ADC and TDC Implemented Using FPGA

Jinyuan Wu, Sten Hansen and Zonghan Shi

***Abstract*— Several tests of FPGA devices programmed as analog waveform digitizers are discussed. The ADC uses the ramping-comparing scheme. A multi-channel ADC can be implemented with only a few resistors and capacitors as external components. A periodic logic levels are shaped by passive RC network to generate exponential ramps. The FPGA differential input buffers are used as comparators to compare the ramps with the input signals. The times at which these ramps cross the input signals are digitized by time-to-digital-converters (TDCs) implemented within the FPGA. The TDC portion of the logic alone has potentially a broad range of HEP/nuclear science applications. A 96-channel TDC card using FPGAs as TDCs being designed for the Fermilab MIPP electronics upgrade project is discussed. A deserializer circuit based on multi-sampling circuit used in the TDC, the "Digital Phase Follower" (DPF) is also documented.**

***Index Terms*— Front End Electronics, TDC, ADC, Deserializer, FPGA Firmware**

I. Introduction

INTRINSICALLY FPGA is a digital device. However, with suitable use of the FPGA resources, it is possible to use FPGA to digitize multi-channel analog waveforms. The digitized waveforms can be directly processes in the FPGA. There are several possible schemes of digitizing analog signals. One of the schemes we used in our FPGA ADC study is based on the ramping-comparing approach as shown in Fig. 1.

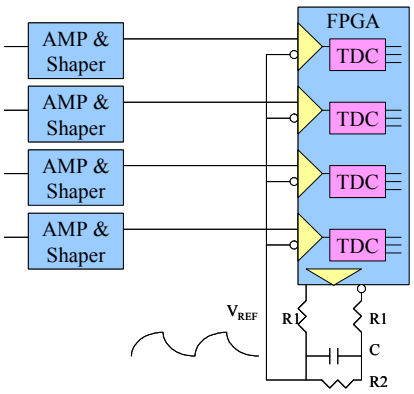

Fig. 1. FPGA-based ADC

The analog inputs are directly connected to the FPGA input pins. A passive RC network is connected to the FPGA output pins so that a periodic reference voltage ramp can be generated. The differential input buffers are used as comparators to generate logic transitions inside the FPGA when the reference voltage ramps across the input voltage levels. The transition times are digitized by the TDC block implemented in the FPGA. Since the period, the RC network parameters and the starting time of the ramps are known the input voltage levels can be derived from the transition times.

In today's FPGA devices, differential input buffers are good comparators within a sufficiently large range of input voltage levels, since they are designed to be compatible with various differential signaling standards. Many comparator-based ADC schemes can be implemented with FPGA. For example, with the delta-sigma scheme [1, 2], the signal can be tracked promptly yielding smaller digitization errors at a cost of higher FPGA resource usage (typically, 4 I/O pins per channel). With Wilkinson rundown scheme [3, 4], charge integration of narrow pulse can be combined with the digitization, although more external analog circuits are needed. The ramping-comparing scheme we studied here (or single-slope ADC based on classification in Reference [1], although both ramping slopes can be utilized) is a suitable choice for applications with large channel count of relatively slow signals. (In some references, the single-slope scheme is mistakenly referred as Wilkinson ADC that is based on dual-slope principle.)

A key functional block, Time-to-Digit-Converter (TDC) is needed in FPGA. There are two TDC schemes that can be implemented in FPGA: delay chain scheme [5, 6] and multi-sampling scheme [7]. The TDC we used in this work is multi-sampling scheme with quad clock as in Reference [7]. In Reference [7], four sets of sample, edge detect, pulse filter and count latch are driven by four clocks with 90^o phase separations. These four sets of data collected by four sets of circuits are excessive and they become valid at different time, which makes the meta-stability elimination and encoding logic complicate. In our TDC design, the four samples are transferred into a bit pattern in a single clock domain immediately and only one set of edge detect, pulse filter and count latch circuit is used. The meta-stability is limited at the sampling stage only and in fact, the meta-stability in sampling stage does no harm but carrying the input signal arrival time information. The decoding becomes very simple in our design. The detail is described in Section II.

The TDC in FPGA alone is already very useful. The TDC card designed for Fermilab MIPP upgrade project is

Manuscript received October 30, 2007. This work was supported in part by Universities Research Association Inc. under Contract No. DE-AC02-76CH03000 with the United States Department of Energy.

The authors are with Fermi National Accelerator Laboratory, Batavia, IL 60510 USA (phone: 630-840-8911; fax: 630-840-2950; e-mail: jywu168@fnal.gov).

documented in this paper.

The multi-sampling structure can have other applications. A deserializer circuit known as "Digital Phase Follower" (DPF) is also documented. Using DPF, any FPGA input can be used to receive serial data without needing dedicated deserializer that is only available in high-end FPGA families. The DPF can compensate input data phase drift not only due to cable temperature variation, but also due to crystal oscillator frequency difference between transmitter and receiver.

II. TDC Implemented in FPGA

The TDC inside the FPGA is based on multi-phased clock. The input of the TDC is sampled by four registers with four phases of the clock as shown in Fig. 2.

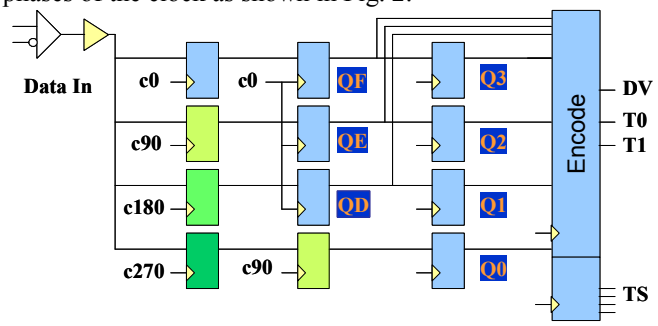

Fig. 2. The multi-sampling based TDC circuit

The input is buffered, and then sent to four registers with equal propagation delays. The four registers are connected to four internal clocks each with 90° phase difference. The 0° and 90° clocks are generated by the phase-lock-loop (PLL) clock synthesizer and their inversions are used for 180° and 270° clocks. Depending on arrival time, the transitions of the input logic levels are recorded at different locations within the four registers. The clock frequency used in our Altera Cyclone FPGA device [8] (EP1C6Q240C6) is 360 MHz, which provides a time resolution of 0.69 ns (LSB). A transfer to a single phase clock domain occurs in the second and third register layers. Then the arrival time of input signal is encoded into two time bits (T0 and T1) and a data valid signal (DV). A counter provides the upper order time bits.

Transition edge detection and pulse filtering logics are included in the encoder. For many applications, a simple leading edge encoding is sufficient. In some applications, for example, to estimate input pulse charge from a wire chamber, both leading and trailing edges may be digitized. An additional output indicating the type of edge may be needed in this case. The function of pulse filtering prevents ultra short pulses due to input circuit ringing from being mistakenly digitized. In our 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. 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 as shown in Fig. 3. This symmetric placement assures equal propagation delays from input buffer to the sampling registers, resulting in uniform bin widths and thus minimizes differential non-linearity.

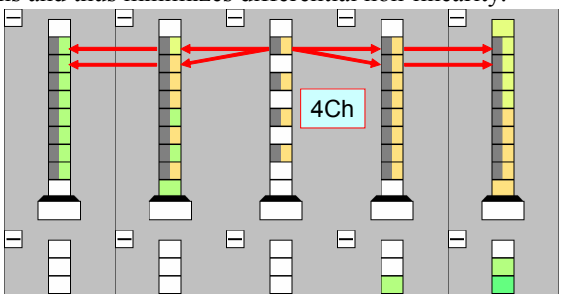

Fig. 3. Timing critical paths in the FPGA

The logic element layout is done "manually" with a spread sheet. The locations of the input buffer and flip-flops, (about 10 items per channel) for all TDC channels are kept in the spread sheet. In Cyclone FPGA devices, four channels are grouped together in five logic array blocks (LAB) as shown above. The designer may further arrange the location of each four-channel group to adjust the input delay from input pin to the group so that the skews between different channel groups are minimized. The spread sheet is coded to output an ASCII file that is pasted into the assignment file for compilation with the Quartus II Altera FPGA design software.

III. Test Results of FPGA Based ADC

Several tests of FPGA ADC are done with circuits shown in Fig. 1 with two sets of values of R1, R2 and C to achieve different time constants for the ramping reference voltage.

A. Quasi-linear Reference Voltage

In the first configuration, the values of R1 = 50Ω, R2 = 100Ω and C = 1000pF. The FPGA drives the RC network with a toggling rate of 11.25 MHz in differential 3.3V level. The reference voltage is nearly a triangle wave with not much exponential feature. The oscilloscope traces of the input voltage to the ADC and the reference voltage are shown in Fig. 4. The input to ADC is a sequence of four pulses with different widths and peak amplitudes.

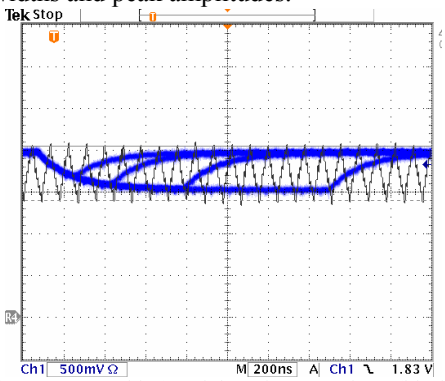

Fig. 4. The input to ADC (blue) and the reference voltage (black)

The input signal is crossed by the reference voltage twice every 88ns by both leading and trailing ramps and each crossing creates a flipping edge that is digitized by the TDC. The sampling rate is approximately 22.5 M samples/sec,

(although the sampling intervals between the points sampled by the leading and trailing ramps are not the same).

The transition times sampled by both the leading and trailing ramps are digitized by the TDC with 6-bit measurement range. The raw TDC data are shown in Fig. 5(a). The transition times are further converted to input voltage levels in natural unit as plotted in Fig. 5(b).

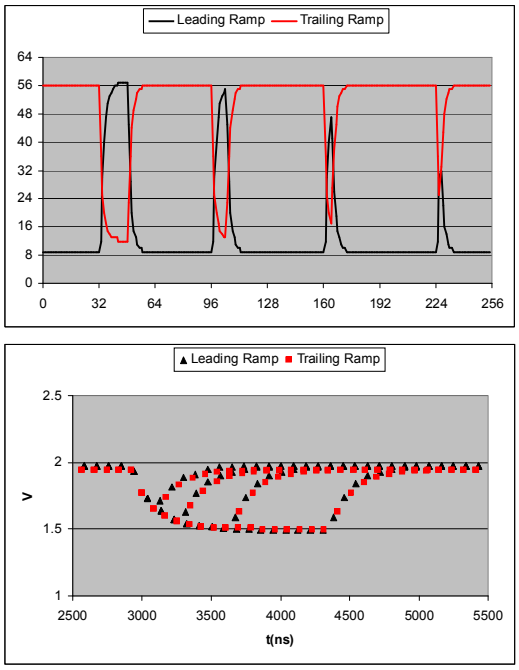

Fig. 5. (a) Raw data from TDC (b) Digitized waveform

It should be pointed out that the TDC values represent not only the voltage levels at the sampling points, but also the sampling times. In high precision applications, the differences of the sampling times should be taken into account but it is not too difficult to do so.

B. *Exponential Reference Voltage*

The exponential discharge property of the RC networks can be used to increase the dynamic range of the ADC. In the second configuration, the values of R1 = 50Ω, R2 = 100Ω and C = 150pF. The reference voltage shown in Fig. 6 has a short time constant.

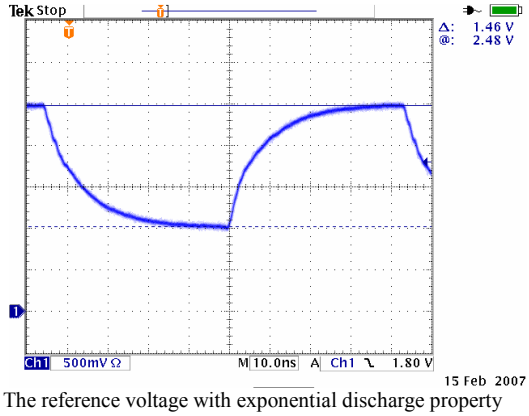

Fig. 6. The reference voltage with exponential discharge property

The exponential reference voltage samples the input waveform shown in Fig. 7(a). (Note that the oscilloscope time scales for Fig. 6 and 7(a) are different but the voltage scales are them same.) It can be seen from Fig. 7(b) that a smoother waveform is digitized by the trailing ramp. This test shows a 6-bit measurement range at 22.5 M samples/sec while the dynamic range of the trailing ramp samples is approximately 8 bits.

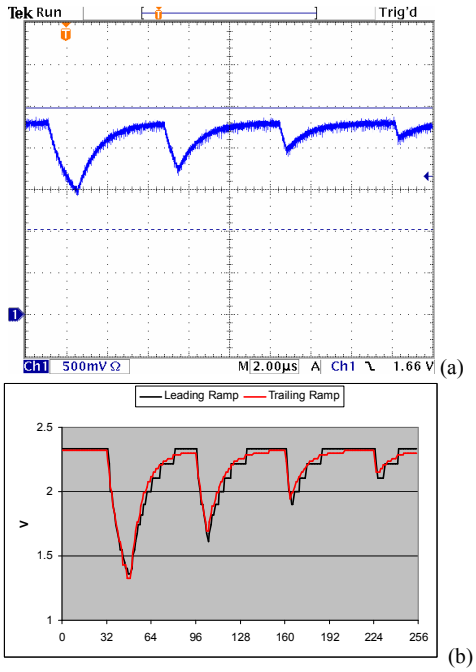

Fig. 7. (a) Input waveform (b) Digitized waveform

Passive components are chosen for the ramping reference voltage generation network primarily for simplicity. The ramping voltage from passive RC network is intrinsically non-linear which sometimes is viewed as a disadvantage. In FPGA, however, correcting nonlinearity is merely a transform via a look-up table. In our example here, the exponential voltage ramp can be further used to increase measurement dynamic range, which becomes an advantage.

In many applications, only relative precision in a measurement is needed, i.e., finer measurements are only needed for small signals while for larger signals, coarser measurements are sufficient.

IV. The Enclosed Loop Micro-Sequencer

A 96-channel TDC card is designed for the Fermilab MIPP experiment electronics upgrade project as shown in Fig. 8.

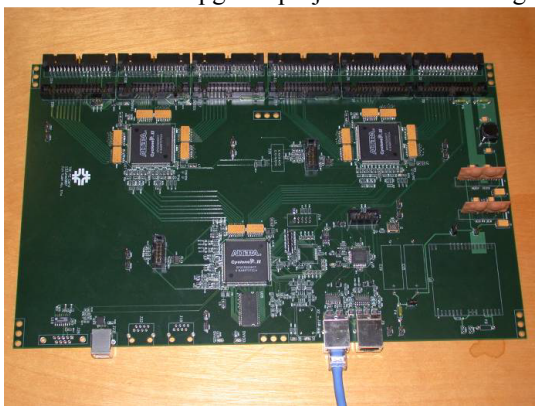

Fig. 8. A TDC card designed for MIPP project.

The TDC card (named TDC_FE) has three Altera Cyclone II FPGA devices [9] (EP2C5Q208C7, $16 ea.): two named MIPP_TDC and one named MIPP_DCC. Each MIPP_TDC digitizes 48 channels of inputs and the TDC resolution is chosen to be 1.18ns (LSB). The MIPP_TDC then sends data to MIPP_DCC which interfaces with a 16MB SDRAM and the daisy-chained interface to ship data back to the DAQ controller. A TDC_FE card supports 96 input channels. Some parameters are shown in Table I.

TABLE I
MIPP TDC CARD PARAMETERS

Main Input Clock: RF	53.102MHz, 18.8ns
Internal Clock: CK212	212.4MHz, 4.708ns
TDC Resolution: LSB	1.18ns
Hit Rate Limiter Setting	4hits/256CK212
Double Hit Min. Separation	4xCK106, 37.7ns
Event Window	2 x 1.2μs
Absolute Maximum Hits/event/48ch	123hits+5header/trailers
Event Maximum Size/96ch	256 x 2Bytes
SDRAM port data rate	53MHz x 2Bytes
Number of events/spill	<32K
Absolute Maximum data/spill/FE card	8M words
Absolute Maximum data/spill/8 FE	128MB, 1280Mbits
Readout Chain Data Rate	26.5Mbits/s
Absolute Maximum Spill Readout Time	48.3sec

Each channel in the MIPP_TDC has a hit limiter. The time is split into 1.2 μs (or 64 RF, 1 RF = 1/53MHz) periods which are counted after the reset at the start of each accelerate beam spill. In each of the 1.2 μs window, the number of hits in each channel is limited to 4 hits.

A channel group contains 4 channels. Every 4 cycles of CK106 (106MHz), hit data in each channel is written to a hold-shift register. The hold-shift register then shift data of the 4 channels, one in each CK106 cycle, to the zero suppression block. Therefore, the minimum hit separation in each channel is 4 x 1/106MHz = 37.7 ns.

The zero-suppressed data from the 4-channel groups, i.e., only the valid hits are stored in the L1 pipeline buffers. A pipeline buffer is organized as 16 blocks of 16 words with 16-bit/word which uses a M4K RAM. Each hit is stored as a 16-bit word; each channel can have maximum 4 hits in 1.2 μs. Therefore, a 16 words block holds data of 4 channels in 1.2 μs. The entire L1 pipeline length is 16x1.2 μs = 19.3 μs.

Up to about 18 μs trigger latency are supported. For each trigger, data in two 1.2 μs time windows are readout, which corresponding to 2 16-word blocks in the L1 pipeline buffer. Data from 16 channels are grouped together, taking 1.2 μs to transfer from the L1 pipeline buffer to the EV buffer. Data of 48 channels are transferred in 3.6 μs.

Truncating happens while storing data into EV buffer if too many channels are hot. The data for an event stored in the EV buffer contains 2 headers and 3 enders, one at the end of each 16-channel group. The total number of hits allowed in 48 channels is 123. The truncated data block is not good for experiment data, however, it contains all necessary header and enders to avoid hanging up the remaining readout system.

Up to 256 16-bit words/event for 96 channels are stored in the 16MB SDRAM. A total of 32K events can be held in the SDRAM.

Daisy chained 8 TDC_FE cards are readout through a 26.5Mbit/s data link to the controller. The total time to readout 8 x 16 MB is about 48.3 sec.

This design is an example of a low cost multi-channel TDC suitable for a broad range of applications.

V. THE DIGITAL PHASE FOLLOWER

The multi-sampling structure in our TDC may be utilized in various applications. A deserializer suitable for low cost FPGA implementation: the "Digital Phase Follower" (DPF) [10][12] is documented here as an example. (See also: [11]) The DPF can be viewed as a TDC immediately followed by a data extractor.

Serial communication is a popular data transmission scheme since the communication channel is simply a twist-pair of a cable. There are serial transceivers available in several FPGA families with operating bit rate higher than 1 Gb/s. However, the costs of FPGA with build-in transceivers are normally higher than the comparable devices without transceivers. There are applications of serial communications to be implemented in low-cost FPGA families where there are no dedicated transceivers, but relatively lower bit rates are sufficient. The Digital Phase Follower (DPF) is developed to fill this gap.

The transmitter of serial data is relatively simple which is a parallel to serial converter implemented either with a shift register or dual-port memory with single-bit output port. The receiver is more complicate since the cable delay causes the data to arrive at any possible phases. When temperature of the cable varies, the phase of the serial data may drift away from the original phase. If the clocks of the sender and the receiver are not derived from the same source, a continuous and indefinite phase drift is expected.

The DPF uses multiple samples of the data stream to detect and to keep track of the input data phase. In each bit time, the input data is sampled 4 times at 0, 90 180 and 270 degrees. The 4 clocks (or 0 and 90 degrees plus their inverted versions) can be generated with a PLL block now available in most low-cost FPGA families. The block diagram of the DPF is shown in Fig. 9.

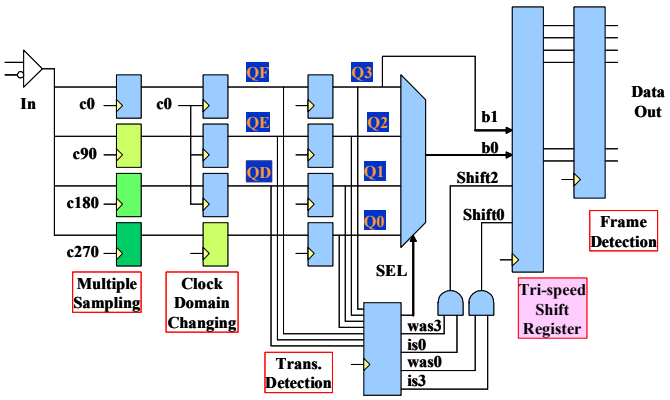

Fig. 9. The Digital Phase Follower (DPF)

The multi-sampling part is the same as in FPGA TDC discussed before. In fact, the operation of the DPF is based on the transition time of the input data stream. After multi-sampling, the sampled pattern is first converted to the 0 degree clock domain. Then 7 samples, QD to Q3 are sent to the transition detection logic to find the relative phase of the input data as shown in Fig. 10. Note that the sample pattern jumps up 4 bits every clock cycles, i.e., QD jumps to Q1, QE to Q2 and QF to Q3 which is obvious from the pipeline structure shown in Fig. 9.

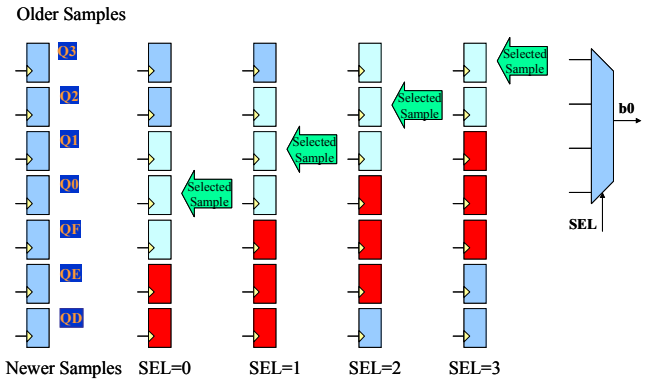

Fig. 10. The Transition Detection in the Digital Phase Follower

The designers may choose to detect both 0-to-1 and 1-to-0 transitions, but detecting only one transition is recommended since the raising and falling time of the input circuit may be different. Once the first transition is seen, the location of it is registered and the data sample sufficiently far away from the transition points is selected as an input of the shift register in the later stage. For example, when a 0-to-1 transition is seen between QF and QE, i.e., (QF==0)&&(QE==1), the sample Q0 is selected, and so on.

When the phase of the input data drifts away from the original point, the transition at different location is detected. The sample point being selected follows the change of the transition location change accordingly.

As mentioned earlier, the input data phase may drift indefinitely if the clocks of the sender and receiver have very close but slightly different frequencies. Even in the systems with same clock source for the sender and receiver, the phase drift due to cable temperature variation may also be bigger than a bit time.

We can assume that the phase drift rate is not too high so that position of the current transition is either the same as what previously detected, or +1 or -1 from the previous position. Under this assumption, there are two possible cases for the bit phase drifting out of one bit time. The two cases: "was-0-is-3" and "was-3-is-0" are shown in Fig. 11.

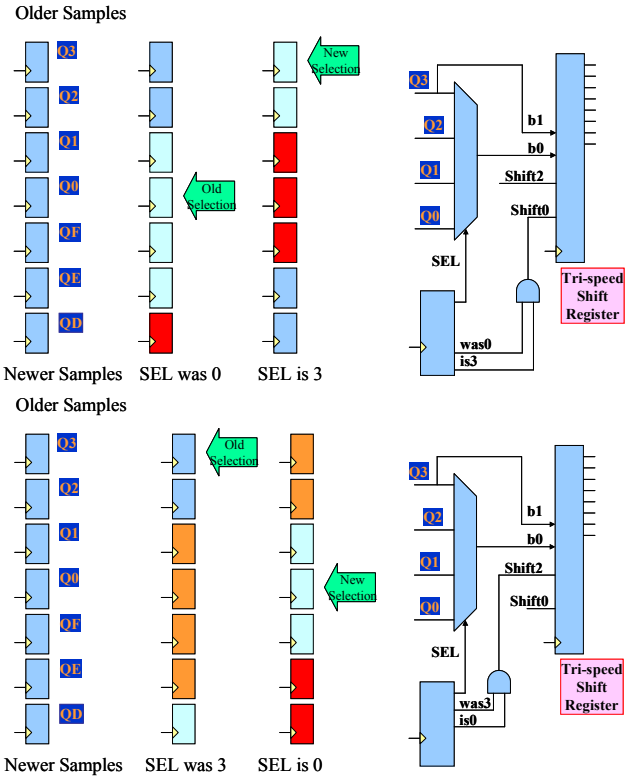

Fig. 11. The Digital Phase Following Processes

The first case is finding the transition between Q2 and Q1 requesting that the sample Q3 being selected in current clock cycle while the previously registered selection was Q0. This was-0-is-3 case indicates that the input data clock is slower than the local clock. The selection point is actually drifted from Q0 to QF. However, to prevent the sampling point from drifting down indefinitely, it is wrapped over to Q3 instead of QF. Since the current sample at Q3 has been shifted into the shift-register in previous clock cycle (which was Q0), the shift register stops shifting for one clock cycle to compensate for the slower input data clock.

The second case is finding the transition between QF and QE requesting that the sample Q0 being selected in current clock cycle while the previously registered selection was Q3. This was-3-is-0 case indicates that the input data clock is faster than the local clock. The selection point is actually drifted from Q3 to Q4 (which is not implemented). However, to prevent the sampling point from drifting up indefinitely, it is wrapped over to Q0. In this situation, two sample points, i.e., Q3 and Q0 must be pushed into the shift register causing it shift by 2 bits in the current clock cycle to compensate for the faster input data.

The shift register normally shifts by 1 and it shifts by 0 or by 2 in the "was-0-is-3" or "was-3-is-0" cases, respectively, which is why it is called "tri-speed shift register". Typical de-

serialization circuits, either in FPGA or single IC chip, recover the receiving clock using either PLL or clock swapping schemes. The digital phase follower is a pure digital circuit and the clock recovery is avoided.

VI. Conclusion

Multi-sampling based TDC has been studied, implemented in low cost FPGA and bench tested.

Three applications: multi-channel FPGA-only TDC, FPGA-only ADC and a deserializer "Digital Phase Follower" are discussed. Interfacing FPGA directly with the continuous variables (arrival time and input voltage) eliminates external devices and simplifies system design. The measurement made can be processed immediately in the FPGA without having to pass data via on board busses.

References

[1] P. Allen & D. Holberg, *CMOS Analog Circuit Design*, Second Edition, New York, New York: Oxford University Press, 2002.

[2] B. G. Tomov & J. A. Jensen, "A new architecture for a single-chip multi-channel beamformer based on a standard FPGA," in *Ultrasonics Symposium, 2001 IEEE*, 7-10 Oct. 2001 Page(s):1529 - 1533 vol.2.

[3] D. Wilkinson, "Blood, Birds, and the Old Road," in *Annu. Rev. Nucl. Part. Sci*. 1995, Pages 1-39, vol. 45.

[4] G. Blanar, K. Roberts & R. Sumner, "A new concept for a multi-range low cost calorimeter ADC," in *Nuclear Science Symposium and Medical Imaging Conference, 1994 IEEE Conference Record*, 30 Oct.-5 Nov. 1994 Page(s):999 - 1001 vol.2.

[5] Jinyuan Wu, Zonghan Shi & I. Y. Wang, "Firmware-only implementation of time-to-digital converter (TDC) in field programmable gate array (FPGA)," in *Nuclear Science Symposium Conference Record, 2003 IEEE*, 19-25 Oct. 2003 Page(s):177 - 181 Vol. 1.

[6] S. S. Junnarkar, et. al., "An FPGA-based, 12-channel TDC and digital signal processing module for the RatCAP scanner," in *Nuclear Science Symposium Conference Record, 2005 IEEE*, Volume 2, 23-29 Oct. 2005 Page(s):919 - 923.

[7] M. D. Fries & J. J. Williams, "High-precision TDC in an FPGA using a 192 MHz quadrature clock," in *Nuclear Science Symposium Conference Record, 2002 IEEE*, 10-16 Nov. 2002 Page(s):580 - 584 vol. 1.

[8] Altera Corp., "*Cyclone FPGA Family Data Sheet*", (2003) available via: {http://www.altera.com/}

[9] Altera Corporation, "*Cyclone II Device Handbook*", (2007) available via: {http://www.altera.com/}

[10] Jinyuan Wu, "Digital Phase Follower -- Deserializer in Low-Cost FPGA", BTeV Document 3295-v1, Fermi National Accelerator Laboratory, (2004), available online: {http://www-btev.fnal.gov/cgi-bin/public/DocDB/ShowDocument?docid=3295}

[11] Nick Sawyer, "Data Recovery", Xilinx Application Note 224, v2.5 (2005), available via: {http://www.xilinx.com}

[12] Jinyuan Wu, "*How to Design Compact FPGA Functions-- Resource Awareness Design Practices*", (2006) available via: {http://www-ppd.fnal.gov/EEDOffice-W/Projects/ckm/comadc/CompactFPGAdesign.pdf }