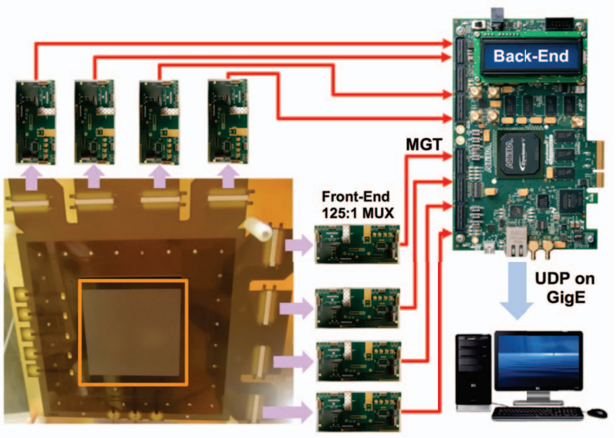

Figure 3. Top-level architecture of our design, showing the eight FE boards that each provide 125:1 multiplexing of detector channels onto a single digital transceiver line (MGT) running at 2.5 Gbps. The BE board handles the 16 Gbps data stream and writes triggered events to disk via UDP on gigabit ethernet (GigE).

B. *Back-End: digital data processing and storage*

The back-end (BE) must receive the digital data stream from eight FE boards (total data generation rate of 16 Gbps), store the data in a circular buffer, apply triggering conditions, and save triggered data to a PC for off-line analysis. We use a COTS Altera Cyclone V GT FPGA development board for the BE ($1,300). The board features a Cyclone V 5CGTFD9E5F35C7 FPGA, 2 HSMC connectors that expose 8 transceiver ports, and 512 MB of DDR3 SDRAM with a ×64 soft memory controller. The development environment is the Quartus II 14.1 Development Suite.

The FE-BE data link is a multi-gigabit transceiver (MGT). An MGT is a serial link that provides low latency, high bandwidth, and low energy cost for FPGA-to-FPGA connections [17]–[19]. Upon arrival at the BE board, data goes through three stages. First, a pattern detector looks for triggering patterns. Next, the serial data is aligned by time and channel number using time-stamp data attached to each data packet by the FE board. Because the block RAM (BRAM) size on the FPGA is not large enough to buffer pre-trigger data from all detector channels, the aligned data is written to off-chip DRAM for temporary storage. Finally,

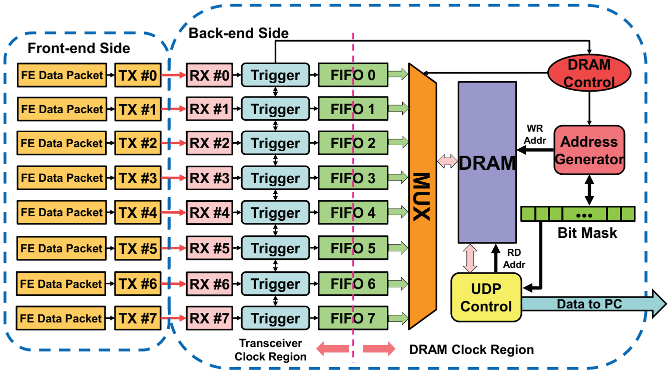

Figure 4. Design block diagram.

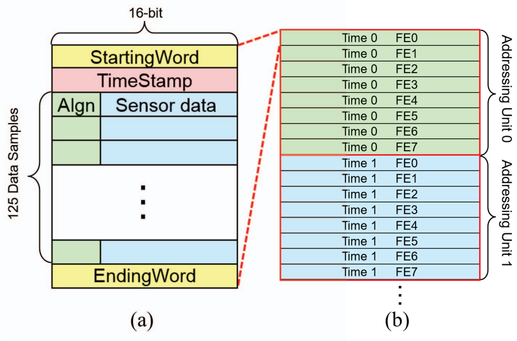

Figure 5. Data organization of a FE packet (a) and in DRAM (b).

triggered data is transferred from DRAM to a PC via UDP on Gigabit ethernet for off-line analysis.

IV. Implementation

As shown in Fig. 4, our design consists of the following parts: front-end data packet generators to organize data on each FE board; send and receive transceiver controllers for FE-BE data exchange; a trigger for event detection; and DRAM control logic for data buffering and replacement.

A. Front-end data packet generation

At a given time the detector data provides an x-y slice of the recoil track as it reaches the readout plane. Repeated sampling gives information about the extent of the track in the z direction.

The ADC ASIC has 12-bit precision, while the transceiver encrypts and serializes 16-bit data each time. We use the extra 4 bits for in-packet indexing. In our current configuration, we simply cycle the index for the 125-channel payload. In this way, data loss during transmission can be easily detected. In addition to the payload, we add three extra 16-bit words for alignment across different packets: a starting word, an ending word, and a sampling time-stamp.

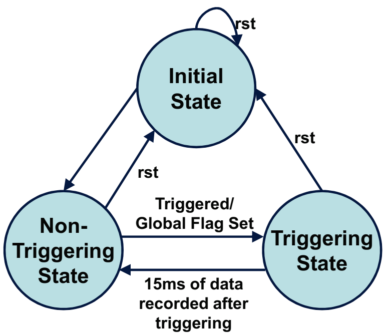
rst
Initial State
rst
rst
Non-Triggering State
Triggered/ Global Flag Set
Triggering State
15ms of data recorded after triggering

Figure 6. Finite State Machine of the event trigger.

B. Transceiver controller

Altera FPGAs support a wide range of protocols and data rate standards [17], [18], though the supported protocols vary by FPGA model. As the sending and receiving sides in our design use different FPGAs, we use a customized communication protocol for our transceivers.

On the FE, we instantiate an ALTGX IP core with a single TX channel and 8b/10b encoding. 8b/10b is a byte oriented binary transmission code that provides DC balance and clock recovery [20], [21]. A single sample from the FE comprises a 128×16-bit data packet. For 1 MHz sampling, the data generation rate on each FE is 2 Gbps; we therefore chose a transceiver rate of 2.5 Gbps.

On the BE, we chose the Native PHY IP, which exposes all low-level MGT control and status signals. An alignment pattern is set in the IP for channel synchronization. We implement eight single-channel Native PHY IP transceivers, one for each FE.

To synchronize the eight FE boards, a system reset signal is generated by a button push on the BE. The signal is propagated to the FE boards via GPIO in a daisy chain. On reset, the FE transceivers prepare the synchronization pattern and the BE transceiver wait for the sync pattern to arrive. After the reset clears, all FE boards will repeatedly send out their alignment words for 1 ms. We have confirmed that this scheme synchronizes all transceiver channels.

C. Event trigger

A design requirement is to record waveforms to disk from channels that satisfy the trigger, as well as on neighboring channels that may not generate a trigger, but that may contain some low level of signal charge that can be recovered with off-line analysis. A neighboring channel is not necessarily digitized by the same FE board. A BE event trigger is implemented on each of the eight receiving transceiver output ports to handle this.

Fig. 6 shows the event trigger finite state machine. When in reset mode, the trigger controller resets all control signals and loads the threshold value. The controller stays in *Initial State* during that time. After the reset clears, the event trigger

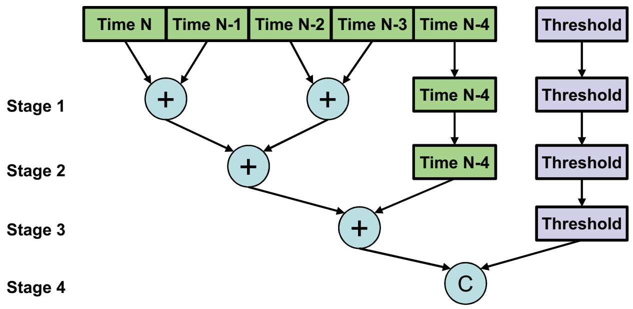

Figure 7. Pipeline for integral-based triggering.

enters the *Non-triggering State* and processes each 16-bit data output from the receiving transceiver. The detector first looks for the starting word of a packet and then records the time-stamp from the following word. The 12 LSB of the payload is then sent to the comparator. Upon triggering, the detector sets a global flag along with the trigger time-stamp, which is checked by all channels every cycle. In this way, all 10^3 detector channels enter the triggered state at the same time. In the *Triggering State*, the BE records 15,000 packets (15 ms at 1 MHz) before returning to the non-triggered state. The BE also sends the trigger time-stamp to the DRAM controller to ensure preservation of 5 ms of pre-trigger data.We implement two different triggering schemes: a per-channel voltage threshold trigger and a per-channel integral trigger.

1) Threshold trigger, individual channel: In this scheme, the detector is triggered when the voltage rises above a fixed threshold on a single channel. To account for different noise levels and offsets on each detector channel, a per-channel threshold is pre-determined from baseline RMS and stored in BRAM. For each FE, a 125×12-bit BRAM module is implemented. As the index for each datum is only 4-bit, we keep a 3-bit external counter in the controller. Whenever an index overflow (jump from 15 to 0) is detected, we increment the external counter. Hence, this combined 7-bit index can address 125 data samples inside a packet. A comparator fetches the threshold values based on that index and makes the comparison.

2) Integral trigger, individual channel: Because the pulse shape on a channel depends on the recoil track geometry in the detector, a better trigger takes into account the integral of the waveform. In our current implementation, we integrate over 5 samples. We therefore use 125 12×5-bit shift registers to buffer the data. When a new sample arrives from the FE, the 60-bit word is updated by shifting new data into the 12 MSB and discarding the 12 LSB. The 60-bit data is sent to a 4-stage pipeline for summation and comparison as shown in Fig. 7.

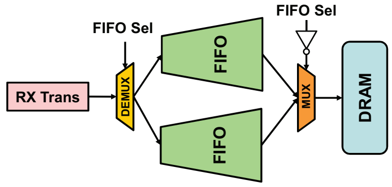

Figure 8. Double buffering between transceiver and DRAM.

D. DRAM controller

To buffer 5 ms of pre-trigger data for 10^3 channels at 1 MHz sampling takes about 10 MB, which is larger than the on-FPGA BRAM capacity. However, the BE Cyclone V development board has four ×16 SDRAM chips, which provide 512 MB of storage. The total data generation rate from the FE is 16 Gbps. At 300 MHz, the theoretical read and write bandwidth is 38.4 Gbps [22], fast enough to write all received data directly to DRAM.

1) DRAM IP and Avalon Bus: In our design, we put the DRAM IP on the Avalon bus, which is an interface that simplifies the communication between modules [23]. The Avalon bus organizes the DRAM into 256-bit words. The 300 MHz DRAM working frequency is different from the transceiver block. Therefore, we implement asynchronous FIFOs between these two modules to deal with the metastability issue [24]. The FIFO write port has 16-bit width, while the output port is 256-bit, which matches the transceiver output port and DRAM interface, respectively. The buffer depth is set to 130 16-bit words to hold an entire packet from a single FE board. For easy data alignment, and to avoid data congestion, a double buffering mechanism is used (Fig. 8). A one-bit global flag is implemented to bias the read and write FIFO. Whenever the transceiver detects a FE packet header, the FIFO selection flag is reversed to write to the other FIFO, while the FIFO that was previously being written to will be used as input for the DRAM.

2) Double buffering: Writes to DRAM are organized by sample time and aligned by the FE board number-of-origin (Fig. 5b). We use the FE packet as the DRAM read and write unit. The 512 MB DRAM space is pre-allocated into 2^{21} slots, each slot holds eight 256-bit words to fit a single FE packet. When an entire FE data packet is written into the related FIFO and a ready signal is received, the DRAM controller enables the address generator to send a starting address based on the time-stamp and board number pointing to the pre-allocated space for that packet.

3) Address generation: At a 16 Gbps data generation rate, 512 MB DRAM can only hold ~250 ms of data. We must therefore reuse the DRAM space, especially at low trigger rates. A replacement mechanism is implemented inside the DRAM address generator. From the address generator point of view, the DRAM is divided into units of eight FE packets that hold data sampled at the same time, resulting in a total of 2^{18} units. As shown in Fig. 5b, FE packets with the same color belong to the same addressing unit. A hash table is used to provide within-unit address offsets based on the FE board number. The reasoning for this organization is as follows: we need data from all channels sampled during an event trigger. Organizing data based on sample time both simplifies the replacement algorithm and minimizes DRAM addressing latency.

4) Replacement mechanism: A 1-bit address mask is assigned to each of the above units. If data is determined to be useful by the event trigger, then the related mask bit is set so all the data sampled at the same time will be kept. When the address generator reaches the end of the DRAM address space, the new incoming data set is again assigned to address unit 0. Before that address is assigned, the mask bit is examined. If the bit is set, then the controller looks for the next unset mask bit and uses that related address as the starting point for the following data sets. In this way, we can keep the useful triggering data while fully reusing the non-triggering part.

We record 20 ms of data for each trigger (5 ms pre-trigger and 15 ms post-trigger), or 16 Gbps×20 ms = 0.32 Gbit of data per trigger (assuming all channels are saved). Our DRAM capacity is 4 Gbit, so we can store data from up to 12 triggering events.

E. Synchronization across different boards

Our proposed system consists of nine FPGA boards. We must synchronize the ADCs on the FE boards to ensure simultaneous sampling of all detector channels. To do so, we propagate a low-frequency (125 MHz) clock from the BE to all FE boards using a stand-alone 1:8 clock fan-out.

V. Evaluation

We evaluate three aspects of our design. First, we present the FPGA resource utilization. Next, we evaluate the performance of the transceivers (speed) and DRAM (read & write bandwidth). Finally, we present event trigger evaluation results for several sample waveforms using the threshold and integral trigger schemes described in Sec. IV.

BNL has loaned us two FE boards for our demonstration system. All FE boards share the same design. All the necessary modules for eight FE boards–such as receive transceivers, threshold detectors, and FIFOs–are instantiated on the BE board and related pins are assigned. Our demonstration system thus also serves for verification.

A. FPGA resource utilization

The resource utilization of the FE and BE FPGAs is shown in Table I. The FE utilization is the same for all FE boards as the design is common to all. The BE utilization includes all modules required to interface with 8 FE boards.

Table I
Resource utilization for the FE and BE FPGAs.

FE	ALUTs	Registers	Memory Bits	TX Channel	PLL
Usage	6300	3753	33492	1	3
%	13%	NA	1%	13%	38%
BE	ALUTs	Registers	Memory Bits	RX Channel	PLL
Usage	6936	9334	499232	8	1
%	6%	NA	4%	67%	5%

B. Performance evaluation

In this section, we test the performance of transceivers and DRAM independently.

1) Transceiver data rate: The Altera Cyclone V GT development board comes with a Board Testing System which provides a GUI displaying key parameters and a transceiver reference design. Using that reference design as a baseline, we expose two RX pins to the TX pin from our two FE boards. Further modifications are made by enabling 8b/10b encoding to match our FE transceiver design. Using the board testing GUI we verified that the data rate on each transceiver channel is **2.5 Gbps**, which meets the design requirement to handle the 2 Gbps data generation rate of each FE board.

2) DRAM write bandwidth: Similar to our transceiver testing, a DRAM read and write reference design is also provided. In that project, the DRAM controller keeps sending read and write requests independently. Using that reference design, the returned write bandwidth is 17.6 Gbps, which meets our 16 Gbps design goal. In the above test, DRAM is being read and write simultaneously, which affects the bandwidth. In our implementation, the DRAM receive mostly write requests, so the actual write bandwidth in our implementation is higher. We confirm that all the received data are written to DRAM without congestion.

C. Event trigger evaluation

We pre-load simulated data onto the two FE board FPGAs. These data represent low-energy and high-energy events. Each set of sample data contains 2000 data points stored in BRAM. We load one set of data onto each of the FE boards, and read out one sample every microsecond for transmission to the BE via the MGT. On receipt at the BE, the data passes through the event trigger and is then written to DRAM. Here, we add extra logic to also store the data into 2 special BRAM modules dedicated for this test, one for each FE board, upon trigger. Using the data sample ID (essentially a counter for incoming samples) that initiates the trigger, we locate the pre- and post- trigger data in BRAM. Once the full 2000-sample waveform from each FE board has been transmitted and processed, we read out data from BRAM using the *In Memory Content Editor* tool in the Altera Quartus II development suite.

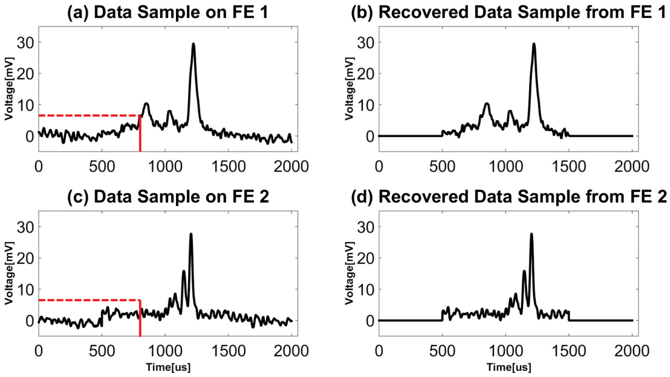

Figure 9. Voltage threshold trigger results. (a) & (c) are the pre-loaded sampled data. (b) & (d) are the triggered data as read out from the BE BRAM module implemented for this test.

We first test the voltage threshold trigger with the threshold set to 7 mV. Due to the fact that our sample data has only 2000 data points (2 ms of sampled data), we set the pre- and post-trigger time windows to 0.3 and 0.7 ms, respectively. Samples outside of this window are set to zero. Fig. 9 presents the results of this test. We see that the event is triggered when data from FE 1 reaches 7 mV at $t = 790\,\mu$s. Both FE 1 and FE 2 are read out over the same time period, even though data from FE 2 did not initiate the trigger. Thus we confirm that our demonstration system meets the requirements of keeping pre- and post-trigger data from adjacent channels when one channel triggered.

We also evaluate the integral trigger design. We use a low-energy set of waveforms for this test and replace the threshold trigger in the previous test with the integral one. We set the integral threshold value to 10 mV μs. The original and integrated signals, as well as the triggered data, are shown in Fig. 10. We see that this narrow integral filter essentially serves as a low-pass filter, and suppresses the high-frequency noise. Because the integral window size is small ($N_{samples} = 5$) compared to the widths of the peaks (about 512 samples total), the peaks are not integrated. By expanding the integral window size, it would be possible to integrate the signal under all three peaks. The bottom plot in Fig. 10 shows a simulated integral waveform with $N_{samples} = 512$. Implementing such a design on the FPGA requires buffering more data in BRAM and an enlarged adder tree (Fig. 7) with $\log_2 N_{samples} + 1$ stages (10 stages for $N_{samples} = 512$). These factors will lead to an linear growth of on-chip resource usage and incur larger delay. We will explore this in future work.

VI. Conclusion and Future Work

In this paper, we describe an FPGA-based DAQ for directional dark matter detection. The design features eight FE ASIC+FPGA boards for data collection, and a single digital BE FPGA board for data pre-processing and temporary storage. The final system will be capable of processing 10^3 detector channels, each sampled at 1 MHz. Although our demonstration system currently contains two FE boards and one BE board, all the necessary control logic is implemented for our final system. Thus the expansion to eight FE boards is a straightforward extension of our two-FE-board prototype. Using this system, we have evaluated several key parameters, including transceiver data rate, DRAM read and write speed, and triggered data quality. These confirm that our system meets our design goals.

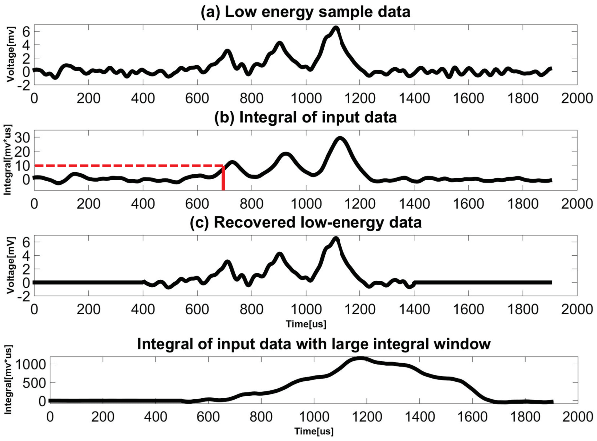

Figure 10. Integral trigger results. (a) preloaded low-energy sampled data, (b) integral filtering of the input data, and (c) triggered data. At bottom is the input data filtered with a 512-point integral window.

Looking forward, we will expand the system to include all eight FE boards. We will also implement more advanced event triggers, including an expanded integral window trigger, and one that computes the integral over multiple detector channels (*e.g.* a channel and its four nearest neighbors).

Acknowledgements

This work was made possible by the extensive support of members of the Brookhaven National Laboratory, especially Hucheng Chen, Gianluigi de Geronimo and Jack Fried. JBRB acknowledges the support of the National Science Foundation (EAGER PHY-1649966), the Research Corporation Cottrell College Science Award (Award #23325) and the Sloan Research Fellowship (BR2012-011). CY and CN acknowledge the support of the Massachusetts Space Grant Consortium (NNX16AH49H). We also thank James MacArthur of the Harvard Electronic Instrument Design Laboratory and James Phillips for their continued support.

References

[1] P. A. R. Ade *et al.*, “Planck 2015 results. XIII. Cosmological parameters,” *Astron. Astrophys.*, vol. 594, p. A13, 2016.

[2] G. Bertone, D. Hooper, and J. Silk, “Particle dark matter: Evidence, candidates and constraints,” *Phys. Rep.*, vol. 405, pp. 279–390, 2005.

978-1-5386-3472-1/17/$31.00 ©2017 IEEE

[3] F. Mayet *et al.*, "A review of the discovery reach of directional Dark Matter detection," *Phys. Rept.*, vol. 627, pp. 1–49, 2016.

[4] D. N. Spergel, "The motion of the Earth and the detection of WIMPs," *Phys. Rev.*, vol. D37, p. 1353, 1988.

[5] J. B. R. Battat *et al.*, "Readout technologies for directional WIMP Dark Matter detection," *Phys. Rept.*, vol. 662, pp. 1–46, 2016.

[6] J. N. Marx and D. R. Nygren, "The Time Projection Chamber," *Phys. Today*, vol. 31N10, pp. 46–53, 1978.

[7] J. B. R. Battat *et al.*, "Low Threshold Results and Limits from the DRIFT Directional Dark Matter Detector," *Astropart. Phys.*, vol. 91, pp. 65–74, 2017.

[8] D. P. Snowden-Ifft, "High Precision Measurements of Carbon Disulfide Negative Ion Mobility and Diffusion," *Rev. Sci. Instrum.*, vol. 84, p. 053304, 2013.

[9] N. S. Phan, R. Lafler, R. J. Lauer, E. R. Lee, D. Loomba, J. A. J. Matthews, and E. H. Miller, "The novel properties of SF_6 for directional dark matter experiments," *JINST*, vol. 12, no. 02, p. P02012, 2017.

[10] C. Adams *et al.*, "LAr1-ND: Testing Neutrino Anomalies with Multiple LAr TPC Detectors at Fermilab," 2013.

[11] T. Alexopoulos, J. Burnens, R. de Oliveira, G. Glonti, O. Pizzirusso, V. Polychronakos, G. Sekhniaidze, G. Tsipolitis, and J. Wotschack, "A spark-resistant bulk-micromegas chamber for high-rate applications," *Nucl. Instrum. Meth.*, vol. A640, pp. 110–118, 2011.

[12] R. Orito, O. Sasaki, H. Kubo, K. Miuchi, T. Nagayoshi, Y. Okada, A. Takada, A. Takeda, T. Tanimori, and M. Ueno, "Development of an ASD IC for the micro pixel chamber," *IEEE Transaction on Nuclear Science*, vol. 51, no. 4, pp. 1337–1342, 2004.

[13] J. Richer, O. Bourrion, G. Bosson, O. Guillaudin, F. Mayet, and D. Santos, "Development and validation of a 64 channel front end ASIC for 3D directional detection for MIMAC," *Journal of Instrumentation*, vol. 6, no. 11, p. C11016, 2011.

[14] D. Akerib, H. Araujo, X. Bai, A. Bailey, J. Balajthy, P. Beltrame, E. Bernard, A. Bernstein, and et al., "FPGA-based Trigger System for the LUX Dark Matter Experiment," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 818, pp. 57–67, 2016.

[15] O. Guillaudin, J. Billard, G. Bosson, O. Bourrion, T. Lamy, F. Mayet, D. Santos, and P. Sortais, "Quenching factor measurement in low pressure gas detector for directional dark matter search," *EAS Publ. Ser.*, vol. 53, pp. 119–127, 2012.

[16] D. P. Snowden-Ifft, "Discovery of multiple, ionization-created CS2 anions and a new mode of operation for drift chambers," *Rev. Sci. Instrum.*, vol. 85, p. 013303, 2014.

[17] Altera. Cyclone V Device Handbook vol. 2: Transceivers. [Online]. Available: https://www.altera.com/ content/dam/altera-www/global/en_US/pdfs /literature/hb/cyclone-v/cv_5v3.pdf

[18] ——. Altera Transceiver PHY IP Core User Guide. [Online]. Available: https://www.altera.com/ content/dam/altera-www/global/en_US/ pdfs/literature/ug/xcvr_user_guide.pdf

[19] J. Sheng, C. Yang, and M. Herbordt, "Towards Low-Lantency Communication on FPGA Clusters with 3D FFT Case Study," *Proc. Highly Efficient and Reconfigurable Technologies*, 2015.

[20] A. Widmer and P. Franaszek, "A DC-balanced, partitioned-block, 8B/10B transmission code," *IBM Journal of research and development*, vol. 27, no. 5, pp. 440–451, 1983.

[21] D. Wentzlaff and A. Agarwal, "A Quantitative Comparison of Reconfigurable, Tiled, and Conventional Architectures on Bit-level Computation," *Proc. Field-Programmable Custom Computing Machines*, pp. 289–290, 2004.

[22] Altera. Cyclone V GT FPGA Development Board Reference Manual. [Online]. Available: https://www.altera.com/ content/dam/altera-www/global/en_US/pdfs/ literature/manual/rm_cvgt_fpga_dev_board.pdf

[23] ——. Avalon Interface Specifications. [Online]. Available: https://www.altera.com/ content/dam/altera-www/global/ en_US/pdfs/literature/manual/ mnl_avalon_spec.pdf

[24] Y. Zhang, C. Yi, J. Wang, and J. Zhang, "Asynchronous FIFO Implementation using FPGA," *Electronics and Optoelectronics (ICEOE), 2011 International Conference on*, vol. 3, pp. V3–207, 2011.