

# A Field-Programmable Gate Array (FPGA) TDC for Fermilab SeaQuest (E906) Experiment and Its Test with a Novel External Wave Union Launcher

Su-Yin Wang, Jinyuan Wu, Shi-Hong Yao, Wen-Chen Chang

**Abstract**—We develop a field-programmable gate array (FPGA) TDC module for the tracking detectors of Fermilab SeaQuest (E906) experiment, including drift chambers, proportional tubes and hodoscopes. This 64-channel TDC module has 6U VMEbus form factor and is equipped with a low-power and radiation-hardened Microsemi ProASIC3 Flash based FPGA. The design of new FPGA firmware (Run2-TDC) aims to reduce the data volume and data acquisition (DAQ) deadtime. The firmware digitizes multiple input hits of both polarities while allowing users to turn on a multiple-hit elimination logic to remove after-pulses in the wire chambers and proportional tubes. A scaler is implemented in the firmware to allow for recording the number of hits in each channel. The TDC resolution is determined by an internal cell delay of 450 ps, and a measurement precision of 200 ps has been achieved. We use 5 kinds of tests to ensure the qualification of 93 TDCs in mass production. We utilize the external wave union launcher in our test to improve the TDC's measurement precision and also illustrate how to construct the Wave Union TDC using an existing multi-hit TDC without modifying its firmware. Measurement precision is improved by about a factor of two (108 ps) in data based on four-edge wave union. Even better improvement of measurement precision (69 ps) was achieved by combining the approaches of Wave Union TDC and multiple-channel ganging.

## I. INTRODUCTION

THE Fermilab E-906/SeaQuest experiment is the latest one of a series of fixed target Drell-Yan experiments. The Drell-Yan process is known as the production of dilepton pair via the annihilation of quark and antiquark from colliding hadrons. It has been a clean and important approach to explore the quark and antiquark structure of nucleons. Utilizing the high-intensity 120-GeV proton beam from the Main Injector of Fermilab, the SeaQuest experiment is able

Manuscript received May 4, 2013. This research is supported by National Science Council of Taiwan (Grant No. 100-2112-M-001-015-MY3) and operated by Fermi Research Alliance, LLC under Contract No. De-AC02-07CH11359 with the United States Department of Energy.

Su-Yin Wang is with Institute of Physics, Academia Sinica, Taipei 11529, Taiwan, and Particle Physics Division, Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA. She is also a Ph.D. student from Department of Physics, National Kaohsiung Normal University, Kaohsiung 824, Taiwan. (telephone: +1-630-318-0225, e-mail: grass@phys.sinica.edu.tw)

Jinyuan Wu is now with Fermi National Accelerator Laboratory, Batavia, Illinois 60510, USA. (telephone: +1-630-840-8911, e-mail: jywul68@fnal.gov).

Shi-Hong Yao is now with Institute of Physics, Academia Sinica, Taipei 11529, Taiwan. (telephone: +886-2-27898944 email: shyao@phys.sinica.edu.tw)

Wen-Chen Chang is now with Institute of Physics, Academia Sinica, Taipei 11529, Taiwan. (telephone: +886-2-2789-6794, email: changwc@phys.sinica.edu.tw)

to determine the antiquark structure of the nucleon in the large momentum-fraction region and the modifications of this structure inside the nucleus [1], [2]. The detector system of this experiment is composed of drift chambers, proportional tubes and scintillating hodoscopes. The analog signals of the energy loss of charged particles in the detectors are digitized with time-to-digital converter (TDC) modules as shown in Fig. 1a. The requirement of time resolution is 4 ns. The time measurement is performed by the delay of logic elements (VersaTile) inside the ProASIC3 FPGA (A3P1000) [3]. During the first run of the SeaQuest experiment in 2012, a latch-type FPGA firmware was used for TDC function and the time resolution was 2.5 ns. Due to lack of zero suppression the large data size caused significant DAQ deadtime.

In this paper we report the upgrading of TDC FPGA firmware, “Run2-TDC”, which effectively reduces the data volume and thus improves the DAQ efficiency. In addition we implement scalers, coarse-time counter and external wave union launchers to accomplish the important features of multiple-hit measurement and the improvement of TDC measurement precision.

## II. HARDWARE SPECIFICATION

### A. Form Factor and PCB

The form factor of TDC module complies with VME 6U standard for the mechanical specifications. It is equipped with an input of 64 channels via four flat cables and 3 NIM/TTL LEMO I/O on the front panel. The low-voltage power supply is +5.0 V and -12 V provided by the VME backplane. The PCB is made of FR-4 material with 1.6 mm thickness and double side assembly. An 8-layer board stack-up is used for the PCB design so that the front-end circuit has better capability of EMI rejection.

### B. Input and Output

The input section contains 16 LVDS transceivers (Texas Instruments, SN65LVDT348) to interface with the differential digital signals from the front-end electronics. The transceiver chip can handle a variety of differential and single-ended logic levels, e.g. ECL, LVDS, LVECL, and PECL as the input and provides LVTTTL for the output logic. Each transceiver channel is integrated with a 110-ohm line termination resistor to match impedance with the twisted-pair flat cable (3M, 1700 series). The buffer receives differential signals from the cable and transmits them to signal-end FPGA inputs (Actel, A3P1000).

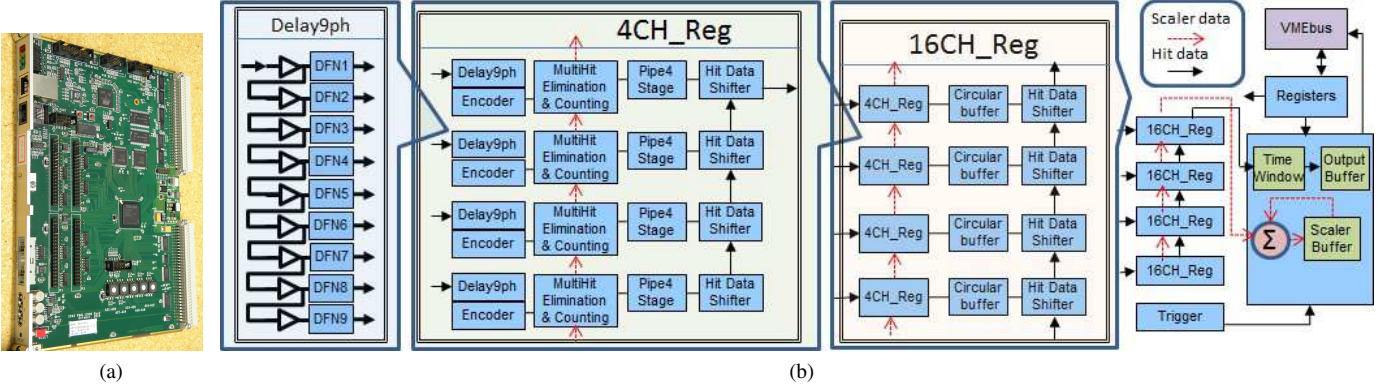


Fig. 1: (a)Photograph of the 64-channel TDC module. (b)Block diagram of the TDC firmware.

In addition, we pay special attention to the layout to ensure that the traces length (or propagation delay) between the LVDS buffer and FPGA were no more than 1 mm to minimize the jittering of arrival time.

### C. FPGA and Clock

The core chip in this TDC module is the Actel ProASIC3 FPGA (A3P1000) [3], which is the most advanced device among the ProASIC3 series. The main features of this FPGA device are as follows: 1 M system gates, 144 kbits dual-port SRAM, one integrated PLL, 300 user-defined I/O pins and 350 MHz system performance. This on-board FPGA is incorporated with three major off-chip silicon resources: the 64-channel front-end transceivers, memory chips (IDT, 32K x 16-bit, IDT70V27) and interfaces to the 32-bit VME back-plane. To relieve the bandwidth bottleneck of these circuits, a large portion of pin count is used in the FPGA to provide most of the throughput advantage. The LVTTTL (Low-voltage TTL) logic is selected for all FPGA I/O buffers and the drive current is limited to  $\pm 6$  mA for better quality of signal integrity.

For the development of the FPGA firmware, Libero IDE 9 design software is used for the synthesis, simulation and timing analysis. The binary design file is downloaded to the flash-based FPGA by connecting FlashPro4 programmer to the on-board JTAG port.

The system clock source is a flash-programmable clock generator (Cypress, CY22394). It is the output clock used for the real-time-clock (RTC), Ethernet synchronization and a differential LVPECL clock for PLL inside the FPGA. The PLL is locked to a reference 135 MHz LVPECL clock source which provides a low peak-to-peak period jittering performance of 200 ps.

## III. CONCEPT AND METHODS

### A. TDC Firmware Overview

The block diagram of the Run2-TDC firmware is shown in Fig. 1b. The input hit in each channel propagates in a delay chain [4] of 9 taps with a nominal delay of 450 ps/tap as shown in the block “Delay9ph”. The delay pattern is registered every clock cycle at 250 MHz and encoded into the fine-time code.

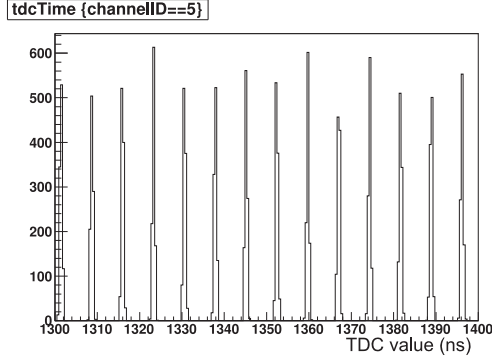
If a valid hit is detected, the fine-time code (a hexadecimal value from 0 to 8) along with the coarse-time count (with a least significant bit equal to 4 ns) are temporarily stored in the first-layer buffer (Pipe4) with up to 4 hits per channel. The stored hits are constantly read out at the speed of 62.5 MHz and put into a block of memory shared by a group of 4 channels.

The memory is organized as 2, 4, or 8 pipelines (circular buffers) with a user-select length of 2048, 1024 or 512 ns to store hits waiting for the trigger selection. If a trigger arrives, the writing pointer is directed to the address of the next circular buffer and the current circular buffer is read out by copying valid hits into the event buffer. During the read out time or copy in progress (CIP) time, it is allowed to record only hits within a preset time window to accommodate various detectors with different sizes and drift time. The TDC data of hits stored in the event buffer are read out by the DAQ system via the VMEbus interface.

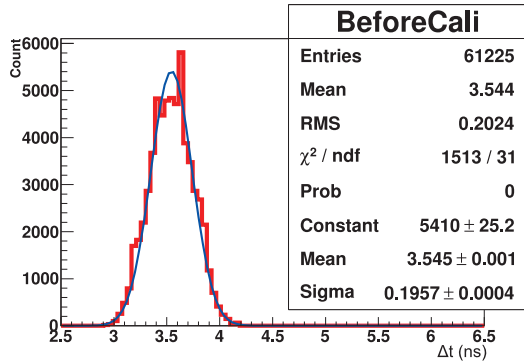
### B. Measurement Precision

We determine the measurement precision of TDC by obtaining the standard deviation of the distribution of difference between the measured timing delay of two consecutive pulses and the true value. Nevertheless the results may not reflect the actual measurement ability of the TDC suppose that the delay between the two pulses is a fixed value such as that generated by a cable. In avoid of this problem, it is necessary to use pulses with varying timing delays.

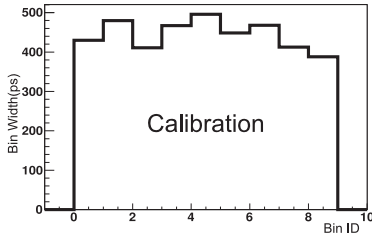
To perform such measurement, we use a commercial FPGA module (CAEN V1495) as the pulser to generate the hit and stop signals to the TDC. The 40 MHz clock on the V1495 board is converted into 137 MHz ( $40 \times 24/7$ ) to drive all counters in the FPGA every 7.2917 ns. In this way, various timing delays are generated and used in the measurement. For each event, the relative timing delay between the hit and the stop signals is given in Eq. 1 where  $\Delta t$  is the time difference between the first pair of the hit and stop signals and  $n$  is the event number. The TDC time distribution is shown as Fig. 2a. There are a total of 96 possible timing patterns between the



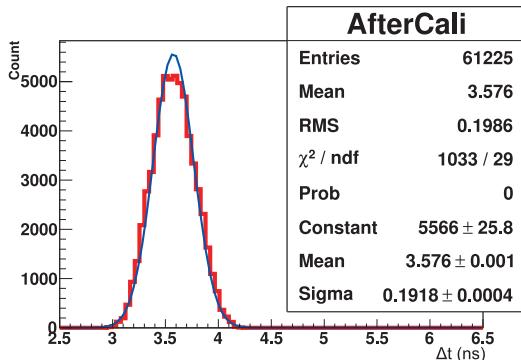
(a) The TDC time of a hit is the time difference between its arrival time and the stop signal. For each event, the measurement of the TDC time begins with a hit signal which is followed by a stop signal with a step increasing by 7.2917 ns in each subsequent event. This histogram shows the distribution of TDC time in the range of 1300-1400 ns.



(b) Distribution of the difference of the measured time delay between two pulses and the true delay value,  $\Delta t$ , before calibration. By using the modulo of time difference method,  $\Delta t$  is calculated by the TDC time. With Eq. 1,  $\Delta t$  is shown in a histogram of the 7.2917-ns width in time and overlaid with a Gaussian fit.



(c) Calibration result.



(d) Distribution of  $\Delta t$ , after bin-width calibration. The  $\chi^2$  of a Gaussian fit decreases by 30% and a 191 ps time precision is seen.

hit and stop signals considered in the 250 MHz clock domain.

$$\begin{aligned}
 & \text{Mod}(\text{TDC}(n), 7.2917) \\
 &= \text{Mod}(\Delta t + n * 7.2917, 7.2917) \\
 &= \text{Mod}(\Delta t, 7.2917) \\
 &= \Delta t
 \end{aligned} \tag{1}$$

To extract the time difference  $\Delta t$  between the first signal pair, the modulo of time difference method, as shown by Eq. 1 is utilized. One can fill  $\Delta t$  in a histogram with a full scale window of 7.2917 ns to see its distribution as shown in the Fig. 2b overlaid with a Gaussian fit.

In our TDC, there are nine delay taps and their actual widths of timing bin may not be identical. Therefore, to achieve high precision measurement, it is necessary to calibrate bin by bin. The TDC coarse-time bin is 4 ns and there are nine fine-time bins in one coarse-time bin. First the time distributions of events is represented by a histogram with fine-time bin. Then we normalize the total event to 4000 as shown in Fig. 2c where the y axis represents the width of each fine-time bin in the unit of ps. After bin-width calibration, the  $\chi^2$  of Gaussian distribution has decreased by 30% and Fig. 2d shows that 191 ps time precision can be achieved.

### C. Scaler

In almost all applications, it is highly desirable to implement a scaler for each TDC channel as a tool for deadtime estimation or quick detector checking. The conventional method of implementing a scaler consumes a large amount of FPGA silicon resource. Therefore it is difficult to pair a scaler with each TDC channel. We make an innovation by developing a new scheme with reduced resource consumption that allows us to fit 64 scalers with the 64 channels. The resolution of each scaler is 16 ns, and there is an 8-bit counter for each channel.

These counters are synchronously read into the first-layer buffer and then reset. The 64 first-layer buffers then pass scaler data one at a time, taking 2048 ns in total to read out all channels. Within this 2048 ns time period, a channel can receive a maximum of 128 hits without the overrolling of the 8-bit counter. In the final buffer stage, a 32-bit adder is used to accumulate the output from the 8-bit counter for each channel and store the value into a 32-bit buffer, so that the scaler can count up to  $2^{32}$ . The buffer is organized as 8 pages with 64 channels for each page. The user may switch to a specific scaler page by setting a register accessible via the VMEbus.

### D. Coarse-Time Counter (TC) Implementation

The ProASIC3 flash based FPGA family is relatively slow in term of clock speed. A frequency of 250 MHz for the delay pattern register clock is used to reduce delay line length and encoder size so that 64 channels can be fit into a low-cost device. However, as many authors in this field have pointed out, it is very difficult to implement other logic blocks at the same 250 MHz, especially the coarse-time counter.

Fig. 2: Determination of TDC measurement precision

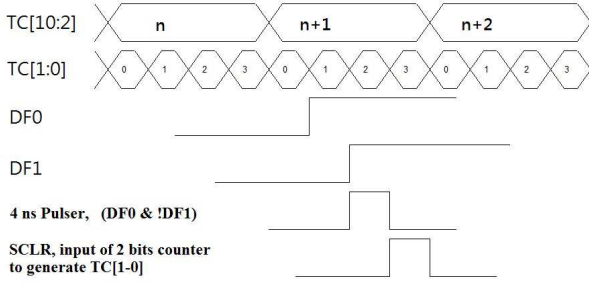


Fig. 3: Coarse-time counter implementation

To overcome this problem, we implemented the higher bits of the coarse-time counter, TC[2-10] with 62.5 MHz, and lowest two bits of it, TC[0-1] with 250 MHz, as illustrated in Fig. 3. To synchronize the lowest two bits of the coarse-time counter, a 4 ns pulse is used to reset the counter running at 250 MHz. The reset pulse is made by an AND gate and three D-flip-flops (DF) in the 250 MHz clock domain. The first flip-flop (DF0) catches the rising edge from TC[2], which flips every 16 ns. The second flip-flop (DF1) catches output from the DF0, so it will be two clocks later in comparison with TC[2]. The inputs of the AND gate are DF0 and inverse of DF1. The output of the AND gate goes to the third flip-flop, and its output (SCLR) is used as a synchronous reset signal of the two-bit counter TC[0-1]. In this way, only two bits of the coarse-time counter are running with a 250 MHz clock.

#### E. Multiple Hits and Multi-hit Elimination

The charged tracks are likely to generate multiple hits passing through chambers with a large volume. It requires the multiple-hit detecting ability of the TDC. In real operation however, it is desirable to capture only the first transition edge of a hit while eliminating pulses within a pre-defined period of time to reduce total data volume. The function of multi-hit elimination is implemented in the firmware and the related parameters could be changed by a register accessible via the VMEbus.

There is a 250 MHz clock used by a 6-bit counter which counts up to the user-selected elimination time window. The user may choose to enable the function of multi-hit elimination or not. If the multi-hit elimination is turned on, the user-select multi-hit elimination time window will be applied. The Run2-TDC will record only the first hit while subsequent hits within the multi-hit elimination time window in the same channel will be excluded. The window of multi-hit elimination may be adjusted to range from 16 ns to 272 ns in a 4 ns step.

There are also two user-select modes of multi-hit elimination: “non-updating mode” and “updating mode”. The user-select multi-hit elimination time window begins immediately after the first hit. In the non-updating mode, a constant duration of the multi-hit elimination window is set during which the received hits are not recorded. On the other hand the multi-hit elimination window will be re-initiated in the updating mode if any hit is received. That means that the multi-hit elimination time window is extended because the TDC receives numbers

of hits more than expected in the multi-hit elimination time window.

#### F. Test of the External Wave Union Launcher

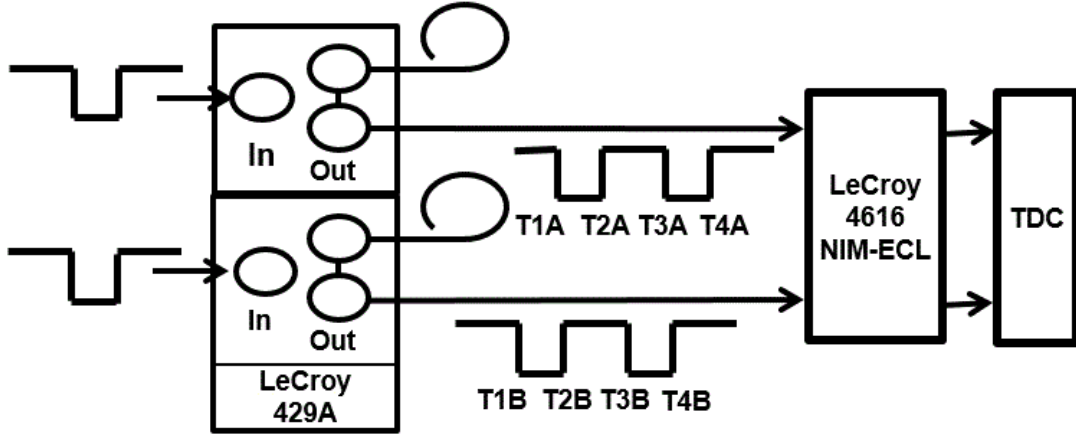
A wave union launcher creates a pulse train so that the TDC can perform multiple measurement to achieve better precision. The wave union method was first introduced and demonstrated in Ref. [5] and further detail can be found in Ref. [6], [7]. Wave union launchers implemented inside the FPGA require careful redesign of the TDC firmware. Below we demonstrate how to implement the Wave Union TDC using an existing multi-hit TDC without changing FPGA firmware.

The launcher is constructed with a LeCroy 429A logic-fan-in/out unit and an open-ended delay cable, which is connected via a T-connection to the main line as shown in Fig. 4a. When a pulse is sent into the unit, a double pulse is generated due to cable reflection. The TDC records four transition time of both the rising and falling edges of the two pulses for each input event. Two wave union launchers generate two double pulses which are fed into two sets of TDC channels, 8 channels per set for digitization.

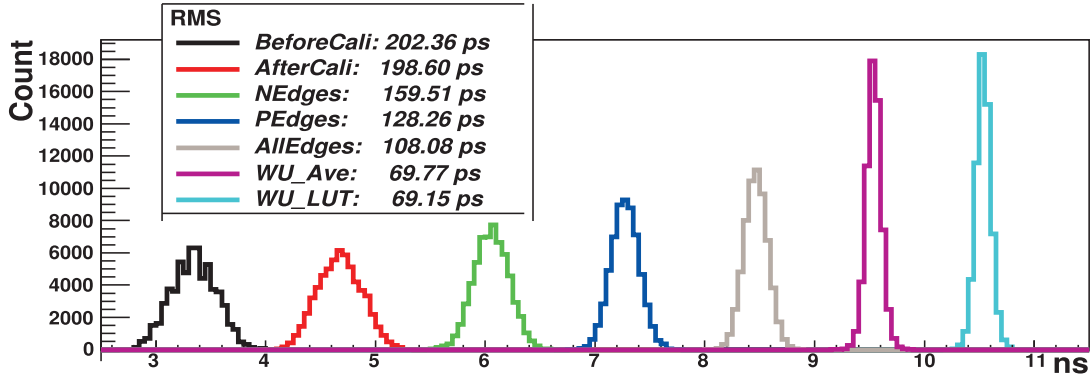
Test results of the Wave Union TDC are shown in Fig. 4b. Arbitrary offsets are added to these histograms so that they can be plotted together. These histograms represent the same data, but are analyzed by different methods. The distributions of the time difference of the two leading edges (T1A-T1B) are plotted in the two left-most histograms. The ones marked with “BeforeCali” and “AfterCali” are the results before and after a bin-by-bin calibration, respectively. These results are done with the measurement of a single transition edge.

With more wave union edges used for time digitization, the measurement precision is improved. The histogram marked with “NEdges” represents the difference of the average times of two trailing edges  $((T2A+T4A)/2 - (T2B+T4B)/2)$ , and similarly “PEdges” represents  $((T1A+T3A)/2 - (T1B+T3B)/2)$ . The leading edges have a faster propagation speed in the TDC delay line, and therefore their measurement precision is better than that of the trailing edges. The “AllEdges” histogram includes all four edges in the average  $((T1A+T2A+T3A+T4A)/4 - (T1B+T2B+T3B+T4B)/4)$  and has even better measurement precision, as expected. Clearly this simple four-edge wave union TDC improves the measurement precision from about 200 ps to 108 ps without any change in the TDC firmware and any significant increase of system cost.

The histograms marked with “WU\_Ave” and “WU\_LUT” are results of combining data from all 16 channels (8 channels per set). Both methods use the difference between the averages of each set, as in TABLE I. For each channel, WU\_Ave takes an average of 4 hits of both signal and trigger hits, and a 69.77 ps time resolution can be achieved. In the Ref. [8], it points out the look-up-table (LUT) method may improve the calibration. WU\_LUT is made by a 2048-value look-up-table which is made by 11 bits (Table II). In the Table II, the  $T_{f1}$  indicates the fine-time value of first hit and the  $T_{c2}$  indicates the coarse-time value of the second hit, etc. The summation of the fine time of 4 hits is recorded in the first 5 binary bits



(a) Schematic of the external wave union launcher



(b) Test results of the Wave Union TDC.

Fig. 4: Wave Union TDC

TABLE I: Legends of Fig. 4b

Label	Decoder	TDC value
BeforeCali	Before calibration	$T1A - T1B$
AfterCali	After calibration	$T1A - T1B$
NEdges	All negative (trailing) edges	$((T2A + T4A)/2 - (T2B + T4B)/2)$
PEdges	All positive (leading) edges	$((T1A + T3A)/2 - (T1B + T3B)/2)$
AllEdges	All edges	$((T1A + T2A + T3A + T4A)/4 - (T1B + T2B + T3B + T4B)/4)$
WU_Ave	Wave union, decode by average method	$\left\{ \sum_{i=1}^8 A\_AllEdges(i) - \sum_{i=1}^8 B\_AllEdges(i) \right\} / 8$
WU_LUT	Wave union, decode by look up table method	$\left\{ \sum_{i=1}^8 LUT_A(i) - \sum_{i=1}^8 LUT_B(i) \right\} / 8$

TABLE II: LUT: 11-bit look up table

Bit	unit	equation
4-0	Fine time	$T_f1 + T_f2 + T_f3 + T_f4$
10-5	Coarse time	$(T_c2 - T_c1) + (T_c3 - T_c1) + (T_c4 - T_c1)$

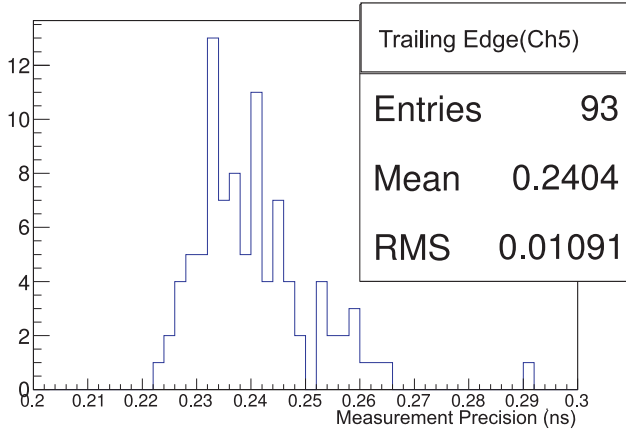
(0-4). The summation of each coarse time subtracts the coarse time of first hit that is recorded in the binary bits (5-10). A 2048-value look-up-table for a calibration of each bin size is made. By averaging both signal and trigger hits, a 69.15 ps time resolution can be achieved.

#### IV. RESULTS OF PERFORMANCE

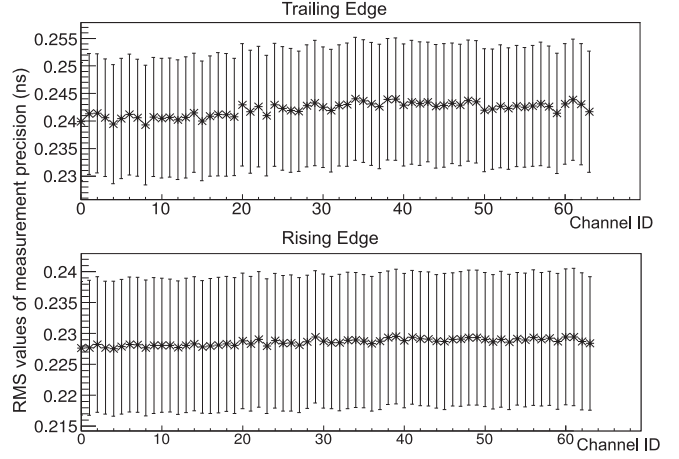
Before the installation of 93 TDC modules in the experiment, we did a series of systematic production tests on them.

A DAQ system was built based on “CODA” [9], which is the DAQ framework developed for the experimental physics program at JLab, U.S. and also the main DAQ system in the SeaQuest experiment. A CAEN V1495 board is programmed as a pulser to produce the signals for the production test. The 40 MHz clock on the V1495 board is converted into 137 MHz ( $40 \times 24/7$ ) to drive all counters in the FPGA (7.2917 ns). The frequency does not match any frequency running on the TDC module so we can use it as random hits in some tests. The DAQ system record TDC hits as an event. We repeat these tests for 2000-78000 events to check for the reproducibility.



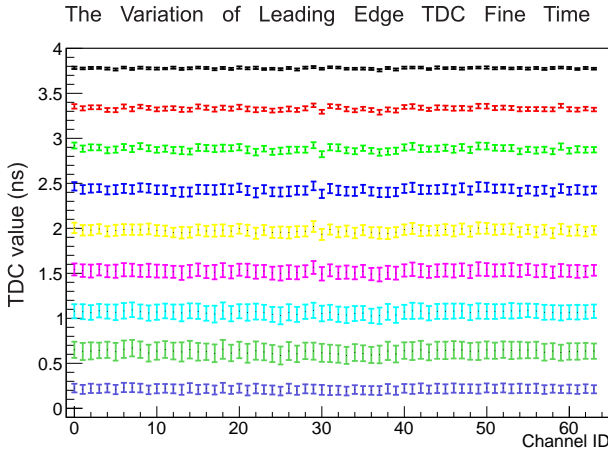


(a)

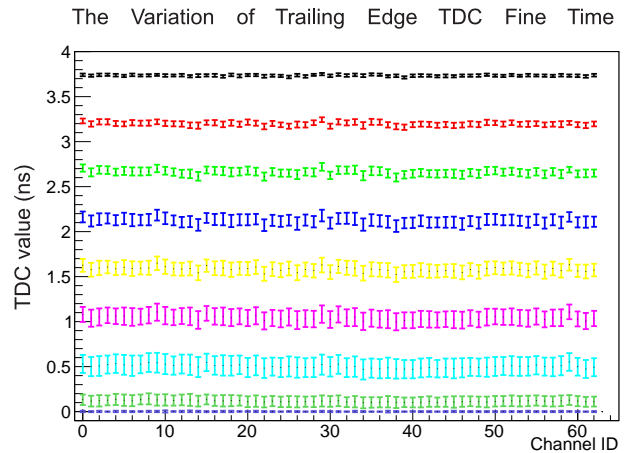


(b)

Fig. 5: (a) The distributions of measurement precision of the trailing (Negative) edge for the channel 5 in all 93 TDCs. The root-mean-square (RMS) value of the distribution is about 11 ps. (b) The RMS values of measurement precision of trailing and rising edges as a function of channel ID in all 93 TDCs.



(a)



(b)

Fig. 6: Using all the histograms we have for each channel, we use The mean value and the associated variation (shown as error bar) of each TDC fine-time bin as a function of channel ID in the measurement of (a) the leading edge and (b) the trailing edge.

#### A. Cross Talk

For the test of cross talk, signals out of phase by one clock cycle are sent to the TDC channels sequentially. For instance, the pulser gives a signal to channel #1, and after 14.5834 ns, the pulser gives a signal to channel #2 and so forth. Therefore each TDC channel receives a signal at different time during 64 periods of 14.5834 and then the pulser sends a common stop signal to the TDC. In this way we collect at least 2000 events for the test and that only one single channel is fired within each clock cycle for the recorded events is required for passing the check of cross talk.

#### B. Measurement Precision

The measurement precision is measured for every channel of 93 TDCs so that the channel-by-channel and board-by-board

systematic variation could be studied. For example the board-by-board variation of time precision for channel 5 is about 11 ps, as shown by the root-mean-square (RMS) value of the distribution in Fig. 5a.

By putting the variation ( $\sigma$ ) of the measurement precision of all 64 channels together, it gives us a picture of the channel-by-channel variation, as shown in Fig 5b. The horizontal axis is the channel ID and the error bar for each channel represents the variation of the measurement precision among these 93 modules. The results of the leading edges appear to be different than trailing edges.

With the calibration results, the TDC value of each bin can also be measured in each channel. For each fine-time TDC bin, we fill a histogram with 93 TDCs. As shown in the Fig. 6a and Fig. 6b, we use the mean value as the fine time, and the

variation ( $\sigma$ ) of distribution histograms as the error bar. Both channel-by-channel and board-by-board variations are shown.

### C. Multiple Hits

We use two ways to check for three-layer buffers. The DAQ system records more than 2000 events to see if the TDC record every hit.

1) *First-layer Buffer* : The first-layer buffer (“Pipe4” in Fig. 1b) can record up to 4 hits in 64 ns. The stored hits are constantly read out at 62.5 MHz and are put into a block of memory shared by a group of 4 channels. If a pulser is sending pulses with a width of 14.5834 ns and a period of 29.17 ns, the TDC should record up to 6 pulses. This test is performed by the pulser, which sends each channel 6 pulses in a row.

2) *Second- and Third-layer Buffer*: The second buffer is read out every 64 ns. In this test, the pulser send 8 pulses with a width of 14.5834 ns and a period of 72.917 ns to each channel.

### D. Multiple-hit Elimination

To test the multiple-hit elimination, the pulser sends 4 pulses to each channel with 14.5834-ns width and 29.1668-ns period. If the window of multiple-hit elimination is set to 36 ns, the TDC should only record the first and third pulses. The DAQ system records more than 2000 events to check if the multiple-hit elimination works properly.

## V. SUMMARY

We successfully develop a TDC firmware which digitizes multiple hits of both polarities and possesses a multiple-hit elimination functionality for removing after-pulses in the wire chambers and proportional tubes. A scaler module is implemented to record the number of hits for each channel. This new firmware significantly reduces the readout dead time by more than a factor of ten such that the SeaQuest experiment achieves efficient data collection. The production test shows that all 93 TDC modules meet the design specifications. Furthermore, we demonstrate the method of constructing the Wave Union TDC using the current multi-hit TDC with external wave union launchers. Based on a four-edge wave union, the measurement precision is improved by a factor of 2, up to 108 ps. A more significant improvement of measurement precision, up to 69 ps, is achieved by combining the approaches of Wave Union TDC and multiple-channel ganging. Many innovations made in this work should be useful for performing TDC measurement using FPGA.

## ACKNOWLEDGMENT

We thank Da-Shung Su for the development of the TDC hardware. The support from the SeaQuest collaboration, especially David Christian in carrying out this project is greatly appreciated.

## REFERENCES

- [1] Drell-Yan Measurements by Fermilab E-906/SeaQuest. [Online]. Available: <http://www.phy.anl.gov/mep/seaquest>
- [2] P. E. Reimer (Fermilab SeaQuest Collaboration). “Opportunities with Drell-Yan Scattering at Fermilab”, J. Phys.: Conf. Ser. 295 012011 (2011).
- [3] ProASIC3 Flash Family FPGAs with Optional Soft ARM Support. [Online]. Available: [http://www.actel.com/documents/PA3\\_DS.pdf](http://www.actel.com/documents/PA3_DS.pdf)
- [4] T. Rahkonen and J. Kostamovaara “The use of stabilized CMOS delay lines for the digitization of short time intervals”, IEEE J. Solid-State Circuits, vol. 28, no. 8, pp.887 -894 1993.
- [5] J. Wu and Z. Shi “The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay”, Proc. IEEE Nuclear Science Symp. Conf. Rec., pp.3440 -3446 2008.
- [6] R.Szplet, Z. Jachna, P. Kwiatkowski, K. Rozyc, “A 2.9 ps equivalent resolution interpolating time counter based on multiple independent coding lines”, Measurement Science and Technology, 24, 035904, pp. 15 2013.
- [7] J. Wu “Several key issues on implementing delay line based TDCs using FPGAs”, IEEE Trans. Nucl. Sci., vol. 57, no. 3, pp.1543 -1548 2010.
- [8] C. Favi and E. Charbon “A 17ps Time-to-Digital Converter Implemented in 65nm FPGA Technology”, Proceedings of the ACM/SIGDA international symposium on Field programmable gate arrays. 2009.
- [9] CODA main page. [Online]. Available: [https://coda.jlab.org/wiki/index.php/Main\\_Page](https://coda.jlab.org/wiki/index.php/Main_Page)