Field-Programmable Gate Array (FPGA) Firmware for the Fermilab E906 (SeaQuest) Trigger

Jinyuan Wu and Shiuan-Hal Shiu

Abstract— **Scintillating hodoscopes trigger firmware in a field-programmable gate array (FPGA) was implemented in a commercially-off-the-shelf 6U VMEbus module for the Fermilab E906 (SeaQuest) experiment. The FPGA receives up to 96-channel inputs and digitizes the leading edge time at 1 ns (LSB) resolution using time-to-digital converter (TDC) blocks in the firmware. Digital processes on the outputs of the TDC include adjusting delay channel-by-channel in 1-ns steps, setting coincidence range and re-align with the accelerator bucket clock. The re-aligned hits are further processed in trigger matrices. E906 uses four scintillating hodoscopes and various 3-out-of-4 (or 4-out-of-4) majority coincidence logic is used to generate valid track information as trigger primitives to form a final global trigger. Zero-suppressed TDC data are read out for each event and thus the module could be used as a 96-channel TDC beyond the functionality if trigger matrices.**

Index Terms— **Front End Electronics, TDC, Trigger, FPGA Firmware**

I. INTRODUCTION

T HE Fermilab E-906/SeaQuest experiment is one of serial fixed target experiments designed to measure the partonic structure of nucleons via the Drell-Yan process[1] produced in the collisions of hadrons[2-4]. The quarks and antiquarks from beam and target hadrons annihilate into virtual photons and then decay into two muons with opposite charge, di-muon pair, to be detected. The primary physics goal of E906/SeaQuest experiment is to determine the antiquark ratio: dbar/ubar at intermediate Bjorken-x region. In addition, the amount of energy loss of a fast parton (quark) travelling through the cold nuclear matter could be extracted.

This experiment uses the 120 GeV/c proton beam extracted from the Main Injector of Fermilab. The layout of E-906/SeaQuest spectrometer is shown in Fig.1. The system consists of a solid iron focusing magnet which also works the hadron absorber and beam dump, and a large open aperture magnet, KMAG, for the precise determination of the momenta of muon tracks. Between these two magnets, there are a few

multi-wire proportional chambers for measuring the trajectories of charged muons. A final large iron absorber with proportional tubes provides the muon identification. Behind each wire chamber and the proportional tube, a scintillation hodoscope plane is set where the readout from each channel is used for fast triggering in selecting out the relatively rare Drell-Yan events where dimuon pairs pass through the whole spectrometer.
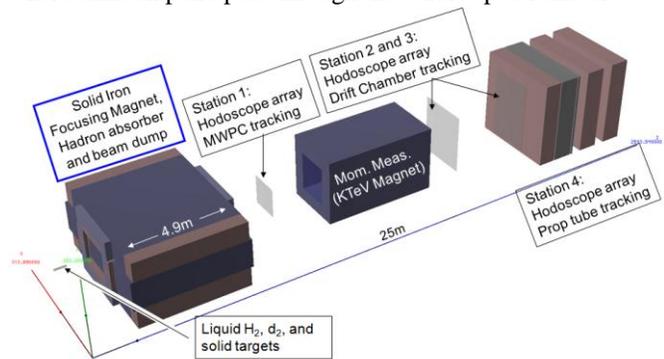

Fig. 1. The layout of Fermilab SeaQuest experiments spectrometer.

The trigger electronics hardware used is a commercially-off-the-shelf 6U VMEbus module (CAEN V1495) which contains a field-programmable gate array (FPGA) with 20,060 logic elements (Altera EP1C20F400C6)[5]. The FPGA receives up to 96 channels inputs and digitizes the leading edge times at 1 ns (LSB) resolution using time-to-digital converter (TDC) blocks in the firmware [6-8].

In this document, we will first describe the TDC blocks utilized in the firmware and the processing functions on the digitized hit times. The trigger matrix for finding muon tracks will also be discussed.

II. THE TDC BLOCK

The TDC is based on the simple multi-phase sampling scheme as shown in Fig. 2.

In this design, the input is buffered with a logic element, and then sent to four registers with equal propagation delays. The four registers are connected to four internal clocks each with 90-degree phase difference. The 0- and 90-degree clocks are generated in a phase-lock-loop (PLL) clock synthesizer and their inversions are used for 180- and 270-degree clocks. The 4 phases of 250 MHz clocks are used, the input signal is sampled

every 1 ns, which forms a TDC with 1 ns bin size. Note that the sampling interval is 1 ns but each register operates at 250 MHz, rather than 1 GHz.

A transfer to the 0-degree clock domain occurs in the second and third stages of the pipeline. Depending on arrival time, the transitions of the input logic levels are recorded at different locations within the four registers. The position of the input signal edge being sampled represents the arrival time and is encoded as lower two bits, T0 and T1 of the time value plus a data valid signal DV. The higher bits TS are generated with a coarse time counter. The coarse time, fine time and data valid signal is sent to later stages for further operations.
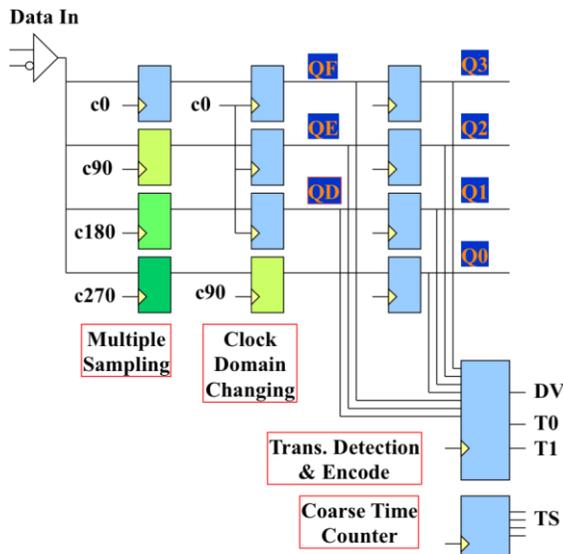

Fig. 2. The TDC structure inside the SeaQuest trigger FPGA

A function of transition edge regulation prevents ultra short pulses due to input circuit ringing from being mistakenly digitized. In this design up to four consecutive bits in the bit pattern QD to Q3 are used by a look-up table in FPGA logic element to determine if a sampling point is at the edge of a well-established pulse. For example, due to impedance mismatch caused by cable aging, signal reflection in a long cable may produce a bit pattern "000010111" on QD to Q3 with several transitions instead of an ideal pattern "000011111". One may design edge detecting logic functions similar as: Q1&(!Q0)&(!QF)&(!QE) to recognize a sub pattern "0001" as a valid transition edge, instead of Q1&(!Q0) which detects any "01" sub patterns as a transition edge. This way, only one valid transition edge will be detected in the bit pattern even with the present of input signal ringing. Recall that the using a look-up table in FPGA, one can implement "any" four-input combinational logic, satisfying the edge detection and pulse filtering requirements of an application.

Timing critical signal paths are controlled by placing the input buffer, multi-sampling registers and clock domain transfer registers in the FPGA to locations to assure equal propagation delays from input buffer to the sampling registers, resulting in uniform bin widths and thus minimizes differential non-linearity.

In each channel, the input buffer cell is placed in the logic

array block in the middle and the registers driven by the clock with 4 phases are placed in the blocks left and right of the center one to ensure equal propagation delays. Further left and right blocks contain registers for the clock domain transfer. Placement of other logic elements is relatively flexible and can be automatically placed with the compiler.

Placing logic element "manually" is a time consuming task but it is possible to use a short C program to do the work efficiently. The locations of the timing critical input buffer and flip-flops, (about 10 items per channel) for all TDC channels can be kept in the program with the FPGA internal coordinates. The designer may further arrange the location of each channel or channel group to adjust the input delay from input pins so that the skews between different channels are minimized. The program is coded to output an ASCII file that is pasted into the assignment file for compilation with the FPGA design software.

The authors wish to point out a technical detail that some designers neglect: the clock domain changing stage. The clocks with four phases cause the sampling registers to flip at different times and therefore they are to be brought to the same clock domain (c0 in our example) as the encoder and the rest of the system via the clock domain changing stage.

Note that in this stage, three registers uses c0 clock while the last one uses c90 clock which is a key point to avoid creating a critical timing path. Should the clock used in the last register be c0, the transition time between the sampling registers driven by c270 and the clock domain changing register driven by c0 would be 1 ns (at 250 MHz) which may cause the setup time unsatisfied if many TDC channels are to be packed inside an FPGA with a reasonable size. The transition time between the c270 and the c90 clocks is 2 ns that is much easier to be satisfied during compiling of the FPGA.

III. THE DELAY ADJUSTMENT

The digitized data are sent into RAM blocks used as pipelines as shown in Fig. 3. A requirement which usually is not found in digitizers and is special to the trigger application is the input delay adjustment. Input channels are allowed to have different signal propagation delays due to different cable lengths and differences in discriminator settings. The input delay in each channel is to be adjusted in the FPGA individually at 1 ns steps.
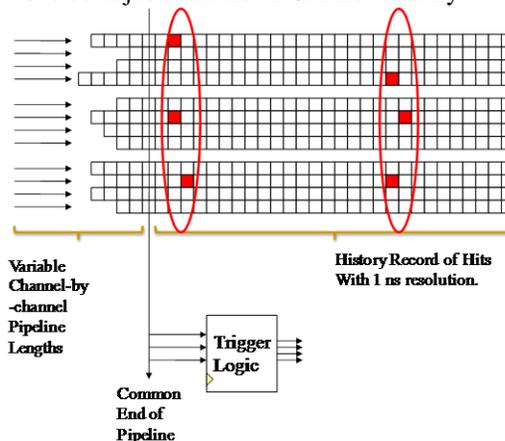

Fig. 3. The pipeline structure inside the FPGA

Each bin in the pipeline represents a 16 ns time interval and each memory location contains 4 bits for hit time in 1 ns (LSB) resolution plus 1 bit to indicate that the hit is valid. In each channel, a relative delay value of 0-255 ns is stored in an 8-bit register. While writing into pipeline, the lower 4 bits of the register and the TDC output are summed into a new 4-bit value to be written into the pipeline. If the carry from the sum of the lower 4 bits is 1, the data is delayed by one clock cycle before writing into the pipeline. The higher 4 bits of the register and the pipeline pointer counter are summed as the pipeline writing address. The channel-by-channel writing operations of a pipeline memory block serving 4 channels are in 250 MHz while the reading operations are in 62.5 MHz parallel for the 4 channels. This way, individual channel delays are compensated for at the output port of the pipeline.

In addition to regular inputs of the hodoscopes, the beam bucket RF signal at 53 MHz is also digitized in an identical TDC/pipeline channel. The pipeline outputs of all channels are checked with the beam buckets within a user specified timing window so that the input hits are realigned with the beam buckets. The re-aligned hits are further processed in trigger matrices.

The pipeline is also used as the event storage. When a global trigger is received, the pipeline stops, a history record of 16 time slots (TS) for all 96 channels i.e., $96*16 = 1536$ words will be copied from the pipeline to the VME interface buffer at 62.5 MHz, which takes 24.576 μs. Most of the time slots are empty and are suppressed during the copying process and only data of non-empty time slots are stored in the interface buffer. The buffer capacity is 256 hits but the users can decide how many to be readout in each event. The copying sequence loops the hits of channels 0-95 in latest time slot first and then earlier time slots. Therefore, if there are more than 256 hits within the 16 time slots (which is unlikely), the latest hits will be readout. In most time there are less than 256 hits and the unfilled words in the buffer will be marked as end of block. The VME readout program can stop reading out additional words upon seeing the end of block mark.

Zero-suppressed TDC data are read out for each event and the module can be used as a 96-channel TDC when the trigger matrices are disregarded.

IV. TRIGGER MATRICES

In E-906 /SeaQuest experiment we are interested in opposite sign muon pair from Drell-Yan process. However J/ψ and ϒ will also decay into an opposite-sign muon pairs. The decay of D-meson, π-meson and other mesons will involve a single muon and random coincidence of them will form a muon pair in our detectors.

In order to have a reliable trigger system which can effectively separate the signal events from the background ones, scintillator hodoscopes were placed throughout the spectrometer. Scintillator hodoscope is a detector array composed by scintillation detectors. Scintillation detector is a rapid response detector; the response time for scintillation

detector is about 10 ns. The trigger system examines the overall hit patterns of scintillator hodoscope for decision making.

We use a FORTRAN based Monte Carlo simulation (Fast MC) to simulate all possible hit patterns generated from interested Drell-Yan process. Fast MC contains the information about the detector geometry and magnetic field strength, and provides a Drell-Yan, J/ψ and open charm event generator. It will output the complete information of momentum and position in the trajectory of charged muons in the spectrometer. The schematic diagram of the muon passes each station of hodoscope is shown in Fig. 4. The invariant mass of Drell-Yan muon pair is proportional to the opening angle of muon pair. A large-mass Drell-Yan event would lead to the muon pair composed of large transverse momentum muons. Since the muon pairs from J/ψ and meson decay coincidence are mostly with low mass, they could be separated out by requiring the hit position on the hodoscope corresponding to large transverse momentum muons.
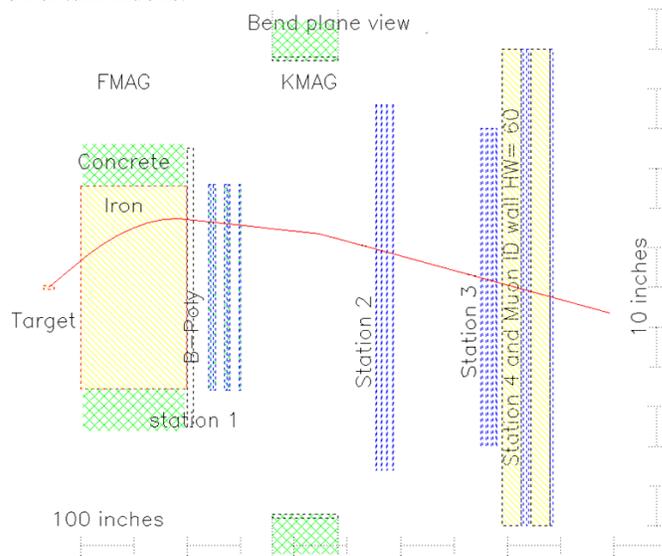

Fig. 4. A Bend plane view of the trajectories of one of the two muons

After sampling all possible interested hit patterns in the Monte Carlo framework, we convert the information into a look-up-table or a trigger matrix which is implemented into the FPGA. The trigger matrices are composed by some basic logic operator like "AND", "OR", "NOT AND", and the logic elements are related to the real hodoscope hits. We use the hit information from four stations of scintillating hodoscopes and various 3-out-of-4 (or 4-out-of-4) majority coincidence logic is used to generate valid track information as trigger primitives to form global trigger. In the trigger matrices, each gate element takes four inputs, and then output the results to next pipeline level. When the re-aligned hits are sent to the trigger matrices, the patterns are compared with the matrices. If the hits consist with the interested events, than a trigger will sent out. In E-906/SeaQuest experiment, we have two different main triggers for interested muon pairs and four pre-scale triggers for calibration.

The hardware trigger system is composed of 5 pieces of FPGA modules. Four of them are used as 1st level track finder

which identifies muon tracks in upper Y-direction hodoscope, upper X-direction hodoscope, lower Y's, and lower X's. The last module works as the 2nd level track correlator. The track correlator receives the information from the upstream track finder and achieves the final decision on the trigger type. Overall it takes 442 ns for the process of the 1st level track finder and 330 ns for the final trigger decision.

The VHDL codes of the trigger matrices are generated automatically using a short FORTRAN program converting the required trigger table into the hardware logic description. This way, human coding is reduced on frequently changed trigger condition implementation to reduce both work load and possible errors.

V. DISCUSSION

The digitization and digital processing task is simple if only a few channels are to be implemented, but it becomes a challenge requiring design with extra cares when a large number of channels are to be fit into the FPGA. On the other hand, the power supply capability of the module restricts using high clock frequency (250 MHz) in only a small portion of the FPGA, limiting the possibility of using fast clock in exchange for smaller silicon area. In order to create a working firmware within these boundary conditions, the design requirements are carefully analyzed and each block is designed with smallest possible silicon area and clocked at lowest possible frequency.

The firmware has been implemented and put in operation in the experiment. The compiling report shows that the preprocessing (including TDC, delay adjustment and hit realignment) takes about 8000 logic elements or 40% of the FPGA (20060 logic elements total) and the remaining resources are available for the trigger matrix. Depending on the demands of the experiment, the complexities of the trigger matrices are various. Different versions of the trigger matrices use from 10% up to 50% of the FPGA logic elements.

REFERENCES

[1] S.D. Drell and T.M. Yan, *Phys. Rev. Lett.* **25** (1970) 316
[2] E. A. Hawker et al. (FNAL E866/NuSea Collaboration) Phys. Rev. Lett, 80 3715 (1998).
[3] J.-C. Peng et al. (FNAL E866/NuSea Collaboration) Phys. Rev. D 58, 092004 (1998).
[4] R. S. Towell et al. (FNAL E866/NuSea Collaboration) Phys. Rev. D 64, 052002 (2001).
[5] Altera Corporation, "*Cyclone Device Handbook*", (2008) available via: {http://www.altera.com/}
[6] J. Wu, S. Hansen & Z. Shi, "ADC and TDC implemented using FPGA", in *Nuclear Science Symposium Conference Record, 2007 IEEE*, Oct. 26 2007-Nov. 3 2007 Page(s):281 - 286.
[7] J. Wu, "Several Key Issues on Implementing Delay Line Based TDCs Using FPGAs," in *IEEE Transactions on Nuclear Science,* 2010, Pages 1543 - 1548, vol. 57.
[8] H. Sadrozinski & J. Wu, "Applications of Field-Programmable Gate Arrays in Scientific Research", Taylor & Francis, December 2010.